IBM

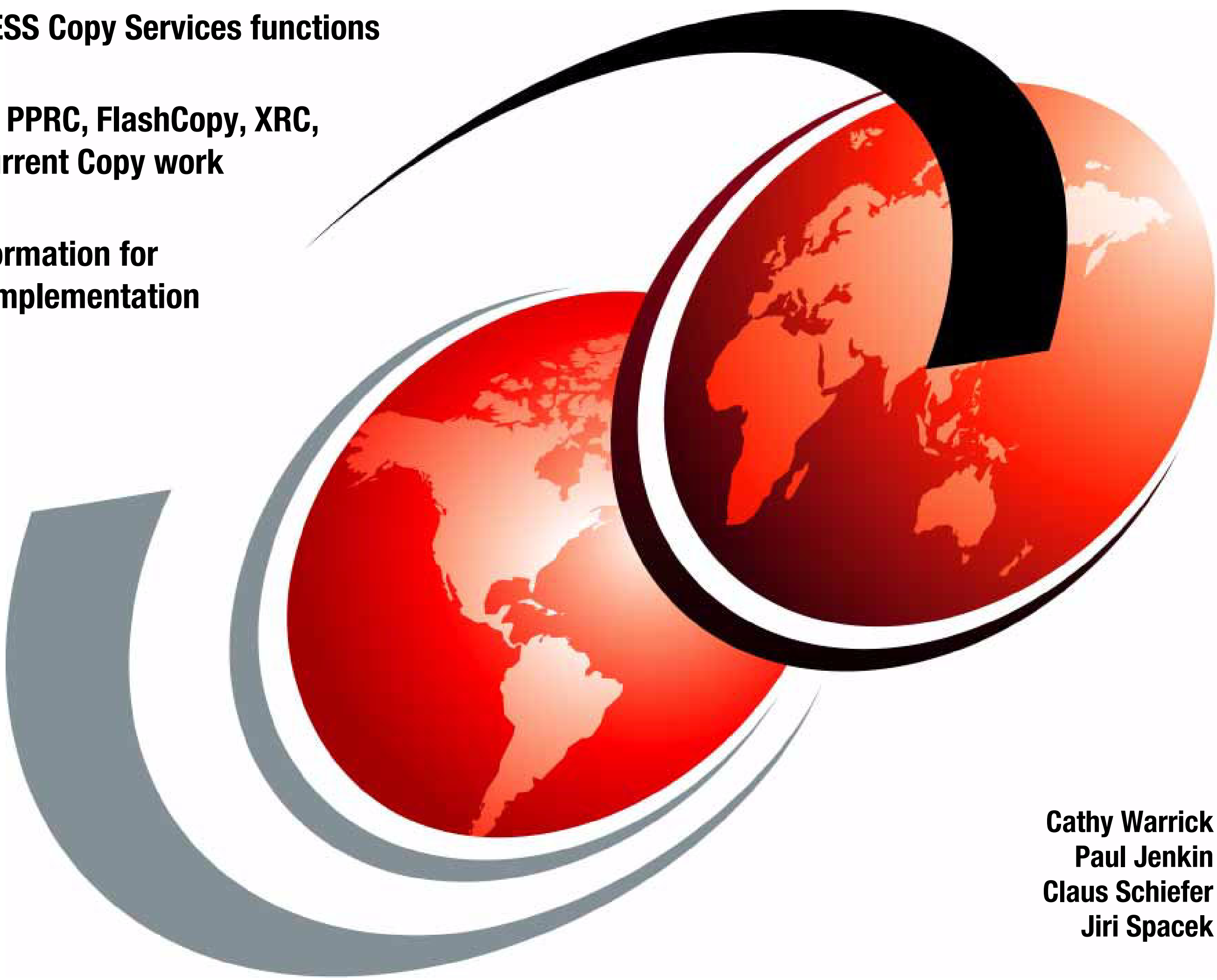
# IBM TotalStorage Enterprise Storage Server:

# Implementing ESS Copy Services with IBM eServer zSeries

**Understand the characteristics of the powerful ESS Copy Services functions**

**Learn how PPRC, FlashCopy, XRC, and Concurrent Copy work**

**Useful information for effective implementation**




Cathy Warrick
Paul Jenkin
Claus Schiefer
Jiri Spacek



Redbooks

ibm.com/redbooks

International Technical Support Organization

# IBM TotalStorage Enterprise Storage Server: Implementing ESS Copy Services with IBM eServer zSeries

July 2004

SG24-5680-04

**Note:** Before using this information and the product it supports, read the information in “Notices” on page xxi.

**Fifth Edition (July 2004)**

This edition applies to the IBM TotalStorage Enterprise Storage Server Model 800 and 750, as well as the previous F models.

# Contents



# Figures



# Tables

# Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:
*IBM Director of Licensing, IBM Corporation, North Castle Drive Armonk, NY 10504-1785 U.S.A.*

*The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law*: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:
This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

# Trademarks

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

- AIX®
- AIX 5L™
- CICS®
- CUA®
- DB2®
- DFS™
- DFSMS/MVS®
- DFSMSdfp™
- DFSMSdss™
- DFSMShsm™
- DFSMSrmm™
- Enterprise Storage Server®
- ECKD™
- ESCON®
- eServer™
- eserver®
- FlashCopy®
- FICON®
- GDPS®
- Geographically Dispersed Parallel Sysplex™
- Hiperbatch™
- HyperSwap™
- HACMP™
- IBM®
- ibm.com®
- IMS™
- IMS/ESA®
- iSeries™
- MVS™
- NetView®
- OS/390®
- Parallel Sysplex®
- pSeries®
- Redbooks™
- Redbooks (logo) ™
- RACF®
- RAMAC®
- RMF™
- S/390®
- Seascape®
- Sequent®
- Sysplex Timer®
- Tivoli Enterprise™
- Tivoli®
- TotalStorage®
- VM/ESA®
- VSE/ESA™
- VTAM®
- Wave®
- WebSphere®
- xSeries®
- zSeries®
- z/OS®
- z/VM®

The following terms are trademarks of other companies:

Intel, Intel Inside (logos), and Pentium are trademarks of Intel Corporation in the United States, other countries, or both.

Microsoft, Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, and service names may be trademarks or service marks of others.

# Preface

This IBM® Redbook describes the copy functions available with the IBM TotalStorage Enterprise Storage Server (ESS). The powerful ESS Copy Services functions are explained in detail, and their characteristics are thoroughly covered. This redbook also provides information on how to manage the various ESS Copy Services functions, and discusses their implementation.

This redbook applies to the ESS Model 800 and the ESS Model 750, as well as the previous F models when used in the IBM @server® zSeries® environments. Because this book provides a broad understanding of the ESS Copy Services functions, as well as presenting details about the management interfaces and the implementation considerations, it is a recommended manual for IT professionals who are planning the implementation of any of the ESS Copy Services functions in a zSeries environment, as well as for those who will manage these environments.

This fifth edition of the redbook has been updated with the latest ESS Copy Services functions available with LIC level 2.4.0:

- Brief overview of the ESS Model 750
- ESS LIC 2.4.0 additional functions:
  - Asynchronous PPRC, also known as IBM TotalStorage® Global Mirror
  - FlashCopy® target can now be a PPRC primary
- RMF™ performance measurement for Fibre Channel links
- Minor additions for ESS API support, Multiple Device Manager Replication Manager, Asynchronous PPRC Utilities for Open System Environment, and Asynchronous PPRC Utilities for ICKDSF Users

Also, some Web User Interface panels have changed with the introduction of LIC 2.4.0 and these changes are presented in this edition of the redbook.

## The team that wrote this redbook

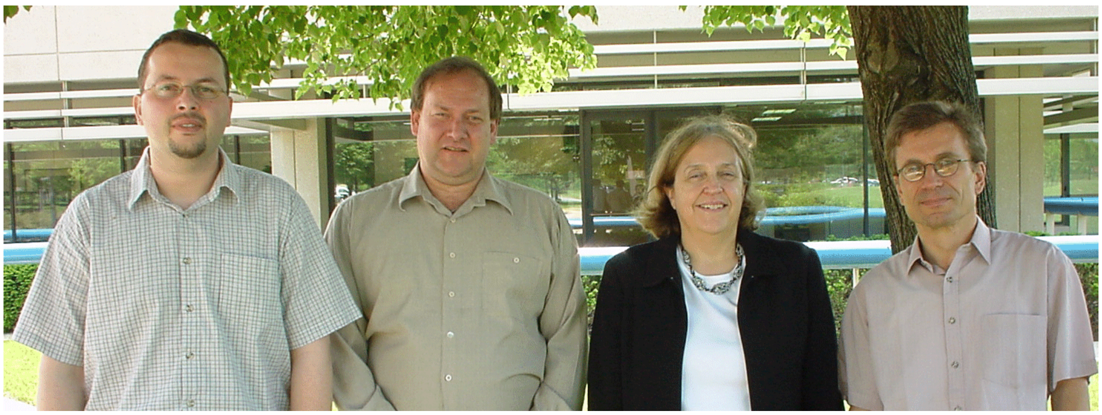
This redbook was produced by a team of specialists from around the world working at the International Technical Support Organization, San Jose Center.

**Cathy Warrick** is a Project Leader and Certified IT Specialist in the IBM International Technical Support Organization. She has over 25 years of experience in IBM with large systems, open systems, and storage, including education on products internally and for the field. Prior to joining the ITSO she developed the Technical Leadership education program for the IBM and Business Partner's technical field force.

**Paul Jenkin** has been with IBM for 23 years starting as a hardware field engineer and later a hardware specialist. He has worked on most of IBM's products over that period from PCs to zSeries, including most of IBM's DASD and storage products. Paul holds an NZ CIT Diploma in Computer Engineering. He has managed large and complex customer sites, including working alongside customers to manage and teach complex system recovery. In his current role as ITS Technical Team Leader for New Zealand, Paul provides leadership and specialist support to ITS Service Representatives across all platforms in both service and implementation and also provides backup support to Australia. Specialist areas include zSeries, all storage products disk and tape, SAN, networking, and complex problem management.

**Claus Schiefer** is an IT Specialist working for the Field Technical Sales Support in Germany. He joined IBM in 1991 and was a Systems Engineer for MVS™ and communication systems during his first three years. He has 10 years of storage experience in the disk and tape storage systems field with a focus on z/OS®. His areas of expertise include high end disk storage systems with PPRC, FlashCopy, XRC, and high end tape storage systems.

**Jiri Spacek** is an IT Consultant at GC System a.s., an IBM Business Partner in the Czech Republic. He has seven years of experience in the IT field. His areas of expertise include IBM pSeries® and AIX®, HACMP™, and disk storage subsystems. Currently he is an IBM Certified Advanced Technical Expert for pSeries and AIX 5L™ and an IBM Certified Specialist for Enterprise Disk Solutions.



*Jiri, Paul, Cathy, and Claus*

The authors of previous editions of this book are:

| | | | |
|---|---|---|---|
| Mark Blunden | Sverre Bergum | Jose Dovidauskas | Clemente Vaia |
| Heinz Blascheck | Tony Cresswell | Tim Starr | Andre Fourie |
| Ying Thia | Robert Tondini | Gustavo Castets | Donald (Chuck) Laing |
| George Kozakos | | | |

Special thanks go to **Rosemary McCutchen** and **Craig Gordon** from IBM Advanced Technical Support, ATS Gaithersburg, for their support with the setup of the environment we used, as well as their invaluable feedback when testing the new functions and discussing with us the technical considerations for our redbook.

Thanks to the following people for their contributions to this project:

John Amann
Bob Bartfai
Werner Bauer
Charlie Burger
Shelley Carton
Steve Chase
Matthias Gubitz
Joe Hyde
Emma Jacobs
Lloyd Johnson
Frank Krueger
Yvonne Lyon
Alison Pate
Deanna Polm
Don Resnick
Bill Robinson

Judy Ruby-Brown
Brian Sherman
Dean Smith
Gail Spear
Warren Stanley
William Supon
Tom Taleck
John Thompson
Jim Tuckwell
Ricardo Urbanejo
Olga Yiparaki

## Become a published author

Join us for a two- to six-week residency program! Help write an IBM Redbook dealing with specific products or solutions, while getting hands-on experience with leading-edge technologies. You'll team with IBM technical professionals, Business Partners, and/or customers.

Your efforts will help increase product acceptance and customer satisfaction. As a bonus, you'll develop a network of contacts in IBM development labs, and increase your productivity and marketability.

Find out more about the residency program, browse the residency index, and apply online at:

**ibm.com**/redbooks/residencies.html

## Comments welcome

Your comments are important to us!

We want our Redbooks™ to be as helpful as possible. Send us your comments about this or other Redbooks in one of the following ways:

- Use the online **Contact us** review redbook form found at:

  **ibm.com**/redbooks

- Send your comments in an Internet note to:

  redbook@us.ibm.com

- Mail your comments to:

  IBM Corporation, International Technical Support Organization
  Dept. QXXE Building 80-E2
  650 Harry Road
  San Jose, California 95120-6099

# Summary of changes

This section describes the technical changes made in this edition of the book and in previous editions. This edition may also include minor corrections and editorial changes that are not identified.

Summary of Changes
for SG24-5680-04
for IBM TotalStorage Enterprise Storage Server: Implementing ESS Copy Services with IBM eServer zSeries
as created or updated on August 4, 2004.

## May 2004, Fifth Edition

This revision reflects the addition, deletion, or modification of new and changed information described below.

### New information

- Brief overview of the ESS Model 750
- ESS LIC 2.4.0 additional functions:
  - Asynchronous PPRC
  - FlashCopy target can now be a PPRC primary
- RMF performance measurement of Fibre Channel links
- Minor additions for ESS API support, Multiple Device Manager Replication Manager, Asynchronous PPRC Utilities for Open System Environment, and Asynchronous PPRC Utilities for ICKDSF Users

### Changed information

- Additional scenarios using Asynchronous PPRC
- Updated Web User Interface screens, CLI commands
- Reorganized some chapters for ease of use

1

# Introduction

This introductory chapter reviews the topics covered in this book and introduces the IBM TotalStorage Resiliency Family of products, which includes the IBM TotalStorage Enterprise Storage Server (ESS) and ESS Copy Services, to be discussed in depth in this book. The features of ESS Copy Services are summarized and the key components described.

Also introduced are the models of the IBM TotalStorage Enterprise Storage Server (ESS), including the model 800 and the new model 750. This is followed by a discussion on the strategies for Disaster Recovery, showing where the IBM TotalStorage Enterprise Storage Server (ESS) advanced copy functions are positioned.

When planning your disaster recovery solutions, refer to the following redbook for further detailed discussion: *IBM TotalStorage Solutions for Disaster Recovery,* SG24-6547.

## 1.1 IBM TotalStorage Resiliency Family

The IBM Total Storage Resiliency Family comprises the IBM TotalStorage Resiliency Core Technology, an extensive set of hardware and software features and products, IBM TotalStorage Resiliency Automation, and various integrated software and services packages. The IBM TotalStorage Resiliency Family is designed to help you implement storage infrastructures to help keep your business running 24 hours a day, 7 days a week.

The following ESS Model 750 and 800 functions are key components of the IBM TotalStorage Resiliency Core Technology:

- FlashCopy, also known as IBM TotalStorage FlashCopy
- PPRC:
  - Synchronous PPRC, also known as IBM TotalStorage Metro Mirror
  - Asynchronous PPRC, also known as IBM TotalStorage Global Mirror
  - Asynchronous Cascading PPRC, also known as IBM TotalStorage Metro/Global Copy
  - PPRC Extended Distance, also known as IBM TotalStorage Global Copy
- XRC (Extended Remote Copy) - *Model 800 only*
  - XRC, also known as IBM TotalStorage z/OS Global Mirror
  - Three-site solution using Synchronous PPRC and XRC, also known as IBM TotalStorage z/OS Metro/Global Mirror

## 1.2 ESS Family of Enterprise Storage Servers

The ESS Family of Enterprise Storage Servers was expanded in April 2004 with the announcement of the ESS Model 750. The family now includes the ESS Model 800, ESS Model 800 with Turbo option, and the new ESS Model 750. Figure 1-1 provides general descriptions of each model and their design points.

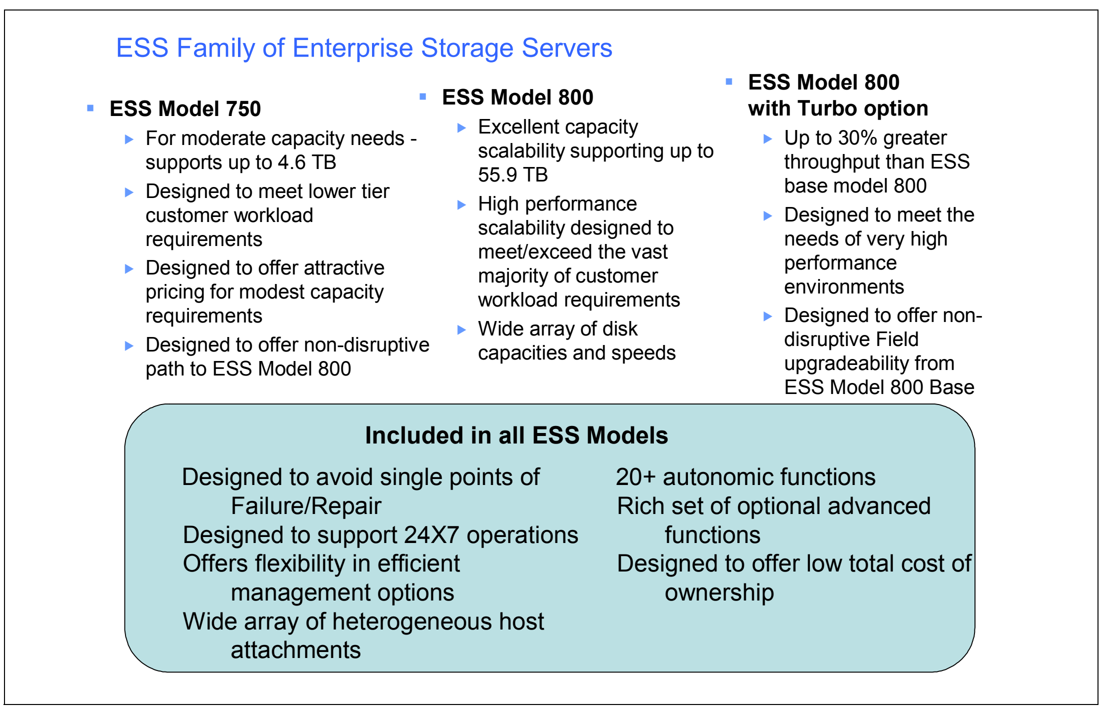



*Figure 1-1 ESS Family*

## 1.3 IBM TotalStorage Enterprise Storage Server Model 800

The IBM TotalStorage Enterprise Storage Server® (ESS) is IBM's most powerful disk storage server, developed using IBM Seascape® architecture. The ESS provides unmatchable functions for all the IBM @server family of e-business servers, and also for the non-IBM (that is, Intel®-based and UNIX®-based) families of servers. Across all of these environments, the ESS features unique capabilities that allow it to meet the most demanding requirements of performance, capacity, and data availability that the computing business may require.

The Seascape architecture is the key to the development of IBM's storage products. Seascape allows IBM to take the best of the technologies developed by the many IBM laboratories and integrate them, producing flexible and upgradeable storage solutions. This Seascape architecture design has allowed the IBM TotalStorage Enterprise Storage Server to evolve from the initial E models to the succeeding F models, to the current 750 and 800 models, each featuring new, more powerful hardware and functional enhancements, and always integrated under the same successful architecture with which the ESS was originally conceived.

The move to on demand business presents companies with both extraordinary opportunities and significant challenges. Consequently, companies also face an increase in critical requirements for more information that is universally available online, around the clock, every day of the year.

To meet the requirements of an on demand business, where massive swings in the demands placed on your systems are common, and continuous operation is imperative, you need very high-performance and intelligent storage technologies and systems, which can support any server application in your business, today and into the future. The IBM TotalStorage Enterprise Storage Server has set new standards in function, performance, and scalability in these most challenging environments.

### 1.3.1 ESS Model 800 highlights

Since its initial availability with the ESS Models E10 and E20, and then with the succeeding F10 and F20 models, the ESS has been the storage server solution offering exceptional performance, extraordinary capacity, scalability, heterogeneous server connectivity, and an extensive suite of advanced functions to support users' mission-critical, high-availability, multi-platform environments. The ESS set a new standard for storage servers back in 1999 when it was first available, and since then it has evolved into the F models and the more recent third-generation ESS Model 800, keeping up with the pace of users' needs by adding more sophisticated functions to the initial set, enhancing the connectivity options, and powering its performance features.

The IBM TotalStorage Enterprise Storage Server Model 800 provides significantly improved levels of performance, throughput, and scalability while continuing to exploit the innovative features introduced with its preceding E and F models such as Parallel Access Volumes, Multiple Allegiance, I/O Priority Queuing, the remote copy functions (synchronous, non-synchronous and asynchronous), and the FlashCopy point-in-time copy function. Also, the heterogeneous server support characteristics (for connectivity and remote copy functions) of previous models are continued with the ESS Model 800.

With the IBM TotalStorage Enterprise Storage Server Model 800 important changes have been introduced, which dramatically improved the overall value of the ESS in the marketplace and provide a strong base for strategic Storage Area Network (SAN) initiatives.

Figure 1-2 shows an overview of the IBM Seascape architecture for the ESS Model 800.

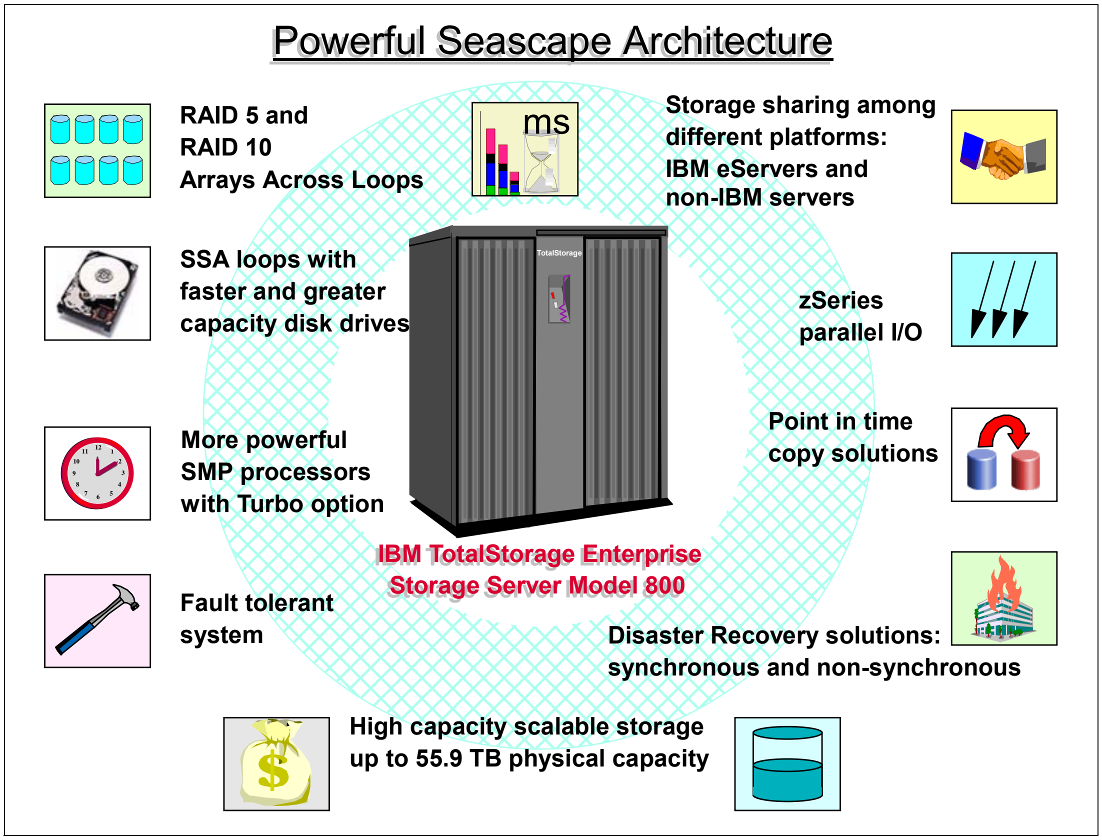



*Figure 1-2 IBM Seascape architecture - ESS Model 800*

## Third-generation hardware: ESS Model 800

The IBM TotalStorage Enterprise Storage Server Model 800 integrates a new generation of hardware from top to bottom, allowing it to deliver unprecedented levels of performance and throughput. Key features that characterize the performance enhancements of the ESS Model 800 are as follows:

- The ESS Model 800 is capable of delivering twice the throughput of its predecessor model F20. Initially, with the optional Turbo feature (feature #3606), the ESS Model 800 was capable of providing 2.5 times the throughput of the model F20, thus allowing increased scalability and better response times. Now, the Turbo II Processor option (feature #3607), a follow-on to the Turbo Processor (feature #3606), offers up to a 30% improvement for high-hit ratio and high operations per second workloads as compared to the standard processor in an ESS Model 800. Processor option feature conversions are available to enable existing Model 800 machines to exploit this new enhancement.
- 64 GB cache supports much larger system configurations and increases cache hit ratios, driving down response times.
- Double the internal bandwidth provides high sequential throughput for digital media, business intelligence, data warehousing, and life science applications.
- Larger 2 GB NVS with twice the bandwidth allows greater scalability for write-intensive applications.
- Third-generation hardware provides response time improvements of up to 40% for important database applications.
- 2 Gb Fibre Channel/FICON® host adapters provide doubled performance sustained and instantaneous throughput for both open systems and zSeries environments.

- 64-bit ESCON® host adapters, enhanced with a faster microprocessor providing increased channel throughput and sequential read bandwidth.
- RAID-10 can provide up to 75% greater throughput for selected database workloads compared to equal physical capacity configured as RAID-5. While most typical workloads will experience excellent response times with RAID-5, some cache-unfriendly applications and some applications with high random write content can benefit from the performance offered by RAID-10.
- 15,000 rpm drives provide up to 80% greater throughput per RAID rank and 40% improved response time as compared to 10,000 rpm drives. This allows driving the workloads to significantly higher access densities, while also experiencing improved response times. Starting with LIC level 2.3.0., the ESS also offers rpm intermix support among same-capacity disks.
- Starting with LIC level 2.3.0, the ESS Model 800 can be configured with Arrays Across Loops (AAL), feature number 9903. AAL is a new configuration option whereby disk arrays are spread across two loops on the SSA device adapter pair. AAL allows you to take full advantage of the ESS subsystem bandwidth because a RAID array is built across two SSA loops within a device adapter pair. Arrays Across Loops provides up to 1.8 times improvement in single array sequential bandwidth performance.

All of this performance boost is built upon the reliable and proven ESS hardware architecture design and unique advanced features.

# 1.4 IBM TotalStorage Enterprise Storage Server Model 750

The ESS Model 750 is well-suited for clients with midrange capacity and performance needs all in an affordable package. It includes many of the functions of the ESS Model 800 and all of its reliability. It is designed to meet the high availability requirements of mainframe and open systems environments and is an especially good fit for the IBM zSeries 8xx servers.

The ESS Model 750 consists of two clusters, *each* with a two-way processor and 4 or 8 GB cache. It can have two to six Fiber Channel/FICON or ESCON host adapters. The minimum storage capacity is 1.1 TB up to a maximum storage capacity of 4 TB. A key feature is that the ESS 750 is upgradeable, non-disruptively to the ESS Model 800 which can grow to more than 55 TB of physical capacity.

## 1.4.1 Features of the ESS Model 750

Here are some more details about the features of the ESS Model 750:

- It attaches to a wide-variety of hosts, including all of the @server® platforms, zSeries®, pSeries®, iSeries™, and xSeries®. It will also attach to a large number of different UNIX, Linux, and Intel processor-based servers. To find the most current list of supported servers, go to:

  http://www.storage.ibm.com/disk/ess/pdf/interop.pdf
- It has redundant, failover hardware.
- Both 72.8 (10K rpm) and 145.6 GB (10K rpm) drives are available. Each pair of 8 packs must have the same capacity and speed. Intermixing is allowed across the loops.

  **Note:** Arrays across loops (AAL) is not supported on the ESS Model 750.
- RAID-5 and RAID-10 are available and intermixable.

- It has 2 GB NVS and 8 or 16 GB cache.
- It has 64 non-arbitrated, pipelined paths to disks.
- It supports zSeries performance enablers - PAV, Priority I/O Queueing, and Multiple Allegiance.
- It comes with LIC 2.4.0 standard at General Availability (GA).
- Advanced function support is available for FlashCopy V1 and V2, PPRC V1 and V2.

> **Note:** XRC (Extended Remote Copy) is not supported on the ESS Model 750.

### 1.4.2 Compatibility

The ESS Model 750 is compatible and can co-exist with the other models of the ESS for:

- Copy Services Domain
- PPRC and FlashCopy Consistency Groups
- PPRC relationships
- Managed under the ESS Master Console

The ESS Model 750 meets the needs of our clients for high availability, entry capacity and performance at a low price. It comes with a 3 year warranty and gives the client investment protection as it can be upgraded, non-disruptively, to the ESS Model 800.

## 1.5 ESS Model 750 and 800 hardware features

In the following sections we describe the hardware features of these models.

#### Efficient cache management and powerful back-end

The ESS is designed to provide the highest performance, for the different type of workloads, even when mixing dissimilar workload demands. For example, zSeries servers and open systems put very different workload demands on the storage subsystem. A server like the zSeries typically has an I/O profile that is very cache-friendly, and takes advantage of the cache efficiency. On the other hand, an open system server does I/O that can be very cache-unfriendly, because most of the hits are solved in the host server buffers. For the zSeries type of workload, the ESS Model 800 has the option of a large cache (up to 64 GB) and most importantly it has efficient cache algorithms. For the cache unfriendly workloads, the ESS has a powerful back-end, with the SSA high performance disk adapters providing high I/O parallelism and throughput for the ever-evolving high-performance hard disk drives.

#### Sysplex I/O management

In the zSeries Parallel Sysplex® environments, the z/OS Workload Manager (WLM) controls where work is run and optimizes the throughput and performance of the total system. The ESS provides the WLM with more sophisticated ways to control the I/O across the sysplex. These functions include parallel access to both single-system and shared volumes, and the ability to prioritize the I/O based upon WLM goals. The combination of these features significantly improves performance in a wide variety of workload environments.

#### Parallel Access Volume (PAV) and Multiple Allegiance

Parallel Access Volume and Multiple Allegiance are two distinctive performance features of the IBM TotalStorage Enterprise Storage Server for the zSeries users, allowing them to reduce device queue delays, which means improving throughput and response time.

### I/O load balancing

For selected open system servers, the ESS in conjunction with the Subsystem Device Driver (SDD), a pseudo device driver designed to support multi-path configurations, provides dynamic load balancing. Dynamic load balancing helps eliminate data-flow bottlenecks by distributing the I/O workload over multiple active paths, thus contributing to improved I/O throughput and response time of the open system server.

### 2 Gb Fibre Channel/FICON host adapters

As the amount of data and transactions grow, so does the traffic over the Storage Area Networks (SAN). As SANs migrate to 2 Gb technologies to cope with this increased amount of data transit, so does the IBM TotalStorage Enterprise Storage Server Model 800 and Model 750 with their 2 Gb host adapters. These host adapters double the bandwidth of the previous adapters, thus providing more throughput and performance for retrieving and storing users' data.

### 64-bit ESCON host adapters

The ESCON adapters have been enhanced with a faster microprocessor that offers up to a 45% improvement in full box sequential read bandwidth and up to 10% increase in channel throughput for random operation workloads as compared to the previous 32-bit ESCON host adapters. When used in a PPRC configuration on both the primary and secondary ESS, up to a 10% increase in PPRC link throughput for random write operations and sequential bandwidth may be achieved as compared to the previous 32-bit ESCON host adapters.

### Arrays Across Loops (AAL)

AAL is a new configuration option whereby disk arrays are spread across two loops on the SSA device adapter pair. AAL allows you to take full advantage of the ESS Model 800 subsystem bandwidth because a RAID array is built across two SSA loops within a device adapter pair. Each SSA loop provides up to 160 MB per second non-arbitrated bandwidth for data transfer. Providing access to two loops almost doubles the bandwidth capacity available to a RAID rank. z/OS logical volumes or open systems LUNs are striped across the data volumes of their associated RAID rank.

Implementing AAL allows full use of this bandwidth over twice as many paths to the disks, thus creating the possibility for multiple processes to access a busy volume or LUN with high demand. This can help improve performance and reduce elapsed times, especially for sequential batch and file maintenance. Overall, an AAL configuration can provide up to a 1.8 times improvement in sequential bandwidth performance on a single array, compared to non-AAL configurations.

AAL is supported on the ESS Model 800, and is an optional feature to the ESS (#9903).

## 1.6 Disaster Recovery

The prime purpose for backing up, copying and/or mirroring data is to be prepared for a possible disaster. Every business and every application has different requirements for the level of recovery needed to protect the data and the business needs. In some cases, not only must the data be protected but facilities and equipment are set up to be able to restart critical applications at a remote site. Many of these applications cannot tolerate loss of data or they require that the data written to its volumes is consistent, the last writes were all at the same point-in-time. In other cases a business can accept the risk and loss of a few hours or days, by rolling back to a recoverable consistent point.

In each situation the solution chosen will depend on the resources available and the cost to implement the solution balanced against the business risk.

In Chapter 8, “Solutions” on page 409 we discuss Disaster Recovery and give some examples of common implementations. Also, in 3.2, “Choosing the best method” on page 75, guidance is provided to assist in choosing the best features of ESS Copy Services that balance your business needs against implementation costs and complexity.

## 1.7 Data protection and availability

Many design characteristics and advanced functions of the IBM TotalStorage Enterprise Storage Server Model 800 contribute to protect the data in an effective manner.

### 1.7.1 Fault-tolerant design

The IBM TotalStorage Enterprise Storage Server is designed with no single point of failure. It is a fault-tolerant storage subsystem, which can be maintained and upgraded concurrently with user operation. Some of the features that contribute to this characteristic of the ESS are shown in Figure 1-3.

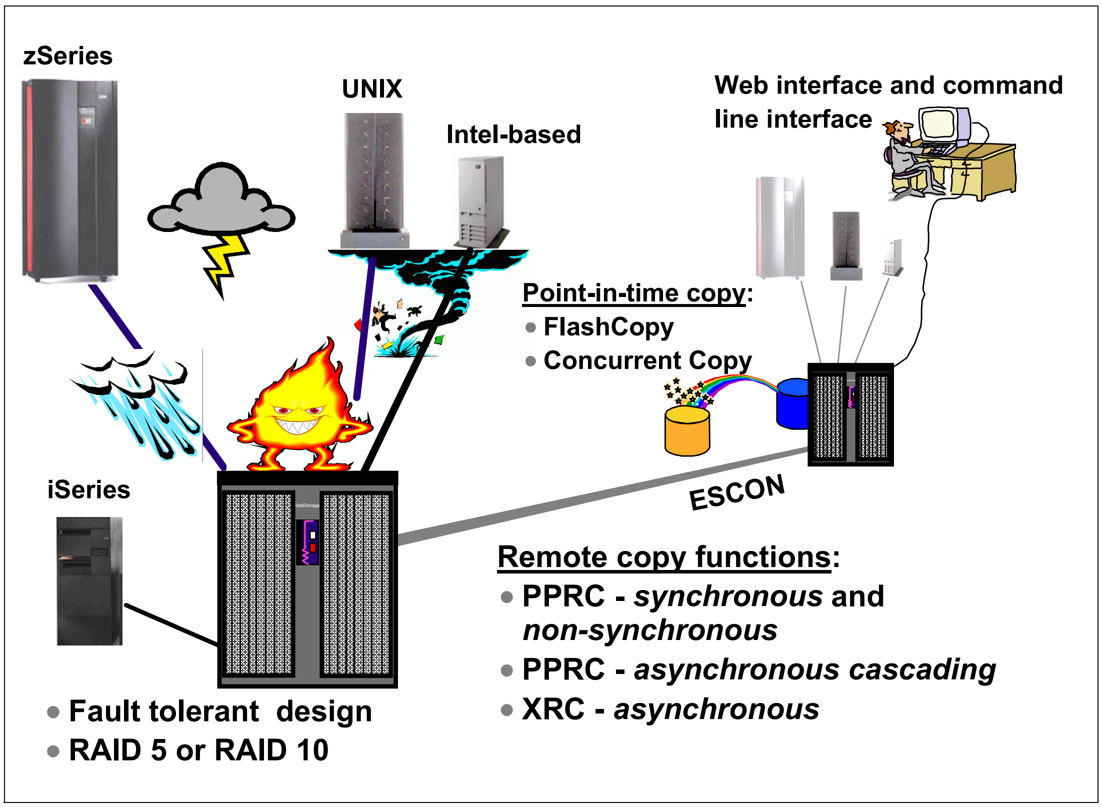



*Figure 1-3 Data protection and availability*

### 1.7.2 RAID-5 or RAID-10 data protection

With the IBM TotalStorage Enterprise Storage Server Model 800, the disk arrays can be configured in a RAID-10 (mirroring plus striping) or a RAID-5 (striping with distributed parity) arrangement, thus giving more options when selecting the redundancy technique for protecting data.

### 1.7.3 FlashCopy Version 1

FlashCopy is designed to provide a point-in-time copy capability for logical volumes. FlashCopy creates a physical point-in-time copy of the data, with minimal interruption to applications, and makes it possible to access both the source and target copies immediately. FlashCopy Version 1 is an optional feature on the ESS.

FlashCopy V1 is discussed in 2.1, “FlashCopy Version 1” on page 14.

### 1.7.4 FlashCopy Version 2

FlashCopy Version 2 delivers new FlashCopy functions and enhancements designed to help improve business efficiency, along with FlashCopy performance improvements designed to help minimize operational disruption. FlashCopy Version 2 is an optional feature on the ESS.

FlashCopy Version 2 includes support for all previous FlashCopy functions, plus these:

- Data set FlashCopy, providing a new level of granularity for the zSeries environments
- Multiple Relationship FlashCopy, allowing a source to have multiple targets
- Incremental FlashCopy, providing the capability to “refresh” a FlashCopy relationship
- Elimination of the LSS constraint: a source and target relationship can span logical subsystems (LSS)
- Establish time improvement, designed to provide up to a 10 times reduction

FlashCopy Version 2 is discussed in 2.2, “FlashCopy Version 2” on page 31.

### 1.7.5 Peer-to-Peer Remote Copy (PPRC) Version 1

A hardware-based disaster recovery solution designed to provide real-time mirroring of logical volumes within an ESS or between two distant ESSs. PPRC has two basic modes of operation: synchronous and non-synchronous.

The Synchronous PPRC implementation (PPRC-SYNC) is a synchronous remote copy solution where write operations are completed on both copies (primary and secondary ESSs) before they are considered to be done. Thus, the recovery data at the remote site will be a constant real time mirror of the data at the local site as the applications do their updates. PPRC Version 1 is an optional feature on the ESS.

PPRC is discussed in Chapter 3, “Peer-to-Peer Remote Copy” on page 69.

### 1.7.6 PPRC Version 2

PPRC Version 2 provides new options for long-distance remote copy solutions:

- **PPRC over Fibre Channel links:** Fibre Channel Protocol (FCP) can be used as the communications link between PPRC primary ESSs and secondary ESSs. FCP reduces the link infrastructure by at least 4 to 1 when compared to ESCON, and relieves logical and physical path constraints. The supported distance for Synchronous PPRC has been increased to 300 km (over FCP) but remains at 103 km for ESCON.
- **Asynchronous Cascading PPRC:** Asynchronous Cascading PPRC provides a long-distance remote copy solution for zSeries and open systems environments by allowing a PPRC secondary volume (involved in a PPRC synchronous relationship) to also simultaneously serve as a PPRC primary volume in a PPRC Extended Distance (PPRC-XD) relationship to the remote site. This new capability enables the creation of three-site or two-site Asynchronous Cascading PPRC configurations.

- **Failover and Failback modes for Asynchronous Cascading PPRC:** This is supported by the ESS Copy Services Web User Interface (WUI) and the ESS Copy Services Command Line Interface (CLI) on supported platforms, for both open systems and zSeries environments. For the zSeries environments, the ICKDSF utility can also be used to manage this function.
- **Asynchronous PPRC:** This is designed to provide a long-distance remote copy solution across two sites using asynchronous technology. It operates over high-speed, Fibre Channel communication links and is designed to provide a consistent and restartable copy of the data at the remote site, created with minimal impact to applications at the local site. Compared to Asynchronous Cascading PPRC, Asynchronous PPRC eliminates the requirement to do a manual and periodic suspend at the local site order to create a consistent and restartable copy at the remote site.

PPRC Version 2 is an optional feature on the ESS and is discussed in Chapter 3, “Peer-to-Peer Remote Copy” on page 69.

### 1.7.7 PPRC Extended Distance (PPRC-XD)

PPRC-XD offers a non-synchronous long-distance copy option whereby write operations to the primary ESS are considered complete before they are transmitted to the secondary ESS. This non-synchronous operation results in a “fuzzy copy” at the secondary site; however, through operational procedures, a point-in-time consistent copy at the remote site can be created that is suitable for data migration, backup, and disaster recovery purposes.

PPRC-XD can operate at very long distances (distances well beyond the 103 km supported with PPRC synchronous transmissions over ESCON) with the distance typically limited only by the capabilities of the network and channel extension technologies. PPRC-XD support is included at no additional charge when PPRC is purchased for the ESS Model 800.

PPRC-XD is discussed in 3.7, “PPRC-XD” on page 94.

### 1.7.8 Extended Remote Copy (XRC)

XRC is a combined hardware and software business continuance solution for the zSeries and S/390® environments providing asynchronous mirroring between two ESSs at global distances. XRC is an optional feature on the ESS.

XRC is discussed in Chapter 5, “Extended Remote Copy” on page 207.

### 1.7.9 Concurrent Copy

Concurrent Copy offers another method for creating a point-in-time copy in zSeries and S/390 environments, with the source data fully available for access and update after initiation of the copy operation.

Concurrent Copy is discussed in Chapter 6, “Concurrent Copy” on page 273.

## 1.8 Combinations of copy services

The ESS provides support for the combination of various copy functions. These include FlashCopy, Concurrent Copy, XRC, and the various forms of PPRC. Some of the different ESS Copy Services functions can be combined, and Figure 1-4 shows the valid and non-valid combinations.

| DEVICE IS / MAY BECOME | XRC primary | XRC secondary | PPRC primary | PPRC secondary | FlashCopy source | FlashCopy target | ConCopy source |
|---|---|---|---|---|---|---|---|
| XRC primary | No | Yes | Yes | No | Yes | No | Yes |
| XRC secondary | Yes | No | Yes | No | Yes | No | Yes |
| PPRC primary | Yes | Yes | No | Yes (1) | Yes | Yes | Yes |
| PPRC secondary | No | No | Yes (1) | No | Yes | Yes (2) | No |
| FlashCopy source | Yes | Yes | Yes | Yes | Yes (3) (4) | Yes (4) | Yes |
| FlashCopy target | No | No | Yes | No | Yes (4) | Yes (4) | No |
| ConCopy source | Yes | Yes | Yes | No | Yes | Yes | Yes |

*Figure 1-4 Allowable combinations of copy operations*

Figure 1-4 notes:

1. Valid combination for Asynchronous Cascading PPRC only (available with PPRC V2).
2. The operation is allowed. Updates of the affected extents will result in implicit removal of the FlashCopy relationship.
3. Valid combination for Multiple Relationship FlashCopy (available with FlashCopy V2).
4. Valid combination for Data set FlashCopy (available with FlashCopy V2).

## 1.9 Terminology

Copy Services functions are ESS implementations of the functions described in the *IBM Total Storage Resiliency Family Core Technologies*.

2

# FlashCopy

In this chapter we describe the characteristics and operation of the ESS Copy Services FlashCopy Version 1 and Version 2 function, also known as IBM TotalStorage FlashCopy. We also discuss considerations that should be regarded when planning to implement FlashCopy.

For more details, refer to the following publications, which complement the information presented in this chapter:

- *IBM TotalStorage Solutions for Disaster Recovery,* SG24-6547
- *z/OS DFSMS Advanced Copy Services,* SC35-0428
- *z/OS DFSMSdss Storage Administration Guide*, SC35-0423
- *IBM TotalStorage Enterprise Storage Server Web Interface User's Guide,* SC26-7448
- *Device Support Facilities User's Guide and Reference Release 17*, GC35-0033
- *IBM TotalStorage Enterprise Storage Server User's Guide*, SC26-7445

# 2.1 FlashCopy Version 1

In this section we discuss the features offered by FlashCopy Version 1.

## 2.1.1 Overview

Today, data processing centers increasingly require applications to be available 24 hours per day, seven days per week, without compromising recoverability in the event of a failure. FlashCopy allows the user to implement such solutions.

FlashCopy provides a point-in-time copy of a logical volume, also called a time-zero copy, with almost instant availability for the application of both the source and target volumes. Only a minimal interruption is required for the FlashCopy relationship to be established, so the copy operation can be initiated. The copy is then created “under the covers” by the IBM TotalStorage Enterprise Storage Server (ESS), with minimal impact on other ESS activities.

FlashCopy is an optional feature that must be enabled in the ESS. See Appendix D, “ESS Copy Services feature codes” on page 383 for the corresponding feature codes.

FlashCopy may also be used in conjunction with either the local or remote copies of data created by PPRC and XRC, making it easy to create additional copies for rapid recovery following application errors or for other uses. For the allowable combinations of FlashCopy with the other ESS copy functions, refer to the table in Figure 1-4 on page 11.

## 2.1.2 How FlashCopy works

FlashCopy V1 is invoked at the volume level. The following considerations apply:

- The source and target volumes must have the same track format.
- The target volume must be at least as large as the source volume.
- The source and target volumes must be within the same ESS logical subsystem (LSS).
- A source and target volume can only be involved in one FlashCopy relationship at a time.

As soon as a FlashCopy establish command is issued (either invoked by a TSO command, the DFSMSdss™ utility, ICKDSF, the ANTRQST macro, or by means of the ESS Copy Services Web User Interface (WUI)) the ESS establishes a FlashCopy relationship between the target volume and the source volume. This relationship exists from the time you initiate a FlashCopy operation until the ESS copies all data from the source volume to the target volume.

You may optionally request FlashCopy not to execute the background copy; in this case the relationship must be specifically withdrawn in order to terminate it. A relationship must also be explicitly withdrawn if it was established with the *Persistent FlashCopy* option (discussed in “Persistent FlashCopy relationship” on page 16).

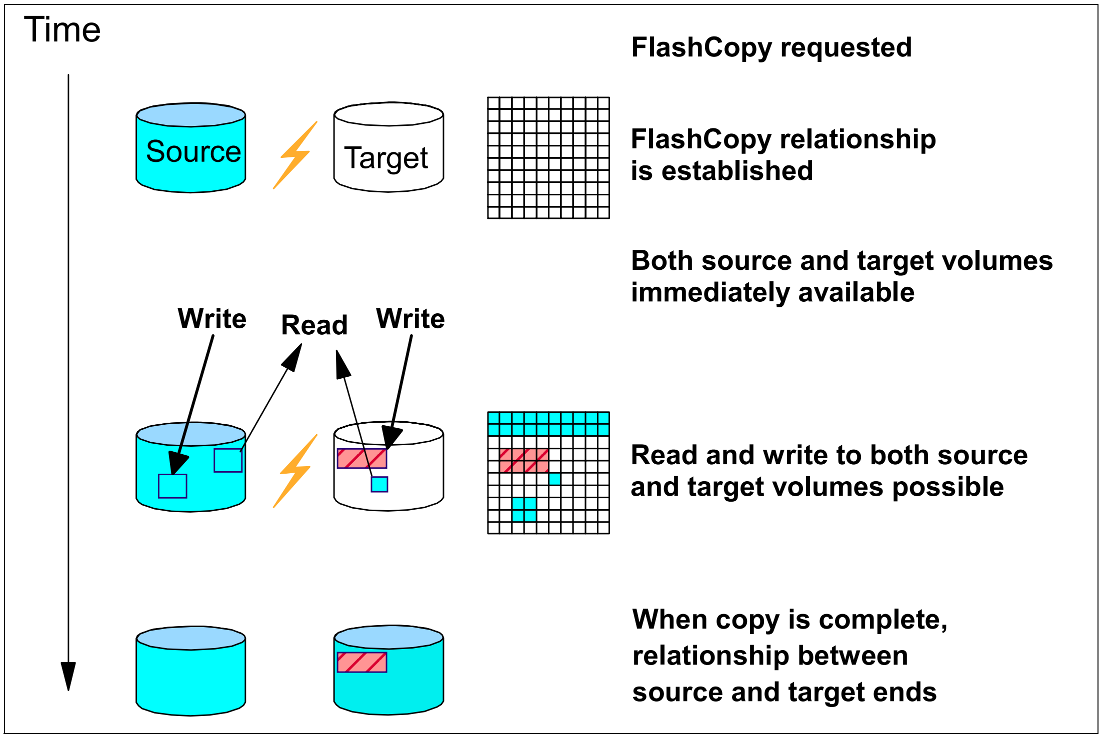



*Figure 2-1 FlashCopy with background copy*

As Figure 2-1 illustrates, there are basically three stages that a FlashCopy relationship goes through: Establishing the relationship, then copying the data, and finally terminating the relationship. Let us examine them in more detail.

## Establish phase

During the *establish* of the FlashCopy relationship, a metadata structure is created for this relationship. This metadata is used by the ESS microcode to map source and target volumes as they were at the time when the FlashCopy was requested (time-zero copy), as well as to manage subsequent reads and updates to the source and target volumes. Updates to the source volume *after* the FlashCopy relationship is established will not be reflected on the target device.

The establish process takes a minimum amount of time. As soon as the relationship is established, user programs have access to both the source and target copies of the data.

## Copy phase

With the relationship already established, and the source and target volumes already available for the applications to use them, the copy phase begins. How this copy phase is conducted depends on the copy option that is selected for this FlashCopy operation.

The FlashCopy relationship may be established either with or without *background copy*. FlashCopy will manage the copy process differently according to the option specified.

### *FlashCopy: Background copy (default)*

By default, FlashCopy will do a background copy. The background copy task does a physical copy of all tracks from the source volume to the target volume. De-staging algorithms are used to efficiently manage the under-the-covers ESS copy process. The background copy task runs at a lower priority than normal I/O on the ESS, not to impact the normal application I/O processing.

The ESS, using the metadata structure created during establish, keeps track of which data has been copied from the source to the target, and manages the integrity of both copies. If an application wants to read some data from the target that has not yet been copied, the data is read from the source; otherwise, the read can be satisfied from the target volume.

Before updating a not-yet-copied track on the source volume, the ESS does an *on-demand copy* of the track to the target volume, thus preserving the time-zero copy. Subsequent reads to this track on the target volume will be satisfied from the target volume. Before updating a not-yet-copied track on the target volume, the ESS will perform an on-demand copy of this track to the target volume. This on-demand activity is done asynchronously after the host has received I/O completion (once the data is written in cache and NVS), so host I/O is not delayed.

After some time, all tracks will have been copied to the target volume, and the FlashCopy relationship will automatically end unless the persistent FlashCopy option was specified (discussed in “Persistent FlashCopy relationship” on page 16).

#### *FlashCopy: No background copy*

When selecting *not* to do the background copy, the relationship is established but the background copy task — of all the source volume tracks — is *not* initiated. Only the source tracks which receive application updates will be copied to the target. Before updating a track on the source volume, the ESS does an *on-demand copy* of the track to the target volume, thus preserving the time-zero copy. Similarly, before updating a track on the target volume, the ESS will perform an on-demand copy of this track to the target volume.

A FlashCopy relationship that was established selecting no-background remains active until one of the following occurs:

- An explicit FlashCopy withdraw is done to terminate the relationship.
- All source device tracks have been copied on the target device because they were all updated.
- All target device tracks have been updated by user applications.

If the persistent FlashCopy option was selected, the FlashCopy relationship exists until it is specifically withdrawn, regardless of the copy status.

When a no-background copy FlashCopy relationship is terminated, the target volume is left in an indeterminate state. Some of the tracks on the volume may contain data from the source volume, while other tracks may contain residual data that was on the target volume before the copy. The volume should not be used in these conditions unless it is reformatted or used for another copy operation.

### Terminating the FlashCopy relationship

The FlashCopy relationship is *automatically ended* when all tracks have been copied from the source volume to the target volume. The relationship can also be *explicitly withdrawn* by issuing the command through TSO, ICKDSF, DFSMSdss utility, or by using the ESS Copy Services WUI.

If the persistent FlashCopy option was specified, the FlashCopy relationship must be withdrawn explicitly.

### Persistent FlashCopy relationship

When the *Persistent FlashCopy* option is selected, the FlashCopy relationship does not automatically end when the background physical copy ends. The FlashCopy relationship persists until explicitly withdrawn.

Persistent FlashCopy relationships can help protect against inadvertent updates of recently created target volumes. For example, if a source volume is regularly copied to alternating target volumes (thereby ensuring that a complete copy of the source volume is always available), the persistent relationship will identify the target volume for the most recently completed FlashCopy.

Consider the following situation:

An installation wants to maintain more than one point-in-time copy version of a set of its production volumes. A FlashCopy could be done every eight hours, each time using a different set of target volumes. We will call the production volume A, and the three copies B, C, and D, each holding a different point-in-time copy version.

The procedure could be as follows:

1. FlashCopy from A to B
2. FlashCopy from A to C
3. FlashCopy from A to D

There is an exposure for mistakes. Suppose in step 2 that the FlashCopy is done from A to B rather than from A to C. This will destroy the original point-in-time copy version in volume B.

To avoid this problem, a FlashCopy relationship can be made persistent. This enforces that the relationship must be withdrawn explicitly before another FlashCopy can be done. This does not guarantee that a point-in-time copy will not be overwritten by mistake, but it does provide an extra check.

When a FlashCopy establish command is issued to a target that is in an existing FlashCopy relationship, the following message is received:

```
ANTF0414E FLASHCOPY ESTABLISH TARGET TRACK IS AN ACTIVE TARGET TRACK
```

Persistent FlashCopy relationships can only be initiated through the ESS Copy Services Web User Interface. For information on how to invoke this option, refer to 2.2.14, "ESS Copy Services Web User Interface" on page 54, and to the publication *IBM TotalStorage Enterprise Storage Server Web Interface User's Guide,* SC26-7448.

### FlashCopy no-background copy to background copy

With this option a FlashCopy relationship can be changed from no-background copy to background copy, thus forcing the background copy of remaining unchanged tracks.

This request can only be initiated through the ESS Copy Services Web User Interface. For information on how to invoke this option, refer to 2.2.14, "ESS Copy Services Web User Interface" on page 54, and to the publication *IBM TotalStorage Enterprise Storage Server Web Interface User's Guide,* SC26-7448.

## 2.1.3 How to invoke FlashCopy

For z/OS and OS/390®, FlashCopy can be invoked in the following ways:

- DFSMSdss utility
- TSO commands
- ICKDSF
- ANTRQST macro
- ESS Copy Services Web User Interface
- Starting with LIC level 2.3.0, the ESS API is enhanced to support ESS Copy Services configuration and use for PPRC and FlashCopy for the ESS Model 800. The ESS API is discussed in Appendix A, "ESS Application Programming Interface (API)" on page 351.

In the following sections we briefly describe these different ways of invoking and controlling the FlashCopy function. Each of these are discussed to a greater extent in the following publications:

- *z/OS DFSMSdss Storage Administration Reference,* SC35-0424
- *z/OS DFSMS Advanced Copy Services,* SC35-0428
- *z/OS DFSMSdfp Advanced Services,* SC26-7400
- *IBM TotalStorage Enterprise Storage Server Web Interface User's Guide,* SC26-7448
- *Device Support Facilities User's Guide and Reference Release 17*, GC35-0033

## 2.1.4 DFSMSdss utility

The following sections discuss how to use the DFSMSdss utility to invoke FlashCopy Version 1 functions. For more detailed information on DFSMSdss, refer to the publication *z/OS DFSMSdss Storage Administration Reference,* SC35-0424.

DFSMSdss can use FlashCopy V1 to perform a full volume copy of the data from the source device to the target device when the following conditions are met:

- The requested DFSMSdss operation is COPY FULL.
- Both the source and target devices are in the same ESS logical subsystem (LSS) and the ESS has the FlashCopy feature enabled.
- The source and target devices must have the same track format.
- The target must be at least as large as the source volume.
- The source and target volumes must be online.
- The designated target volume cannot also be a primary volume in an XRC or PPRC volume pair.
- A volume can be in only one FlashCopy relationship at a time. The physical copy must complete or the FlashCopy relationship must terminate in order to start another one.

Whenever the conditions for using FlashCopy are met, DFSMSdss will automatically invoke FlashCopy. The copy job will complete after a minimum time when the FlashCopy relationship has been established. If another copy job is started while the source still is in a FlashCopy relationship with another volume, then in this case DFSMSdss will start a software host copy.

### COPYVOLID parameter

The COPYVOLID parameter for the COPY command specifies that the volume serial number from the source volume is to be copied to the target volume. This creates an identical copy of the source volume, including the volume serial. The target volume will go offline.

Figure 2-2 shows how the COPYVOLID parameter affects the output of a DFSMSdss full volume COPY operation.

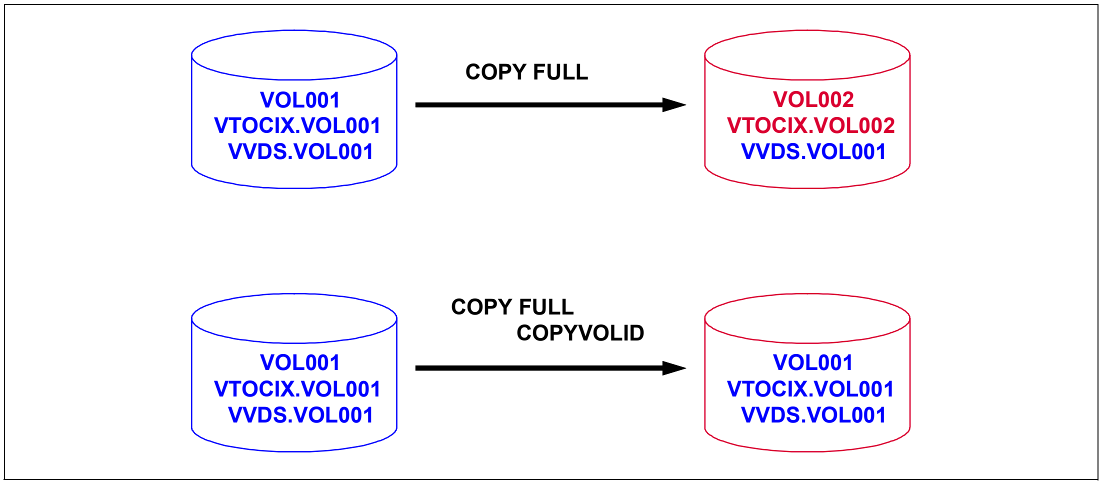



*Figure 2-2 DFSMSdss COPY with COPYVOLID*

Example 2-1 illustrates how DFSMSdss can be used to invoke FlashCopy in batch. In this example, because the COPYVOLID keyword is specified the volume serial (volser), the VTOC index and VVDS of the target volume will be identical to the source volume.

*Example 2-1 DFSMSdss COPY FULL with COPYVOLID*

```
//COPYFULL JOB .....
//*
//CPYVOLID EXEC PGM=ADRDSSU
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  COPY FULL INDYNAM(SRCDEV) OUTDYNAM(TRGDEV) COPYVOLID
/*
```

## DUMPCONDITIONING parameter

The DUMPCONDITIONING parameter of the DFSMSdss COPY FULL command allows both the source and target volumes to remain online after a full volume copy operation, thus creating an interim copy for a subsequent dump to tape (or DASD) that can be done using the same system.

When DUMPCONDITIONING is specified, the volume serial (volser) of the target volume does not change. The VVDS and VTOC index names on the target volume will not change to match the target volume serial, instead they will continue to match the source volume. This volume is a *conditioned volume*. A conditioned volume is not usable in its current state —except for the DFSMSdss DUMP operation, because the volume serial, the VTOC index and VVDS names are not consistent.

A full volume dump of the conditioned volume results in a dump data set that looks like it was created by dumping the source volume. This allows the dump data set to be restored and used without having to clip back the volser.

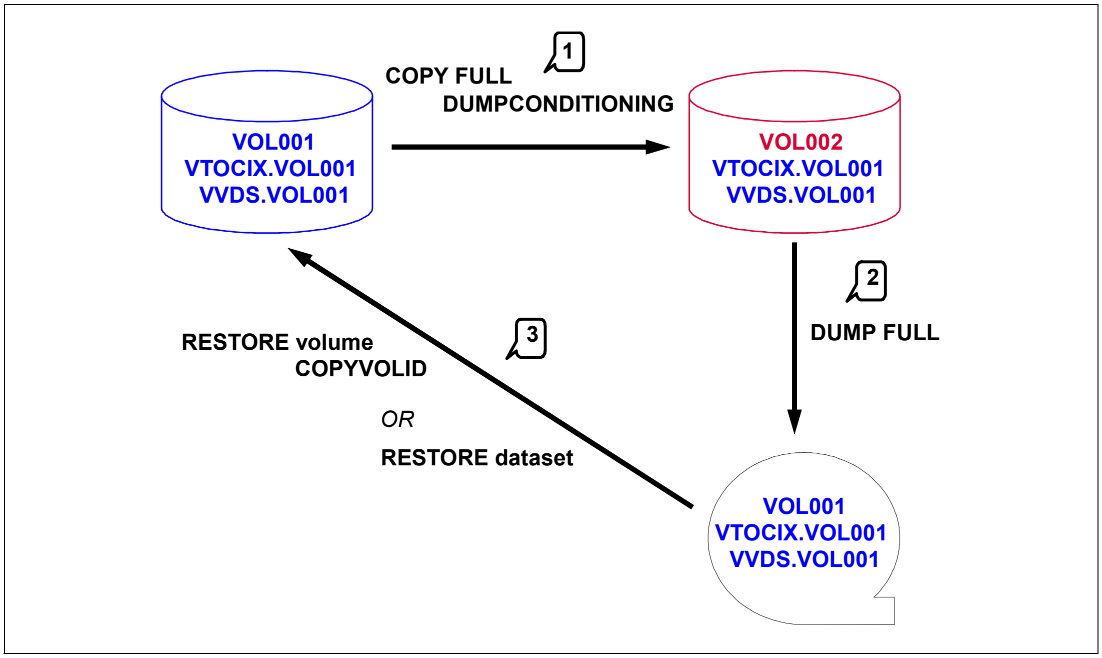



*Figure 2-3 DFSMSdss COPY with DUMPCONDITIONING*

Figure 2-3 illustrates how the volume serial, VTOC index, and VVDS of the copies change during a backup-restore cycle using the DFSMSdss DUMPCONDITIONING option. Example 2-2 illustrates sample JCL that may be used for each stage of the backup-restore cycle.

*Example 2-2 Backup-restore cycle with DUMPCONDITIONING and background copy*

```
//BACKUP JOB .....
//*
//* Step 1 - COPY FULL with DUMPCONDITIONING
//*
//STEP1    EXEC PGM=ADRDSSU
//SYSPRINT  DD  SYSOUT=*
//INDD      DD  UNIT=3390,VOL=SER=RS1510,DISP=SHR
//OUTDD     DD  UNIT=3390,VOL=SER=RS1511,DISP=SHR
//SYSIN     DD  *
  COPY FULL INDDNAME(INDD) OUTDDNAME(OUTDD) DUMPCONDITIONING
/*
//*
//* Step 2 - DUMP FULL
//*
//STEP2    EXEC PGM=ADRDSSU
//SYSPRINT  DD  SYSOUT=*
//INDD      DD  UNIT=3390,VOL=SER=RS1511,DISP=SHR
//OUTDD     DD  DSN=BACKUP.RS1510,DISP=(,KEEP),LABEL=(1,SL),
//              UNIT=3490,VOL=SER=(TAPE01,TAPE02,TAPE03)
//SYSIN     DD  *
  DUMP FULL INDDNAME(INDD) OUTDDNAME(OUTDD)
/*
//*
//* Step 3 - RESTORE FULL with COPYVOLID
//*
//STEP3    EXEC PGM=ADRDSSU
//SYSPRINT  DD  SYSOUT=*
//INDD      DD  DSN=BACKUP.RS1510,DISP=SHR,LABEL=(1,SL),
```

```
//          UNIT=3490,VOL=SER=(TAPE01,TAPE02,TAPE03)
//OUTDD     DD  UNIT=3390,VOL=SER=RS1512,DISP=SHR
//SYSIN     DD  *
  RESTORE INDDNAME(INDD) OUTDDNAME(OUTDD) COPYVOLID
/*
```

## FCNOCOPY and FCWITHDRAW parameters

The FCNOCOPY and FCWITHDRAW parameters can be used to make tape backups of the source data, using FlashCopy with the no-background option.

### *FCNOCOPY parameter*

When DFSMSdss uses FlashCopy to perform a copy operation, the background copy task is started by default.

FCNOCOPY specifies that if FlashCopy is used to perform the copy operation, then the ESS does not perform a physical background copy of the data. When FlashCopy is not used to perform the copy operation, the FCNOCOPY keyword is ignored.

If FCNOCOPY is used, you must withdraw the FlashCopy relationship when the copy is no longer needed. This can be accomplished by performing a full volume dump of the target volume with the FCWITHDRAW keyword, or by using the TSO FCWITHDR command.

### *FCWITHDRAW parameter*

FCWITHDRAW specifies that if the volume that is dumped is the target volume of a FlashCopy relationship, then the FlashCopy relationship is withdrawn when the dump has successfully completed. When the volume that is dumped is not the target volume of a FlashCopy relationship, the FCWITHDRAW keyword is ignored.

### *Example of FCNOCOPY and FCWITHDRAW use*

Doing the volume copy without running the background copy task is an efficient procedure when making tape backups for example. The tape dump is taken from the target volume that is holding a time-zero copy of the source volume (the ESS is keeping the integrity while the relationship exists). The procedure can be done as follows:

1. Start the FlashCopy with the no-background copy option using the COPY command with the FCNOCOPY parameter.
2. Then dump the FlashCopy target to tape using the DUMP command with the FCWITHDRAW parameter.
3. When the tape dump completes successfully, the FlashCopy relationship will be removed automatically.

Example 2-3 illustrates how the backup cycle (steps 1 and 2) shown in Figure 2-3 on page 20 can be implemented with the no-background copy option.

*Example 2-3 Backup to tape with no-background copy*

```
//BACKUP JOB .....
//*
//* Step 1 - COPY FULL with DUMPCONDITIONING & FCNOCOPY
//*
//STEP1    EXEC PGM=ADRDSSU
//SYSPRINT  DD  SYSOUT=*
//INDD      DD  UNIT=3390,VOL=SER=RS1510,DISP=SHR
//OUTDD     DD  UNIT=3390,VOL=SER=RS1511,DISP=SHR
//SYSIN     DD  *
  COPY FULL INDDNAME(INDD) OUTDDNAME(OUTDD) DUMPCONDITIONING FCNOCOPY
```

```
/*
//*
//* Step 2 - DUMP FULL with FCWITHDRAW
//*
//STEP2    EXEC PGM=ADRDSSU
//SYSPRINT  DD  SYSOUT=*
//INDD      DD  UNIT=3390,VOL=SER=RS1511,DISP=SHR
//OUTDD     DD  DSN=BACKUP.RS1510,DISP=(,KEEP),LABEL=(1,SL),
//          UNIT=3490,VOL=SER=(TAPE01,TAPE02,TAPE03)
//SYSIN     DD  *
  DUMP FULL INDDNAME(INDD) OUTDDNAME(OUTDD) FCWITHDRAW
/*
```

When the FlashCopy relationship is withdrawn, the target volume is left in an indeterminate state. Some of the tracks on the volume may contain data from the source volume while other tracks may contain residual data that was on the target volume before the copy. This indeterminate state can cause problems when accessing the target volume following the dump, if the VTOC locations of the source volumes and the target volumes are different before the copy. To avoid these problems, perform one of the following actions:

- Ensure that the VTOC locations for the source volumes and the target volumes are the same before you initiate the copy.
- Initialize the target volume using ICKDSF to return the volume to a consistent state.

### Problem determination

For operations in which FlashCopy is not attempted, the DEBUG=FRMSG parameter can be used to aid in determining why FlashCopy cannot be used. When PARM='DEBUG=FRMSG' is specified on the EXEC statement for DFSMSdss, this parameter causes DFSMSdss to issue an informational message during copy operations for which fast replication methods, such as SnapShot and FlashCopy, cannot be used. The informational message includes a reason code that indicates why DFSMSdss could not use fast replication.

FRMSG (MINIMAL/SUMMARIZERD/DETAILED) can be specified as a subparameter of the DEBUG parameter of the COPY command. The keyword used in the COPY command overrides the optional DEBUG=FRMSG parameter of the JCL EXEC statement.

## 2.1.5 TSO commands

In z/OS and OS/390 systems, FlashCopy can also be invoked using TSO commands. This section describes the commands that can be used to control FlashCopy. For more detailed information on the TSO commands for FlashCopy, refer to the IBM publication *z/OS DFSMS Advanced Copy Services,* SC35-0428.

FlashCopy TSO commands can be issued against the source or target volumes even if they are offline to the operating system. There are three TSO commands for FlashCopy:

| | |
|---|---|
| **FCESTABL** | Establishes a FlashCopy relationship |
| **FCQUERY** | Queries an ESS device |
| **FCWITHDR** | Withdraws a FlashCopy relationship |

### Security

FlashCopy TSO commands are authorized TSO commands. It is recommended that access to these commands be restricted by either putting these commands in a RACF® protected library, or defining resource profiles in the RACF FACILITY class, and restricting access to those profiles.

### *Placing FlashCopy commands in a RACF protected library*

TSO commands can be placed in a RACF protected library to restrict FlashCopy TSO commands to authorized storage administrators.

To RACF protect FlashCopy commands, the following steps should be performed:

1. Issue the following RDEFINE command for each FlashCopy command, and for each command abbreviation that you want defined to RACF:

   **RDEFINE PROGRAM *cmdname* ADDMEM('SYS1.CMDLIB')/*volser*/NOPADCHK) UACC(NONE)**

   The following terms apply to the above example:

   **cmdname** Defines the FlashCopy TSO command name or an abbreviation of a command. Issue a separate RDEFINE command for each command, and any command abbreviations you plan to use. RACF can only perform checking on commands and abbreviations that are defined to it.

   **volser** Defines the name of the volume that contains the SYS1.CMDLIB data set.

2. Issue the PERMIT command for all commands, and authorized FlashCopy TSO command users as follows:

   **PERMIT *cmdname* CLASS(PROGRAM) ID(*name*) ACCESS(READ)**

   The following terms apply to the above example:

   **cmdname** Defines the FlashCopy TSO command or an abbreviation of a command.

   **name** Defines the user ID that will receive RACF access authority to that command.

3. Issue the following SETROPTS command from a user ID that has the appropriate authority:

   **SETROPTS CLASSACT(PROGRAM) WHEN(PROGRAM) REFRESH**

### *Defining resource profiles in the RACF FACILITY class*

You can limit the use of FlashCopy commands by defining resource profiles in the RACF FACILITY class, and restricting access to those profiles. To use a protected command, you need read-access authority to the applicable profile.

Table 2-1 lists the FlashCopy commands and the FACILITY class profiles that can restrict them. Refer to the z/OS Security Server publications of your current RACF release for details on activating the RACF FACILITY class, defining, and authorizing users to the FlashCopy command profiles.

*Table 2-1 FlashCopy FACILITY class profile names*

| Command | Profile name |
|---|---|
| FCESTABL | **STGADMIN.ANT.ESFC.COMMANDS** |
| FCQUERY | |
| FCWITHDR | |
| FCQUERY | **STGADMIN.ANT.ESFC.FCQUERY** |

You can authorize the FCQUERY command with either STGADMIN.ANT.ESFC.COMMANDS profile or STGADMIN.ANT.ESFC.FCQUERY profile. FlashCopy first checks STGADMIN.ANT.ESFC.COMMANDS for authorization. If authorization is not permitted with the STGADMIN.ANT.ESFC.COMMANDS profile, then FlashCopy checks the STGADMIN.ANT.ESFC.FCQUERY profile for authorization to issue the FCQUERY command.

The following examples activate the RACF FACILITY class, define the profile for FlashCopy commands, and give users STGADMIN authority to use this profile:

1. Activate the RACF FACILITY class:

   **`SETROPTS CLASSACT(FACILITY)`**

2. Define the profile for FlashCopy commands, and authorize user STGADMIN to use this profile:

   ```
   RDEFINE FACILITY STGADMIN.ANT.ESFC.COMMANDS UACC(NONE)
   PERMIT STGADMIN.ANT.ESFC.COMMANDS CLASS(FACILITY) ID(STGADMIN) ACCESS(READ)
   ```

### *IKJTSOxx*

Authorize FlashCopy TSO commands by adding the command names to the AUTHCMD PARM parameter of the IKJTSOxx member of SYS1.PARMLIB. After you have added the FlashCopy command names to the IKJTSOxx member, issue the TSO command PARMLIB UPDATE(xx) to activate the new IKJTSOxx member.

## FCESTABL command

Use the FCESTABL TSO command to establish a FlashCopy relationship between a source and a target device as shown in Example 2-4.

*Example 2-4 FCESTABL command*

```
FCESTABL SDEVN(X'9400') TDEVN(X'9470') MODE(COPY)
```

Example 2-4 initiates a FlashCopy operation from source device 9400 to target device 9470 with background copy (the default).

### *Required parameters:*

**`SDEVN`** Specifies the four-digit hexadecimal device number of the source device. The source device can be online or offline.

**`TDEVN`** Specifies the four-digit hexadecimal device number of the target device.

### *Optional parameters:*

**`MODE`** Specifies the type of FlashCopy operation.
**`MODE(COPY)`** specifies that the background copy task will be initiated to copy all source tracks to the target device. This is the default.
**`MODE(NOCOPY)`** is used to establish a FlashCopy relationship without doing the background copy.

**`ONLINTGT`** Specifies whether an online device can be used as the target of a FlashCopy relationship.
**`ONLINTGT(YES)`** allows a FlashCopy relationship to be established even if the target device appears to be online. It is required if the target device is online to any host system.
**`ONLINTGT(NO)`** allows a FlashCopy relationship to be established only if the target device is offline. This is the default.

**`EXTENTS`** Specifies the first and last track of an extent pair to be part of this FlashCopy relationship. A maximum of five extent pairs can be specified. All tracks within the extent pair range become part of the FlashCopy relationship. Tracks in specified extents will be copied to the same CCHH location on the target device. See Figure 2-4.
If **`EXTENTS`** is not specified, the FlashCopy FCESTABL command defaults to all tracks of the source and target volumes (full volume).

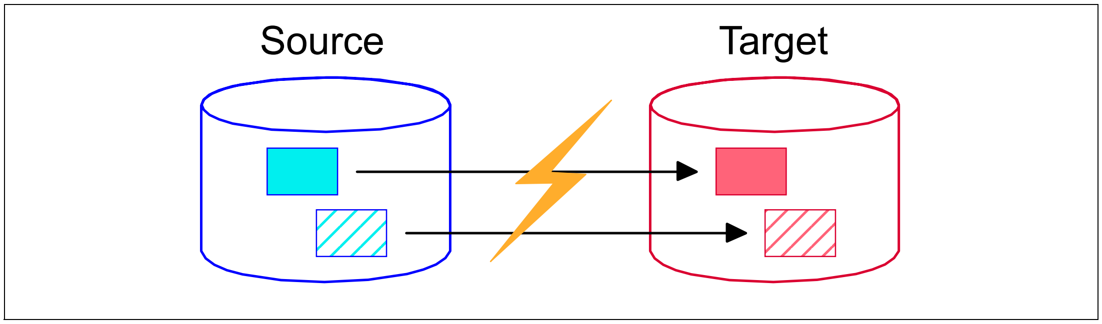



*Figure 2-4 FCESTABL with multiple extents*

## FCQUERY command

Use the FCQUERY TSO command to determine the copy status of an ESS device available to the host system.

Example 2-5 queries the status of device 9400. A sample output of the FCQUERY command is shown in Figure 2-5.

*Example 2-5 FCQUERY command*

```
FCQUERY DEVN(X'9400')
```

### *Required parameters:*

**DEVN** Specifies the four-digit hexadecimal device number of the device.

### *Optional parameters:*

**UNFORMAT** Specifies that the information returned from the FCQUERY is to be formatted as a string of values separated by commas, similar to the information string returned to the ANTRQST API caller of FCQUERY.
The default is formatted data.

```
ANTF0090I FCQUERY Formatted
DEVN SSID LSS CCA CU   SERIAL     STATUS
9400 1384 04  00  2105 0000314361 FC .....88%
---- 1384 04  70  2105 0000314361 FC ........
```

*Figure 2-5 FCQUERY command sample output*

The status area could have any of the following values:

- **SIMPLEX** - The volume is not in any copy mode.
- **CC** - The volume is in a Concurrent Copy session.
- **XRC** - The volume is an XRC primary volume.
- **PPRC** - The volume is in a PPRC relationship.
- **FC** - The volume is in a FlashCopy relationship. If background copy option is active, the percent of tracks copied is displayed on the source device output line.

The first output line in Figure 2-5 shows device 9400 is the source volume in a FlashCopy relation with background copy, the background copy is 88% complete. The second output line shows the target volume of the relation is CCA X'70' in the same LSS.

### FCWITHDR command

The FCWITHDR TSO command is used to end a FlashCopy relationship. This command stops a FlashCopy operation, regardless of whether background copy is active or not.

Whenever a FlashCopy relationship is withdrawn and all the source tracks are not copied, then the target volume will be left in an indeterminate state. With such an event, it is recommended that the WITHDRAW command be followed by an ICKDSF INIT operation.

Example 2-6 withdraws a FlashCopy relationship, which was established between source device 9400 to target device 9470.

*Example 2-6 FCWITHDR command*

```
FCWITHDR SDEVN(X'9400') TDEVN(X'9470')
```

#### *Required parameters:*

**TDEVN** Specifies the four-digit hexadecimal device number of the target device

#### *Optional parameters:*

**SDEVN** Specifies the four-digit hexadecimal device number of the source device

## 2.1.6 ANTRQST API interface

The ANTRQST macro provides an application program call to the z/OS system data mover's (SDM) application programming interface (API). This macro allows you to call XRC, PPRC, and FlashCopy functions.

The ILK keyword can be used to categorize the ANTRQST macro request into the advanced copy services families. The ANTRQST macro supports the FlashCopy requests FCESTABLISH, FCQUERY, and FCWITHDRAW.

For FlashCopy requests, the ANTMAIN address space must be active at the time of a macro invocation. This address space is automatically started at system initialization time.

These requests perform the same functions as the TSO FlashCopy commands described in 2.1.5, "TSO commands" on page 22. For detailed information, see the IBM publication *z/OS DFSMSdfp Advanced Services,* SC26-7400.

## 2.1.7 ESS Copy Services Web User Interface

You can invoke and control FlashCopy functions using the ESS Copy Services Web User Interface (WUI), for all the zSeries operating environments. Also, if a supported server is attached, the saved tasks can be invoked from the ESS Copy Services Command Line interface (CLI).

Chapter 8, "Solutions" on page 409, illustrates how to establish and how to withdraw a FlashCopy relationship using the ESS Copy Services WUI. It also illustrates how to invoke two later enhancements of FlashCopy that can only be requested using the Web interface: Persistent FlashCopy and FlashCopy Start Background Copy.

Chapter 7, "ESS Copy Services Web User Interface" on page 299, discusses the Copy Services server and the ESS Copy Services Web User Interface in greater detail.

When planning to use the ESS Copy Services WUI for FlashCopy invocation, refer to the following IBM publication for detailed descriptions showing how to use the information: *IBM TotalStorage Enterprise Storage Server Web Interface User's Guide,* SC26-7448.

## 2.1.8 ICKDSF utility

With APAR PQ84968, ICKDSF is able to manage Asynchronous PPRC as well as FlashCopy. The following new FlashCopy commands are introduced:

```
FLASHCOPY ESTABLISH
FLASHCOPY WITHDRAW
FLASHCOPY QUERY
FLASHCOPY QUERY INCREMENTSTATUS
FLASHCOPY QUERY RELATIONS
```

These new commands are described in section 4.3, “Using ICKDSF to control Copy Services” on page 156.

Prior to this APAR only the following command could be used, the ICKDSF CONTROL command.

The ICKDSF CONTROL command is used to reset certain subsystem conditions: this feature is not for normal use, but to reset a condition after some special event. The REMOVEFCPY parameter of the CONTROL command can be used to tell the ESS to remove FlashCopy information. When REMOVEFCPY is specified, the ESS will remove *all* FlashCopy relationships for all FlashCopy pairs from the corresponding logical subsystem (LSS).

The ICKDSF CONTROL REMOVEFCPY command can be issued against both online and offline devices. Example 2-7 shows an ICKDSF CONTROL command issued against an offline device.

*Example 2-7 ICKDSF CONTROL command*

```
CONTROL UNIT(4700) REMOVEFCPY
```

For more detailed information on the ICKDSF CONTROL command, refer to the IBM publication *Device Support Facilities User’s Guide and Reference,* GC35-0033.

**Note:** Although the CONTROL command can be used for specifically removing FlashCopy relationship information from the ESS, the ICKDSF INIT command, no matter that it is intended for a different purpose, also removes the FlashCopy relationships of the specified volume.

## 2.1.9 Support for other zSeries operating systems

This section summarizes the FlashCopy support available on VSE/ESA™, z/VM® and TPF.

### VSE/ESA support

The ESS Copy Services Web User Interface can be used to invoke and control FlashCopy in a VSE/ESA environment. The FlashCopy support for ESS is also part of VSE/ESA as shipped. The name of the phase including this support is $IJBIXFP residing in sub library IJSYSRS.SYSLIB. This phase is automatically loaded into the SVA during startup. FlashCopy support for VSE is provided by:

- IXFP SNAP command
- VSAM SYNONYM command. This has not been retrofitted.
- VSE/Fast Copy
- ICKDSF

### IXFP SNAP

The IXFP SNAP command can be invoked in two different ways:

- Attention routine interface
- Batch interface

Example 2-8 provides two examples of IXFP SNAP commands issued through the attention routine interface.

*Example 2-8 IXFP SNAP commands*

```
IXFP SNAP,80E:80F
IXFP SNAP,WORK01:WORK21,VOL1=VSDAT1,NOPROMPT
```

The IXFP SNAP command can also be executed in batch through the DTRIATTN program, as shown in Example 2-9.

*Example 2-9 VSE/ESA FlashCopy batch job*

```
//JOB FLASHCOPY VOLUME
//EXEC DTRIATTN,PARM='IXFP SNAP,80E:80F,VOL1=VSDAT2,NOPROMPT'
```

Considerations for using IXFP SNAP:

- Copying cylinder 0 (overwriting the volume label) may result in two volumes being online at the same time with the same volume serial number. VSE does allow this.
- The target volume must be offline (DVCDN).

For further information on the IXFP SNAP commands, see the IBM publication *VSE/ESA Release Guide,* SC33-6718.

### VSAM SYNONYM

Enhancements to VSE/VSAM Backup/Restore allow you to backup selected VSAM data sets from volumes that have been duplicated using FlashCopy function. For further information on using the VSAM SYNONYM command, see the IBM publication *VSE/ESA Release Guide,* SC33-6718.

### VSE/Fast Copy

This utility exploits the FlashCopy function when using ESS devices. It supports full volume backup from disk. The following VSE/Fast Copy commands are supported:

- COPY ALL
- COPY VOLUME

Optional parameters IV (input volume), OV (output volume), and NV (new volume) are supported; while optional parameters NOPROMPT, NOVERIFY, and LIST are tolerated.

Optional parameters EXCLUDE, NOVSAM, and NOEXPIRED, when specified with COPY VOLUME command, will force FlashCopy to not be used.

For further information on VSE/Fast Copy see the IBM publication *VSE/ESA System Utilities,* SC33-6717.

### ICKDSF

See 2.1.8, “ICKDSF utility” on page 27 for FlashCopy commands for VSE/ESA.

### TPF support

There are several options for running FlashCopy in a TPF environment:

- From the ESS Copy Services Web User Interface, if a supported server is attached
- From saved tasks that can be invoked from the ESS Copy Services Command Line Interface (CLI)
- From TPF itself
- From another server that provides support for control of Copy Services

To use an ESS in a TPF system, at least one logical subsystem in the ESS has to be defined to operate in IBM 3990 Model 3 TPF control unit emulation mode. The volumes defined in this logical subsystem can then be used by TPF.

### z/VM support

FlashCopy is supported in the VM environment. The ESS Copy Services Web User Interface can be used to invoke and control FlashCopy in a VM environment. This section discusses the CP support and the guest support for the FlashCopy function.

#### *CP support*

z/VM supports a native CP user's ability to initiate a FlashCopy function from a source address to a target address on a ESS. FlashCopy provides the ability to make copies between any mix of Log Structured Array (LSA) and non-LSA (Home Address) source and target DASD. Users will find this feature especially beneficial for large databases which normally require a long time to copy. FlashCopy support includes the CP FLASHCOPY command.

With FlashCopy V1, the CP FLASHCOPY command is available to copy a virtual volume to another virtual volume in the same LSS. You can also specify a range of cylinders to be copied. Example 2-10 shows two CP FLASHCOPY commands, first to copy an entire volume and the second to copy a range of cylinders.

*Example 2-10*

```
FLASHCOPY F100 0 END TO F109 0 END
FLASHCOPY F100 0 999 TO F109 0 999
```

Some considerations:

- The target volume must be in write mode.
- If the whole volume is to be copied, then use **END** for the ending cylinder number.
- If cylinder 0 is copied, the target device may become a duplicate of the source device.
- Whole cylinders must be copied. Track extents are not allowed.
- The default background copy option is used; but "no background copy" cannot be used.

For more detailed information, refer to the IBM publication *z/VM CP Command and Utility Reference*, SC24-5967.

#### *Guest system support*

VM/ESA® and z/VM provide FlashCopy support for guest systems for attached volumes, as well as for full-pack minidisks for z/OS, VSE/ESA, and TPF guests.

## 2.1.10 Guidelines and recommendations

In this section, we discuss guidelines and recommendations for using FlashCopy in a zSeries environment.

## Performance

The aim of FlashCopy is to have minimal impact on applications that use the volumes participating in a FlashCopy relationship, either as a source or target.

The majority of work of the FlashCopy background copy is done asynchronously to the host operations at stage/destage time, thus not impacting the applications. At destage time for source track changes in cache, the time-zero image of the source track on the DASD is written to the target location before the DASD image is changed. Similarly, at destage time for target track changes in cache, the unmodified records for the target track are read from the time-zero source track on DASD and updated.

Here are some considerations for planning an efficient FlashCopy operation:

- Select the FlashCopy target volume in a different rank for better performance.
- Distribute the FlashCopy pairs evenly across LSSs for better performance.
- FlashCopy establish times are similar with or without an active workload.
- Background copy rates will decrease when application I/Os are present.
- As FlashCopy V1 requires that source and target volumes must be in the same LSS, configuring 8 LSSs per ESS instead of 16 (more volumes in each LSS) can add flexibility to the FlashCopy configuration.
- If large numbers of FlashCopy relationships are started and the missing interrupt handler (MIH) value is too low, you might encounter START PENDING on application I/Os. The FlashCopy establish process will hold application I/Os for the period of time needed to complete the initialization. After the initialization process completes, the applications will continue. Setting the MIH to a higher value or pacing the launch of the FlashCopy establish operations would alleviate this issue.
- A FlashCopy relationship can be established requesting no background copy. This option allows more efficient use of ESS resources when a physical copy of the source volume is not needed.

Access to the FlashCopy TSO commands and the Web User Interface must be controlled. FlashCopy commands should be used with caution to avoid causing data corruption unintentionally.

## DFSMSdss and TSO considerations

There are some differences in how DFSMSdss and TSO commands (as well as the ESS Copy Services Web User Interface) operate when requesting a FlashCopy:

- TSO initiated FlashCopy establish is somewhat faster than the DFSMSdss process, though both processes provide very fast establish times. The extra time is used by the DFSMSdss process because of the additional functions provided by DFSMSdss that are not provided by TSO, such as reading the source VTOC, performing security verification and ensuring data integrity by reserving the volumes involved in the FlashCopy operation (the average volume reserve time is much shorter than the total establish time). The FlashCopy establish times are the same whether or not there is active workload directing I/Os to the ESS during the establish phase.
- The DFSMSdss method is very efficient when a background copy is required. DFSMSdss copies only the allocated space on the source volume, so considerable time savings can be realized. This is implemented by DFSMSdss passing to the ESS only the allocated track ranges to copy, thus allowing the background copy to complete faster for a partially allocated volume.

- DFSMSdss FCNOCOPY/FCWITHDRAW — You need to use these two keywords in your procedure when using FlashCopy. The FCNOCOPY keyword on the COPY command prevents the ESS subsystem from performing a full physical copy of the volume, thus saving some ESS resources. Specifying the FCWITHDRAW keyword on the DUMP command causes DFSMSdss to withdraw the FlashCopy relationship after the volume has been successfully dumped. This frees the subsystem resources that are used to maintain the FlashCopy relationship. See “FCNOCOPY and FCWITHDRAW parameters” on page 21 for more explanation and an example.

# 2.2 FlashCopy Version 2

This section describes the functions available with FlashCopy Version 2, and discusses considerations for planning FlashCopy V2 implementation.

FlashCopy Version 2 supports all the previous FlashCopy V1 functions plus enhancements that we present in the current chapter. FlashCopy Version 1 is discussed in 2.1, “FlashCopy Version 1” on page 14.

## 2.2.1 Overview

FlashCopy provides the ability to create point-in-time copies. As soon as the FlashCopy command is processed, both the source and target volumes are available for application use.

These are characteristics of FlashCopy V1:

- FlashCopy V1 works at volume level.
- The source and target volumes must be in the same ESS logical subsystem (LSS).
- A source and a target volume can only be involved in one FlashCopy relationship at a time.

With FlashCopy Version 2, the previous characteristics change, and new functionality has also been added:

- FlashCopy V2 can be used for data set copies, as well as volume copies.
- The source and target of a FlashCopy can be on different LSSs within an ESS.
- Multiple FlashCopy relationships are allowed.
- Incremental copies are possible.
- Inband commands can be sent over PPRC links to a remote site.
- FlashCopy Consistency Groups can be created.

FlashCopy V2 supports all the FlashCopy V1 functions plus the preceding list of enhancements. In addition, there has been a reduction in the FlashCopy establish times.

## 2.2.2 How FlashCopy works

The basic FlashCopy processing is explained in 2.1.2, “How FlashCopy works” on page 14, and the reader should refer to it to understand how FlashCopy operates. This section focuses on the aspects of the FlashCopy operation that have changed in FlashCopy V2, and discusses the new functions.

### Extents in a FlashCopy relationship

An *extent* is a contiguous set of allocated tracks. It consists of a beginning track, an end track, and all tracks in between. Extent size can range from a single track to an entire volume. FlashCopy V2 manages source-target relationships at the extent level.

A *FlashCopy relationship* exists between an extent in the source volume and an extent in the target volume, from the time you initiate a FlashCopy operation until either the ESS copies all data from the source extent to the target extent, or the FlashCopy relationship is explicitly withdrawn. The FlashCopy relationship is a *mapping* of the source extent to the target extent, which allows a time-zero image of the source extent to be copied to the target extent while keeping their integrity.

The following examples illustrate the concept of extents as managed by FlashCopy.

### *DFSMSdss copy volume*

DFSMSdss invokes FlashCopy to copy tracks (extents) when a COPY FULL operation is requested.

Consider volume VOL00A, which has data allocated on three extents, and for which the rest of the volume is unallocated free space (see Figure 2-6):

1. When a DFSMSdss COPY FULL command is issued to copy VOL00A to VOL00B, DFSMSdss will establish three FlashCopy relationships, one for each allocated extent on the source volume. A TSO FCQUERY command issued against VOL00A would show three active FlashCopy relationships on the volume. Similarly, a FCQUERY command issued against VOL00B will show three active FlashCopy relationships on the volume.

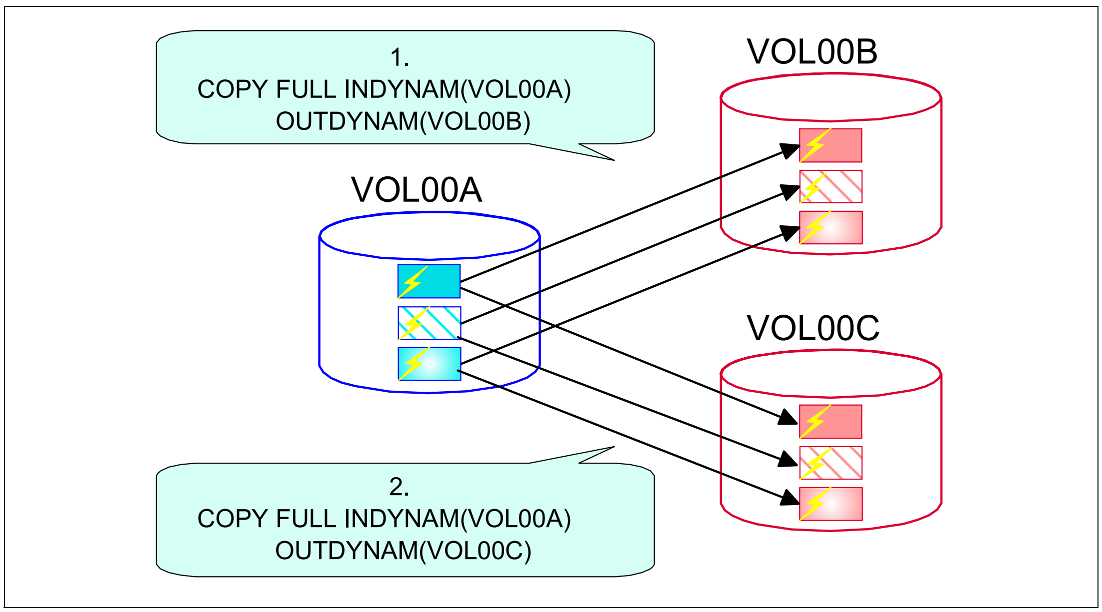



*Figure 2-6 FlashCopy relationships (extents)*

2. If two DFSMSdss COPY FULL commands are issued, one to copyVOL00A to VOL00B and the other to copy VOL00A to VOL00C, then DFSMSdss will establish six FlashCopy relationships. A TSO FCQUERY command issued against VOL00A will show six active FlashCopy relationships on the volume. A FCQUERY command issued against VOL00B or VOL00C will show three active FlashCopy relationships on each volume.

DFSMSdss will establish no more than 255 relationships for each full volume copy. If there are more than 255 extents on the source volume, some extents will be merged (as illustrated in Figure 2-7), so that the total number of relationships does not exceed 255. In this case some free space will be copied.

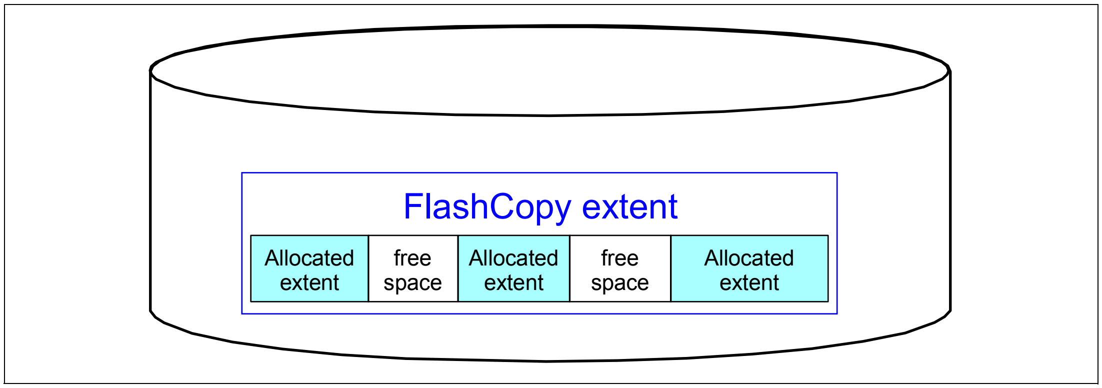



*Figure 2-7 Merging extents*

### *DFSMSdss copy data set*

When a data set FlashCopy operation is requested, one or more FlashCopy relationships may be established “under the covers” between source and target extents — the number of FlashCopy relationships depending on the number of extents the data set has.

Consider the example in Figure 2-8, where the data set SRC.DATA spans two extents on volume VOL00A. For this situation, when a DFSMSdss COPY DATASET command is issued to copy SRC.DATA from VOL00A to VOL00B, two FlashCopy relationships will be established — one for each extent of the source data set.

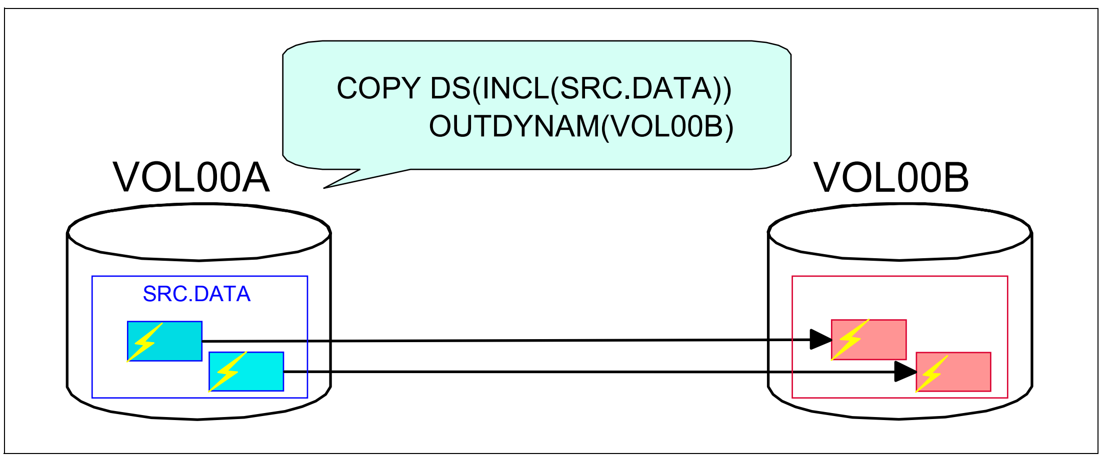



*Figure 2-8 Data set FlashCopy (extents)*

A TSO FCQUERY command issued against VOL00A will show two active FlashCopy relationships on the volume. A FCQUERY command against VOL00B will also show two active FlashCopy relationships.

## Number of FlashCopy relationships

For planning purposes, the following considerations should be regarded:

- The number of relationships per volume is limited. For a volume with $n$ tracks, the maximum number of relationships allowed for the volume is slightly less than $n/64$. Invoke the FCQUERY command to determine what the number is.
- The number of relationships for an ESS is also limited. For an ESS with $N$ tracks, the total number of relationships will be slightly less than $N/128$.

For most practical applications, the above limitations are never reached.

### FlashCopy extents: Terminology

In the current section, to do a detailed discussion on how FlashCopy works, we have referred to the association between a single source extent and the corresponding target extent as a FlashCopy relationship. All the multiple extent relationships that may eventually result are managed "under the covers" by the ESS, and presented to us as a single FlashCopy relationship, which we manage with single FlashCopy commands or requests.

For the rest of our discussions, the term *FlashCopy relationship* includes all the resulting source-target extent pair relationships that may result from the same FlashCopy establish request or command. And when we need to refer to the relationship between a single source and target extent pair, which is the relationship between an individual set of tracks of the multiple that may eventually be established by the ESS "under the covers"; we refer to it as *FlashCopy relationship (extent).*

## 2.2.3 Data set FlashCopy

FlashCopy V2 provides the ability for a source extent and target extent to be at different relative offsets from the beginning of their respective volumes, as illustrated in Figure 2-9.

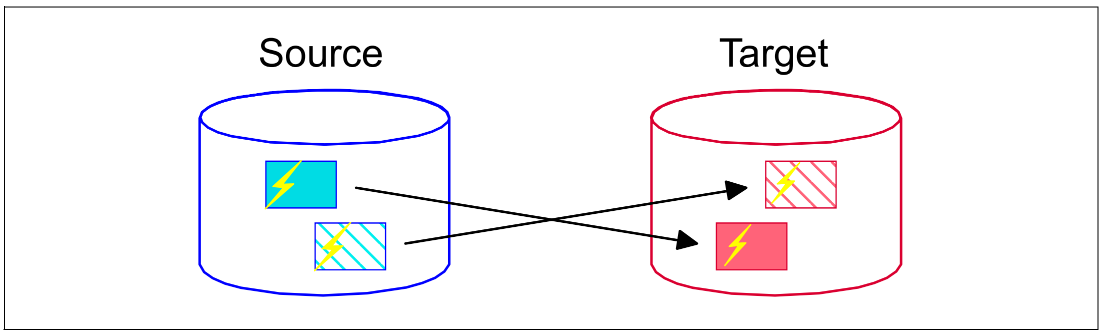



*Figure 2-9 Extent relocation*

Individual Data set FlashCopy is supported for the z/OS and OS/390 operating system users by means of the DFSMSdss utility. This function provides a new level of granularity to the IBM @server zSeries environments.

The following are characteristics of FlashCopy V2 data set copy:

- The source and target volumes do not need to be the same size.
- The location within the target volume does not need to match the location within the source volume.
- Data sets can be copied onto the same source volume.
- With multiple relationship support, FlashCopy V2 allows the same source extent to be copied to different extents on one or more target volumes, or even copy to an extent on the source volume as a target, up to twelve times. Multiple Relationship FlashCopy is explained in 2.2.4, "Multiple Relationship FlashCopy" on page 35.
- If the FlashCopy relationship was requested with no-background copy, a background copy may be started subsequently. For example, if a data set FlashCopy was requested with no-background copy and the source data set is deleted, DADSM (the Direct Access Device Space Management component of z/OS) will start the background copy in order to "clean up" any relationships that the data set may have been in (since the space that the data set occupied will become free space as a result of the delete, the free space must be free of any relationships).

Figure 2-10 shows various mapping possibilities when doing data set FlashCopy.

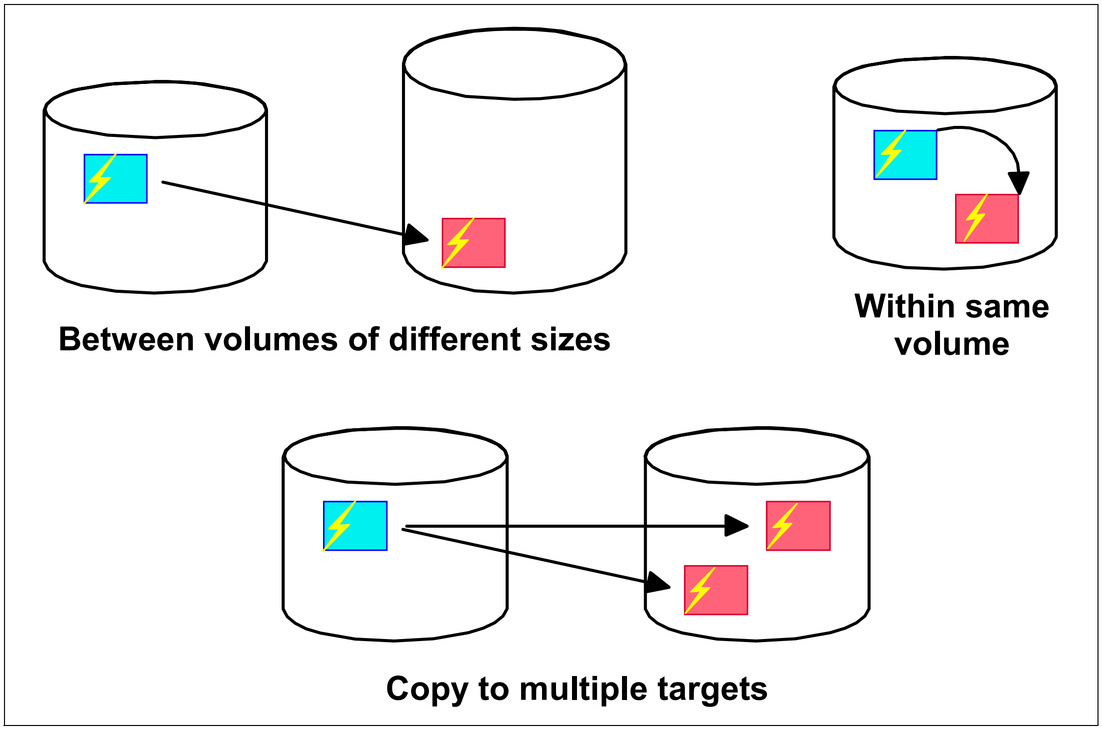



*Figure 2-10 Data set FlashCopy*

Data set FlashCopy is invoked through the DFSMSdss utility. Refer to 2.2.11, “DFSMSdss utility” on page 46 for details on when and how DFSMSdss invokes FlashCopy for data set copies.

## 2.2.4 Multiple Relationship FlashCopy

Multiple Relationship FlashCopy allows a source (extent, data set, or volume) to have multiple targets simultaneously:

- A source can have up to 12 targets.
- A target can only have one source.
- A volume or data set can be only a source or target at any given time.

Figure 2-11 shows examples of FlashCopy multiple relationships.

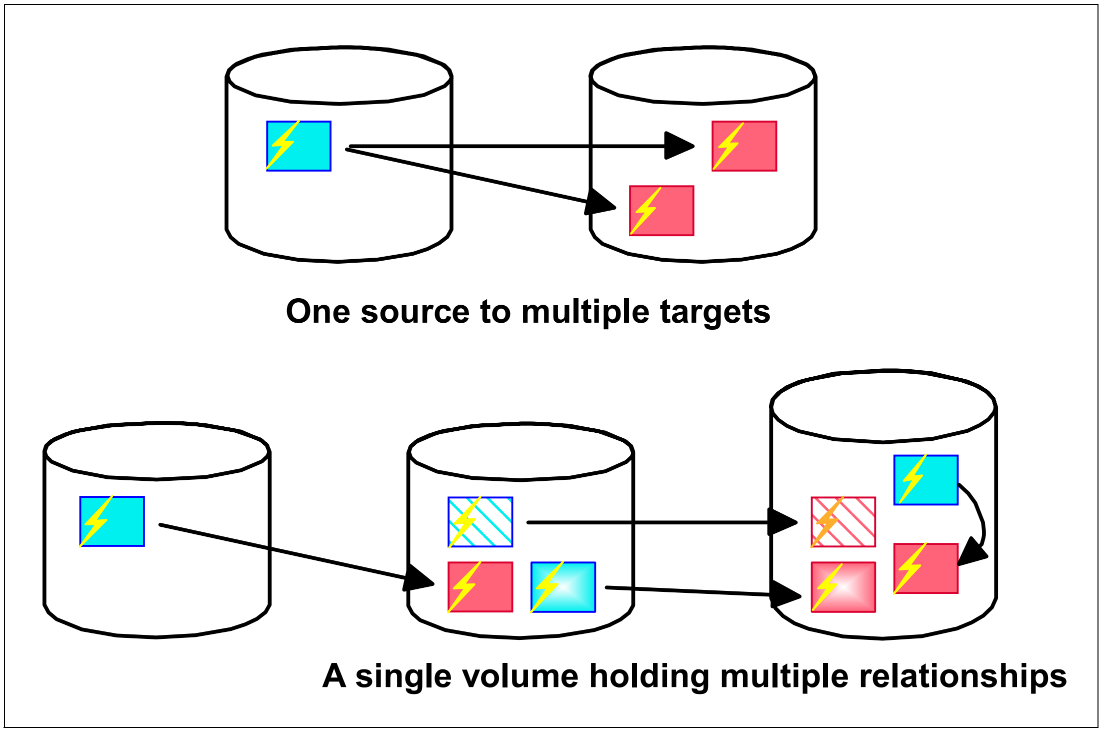



*Figure 2-11 Multiple Relationship FlashCopy*

## 2.2.5 Incremental FlashCopy

Incremental FlashCopy provides the capability to *refresh* a FlashCopy relationship. With Incremental FlashCopy, the initial relationship between a source and target is maintained. When a subsequent FlashCopy establish is initiated, the only data copied is that required to bring the target current with the source's newly established point-in-time.

Incremental FlashCopy helps reduce the background copy completion time when only a subset of data on either the source or target has changed, giving you the option to perform a FlashCopy on a more frequent basis.

Incremental FlashCopy can be invoked from the ESS Copy Services Web User Interface panels — FlashCopy TSO commands do not offer parameters to invoke this function. In order for an Incremental FlashCopy to be performed, the FlashCopy relationship must first be established with the *Start Change Recording* and *Persistent FlashCopy* options enabled:

- With change recording enabled, metadata structures are created to track changes to both source and target volumes from the time the relationship was established. The ESS can then identify the changed tracks that need to be copied when an Incremental FlashCopy is requested.
- Incremental FlashCopy is only possible with a persistent relationship. With persistent relationships, the relation between the source and target is maintained after the background copy has completed. This allows the ESS to continue tracking updates to the source and target extents.

**Note:** Incremental FlashCopy is supported at a volume level. It is not available for Data set FlashCopy.

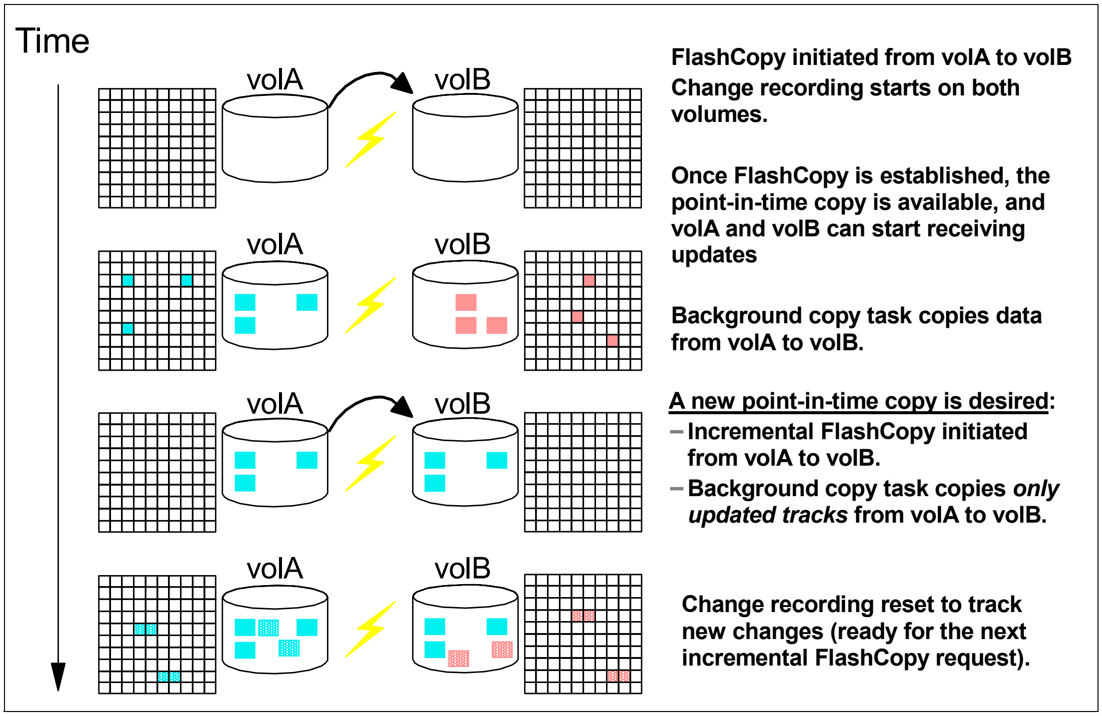



*Figure 2-12 Incremental FlashCopy*

Figure 2-12 illustrates how Incremental FlashCopy works:

1. When the initial FlashCopy is established from volA to volB with change recording enabled, FlashCopy creates metadata structures to track any updates that may be done on those volumes.
2. As changes are made to volA, FlashCopy will keep track of the tracks that are updated. Similarly, FlashCopy will keep track of any updates that are done on volB.
3. When an Incremental FlashCopy is requested from volA to volB, FlashCopy identifies the set of tracks that must be copied from volA to volB. For example, if volA had tracks 1, 3, 5 updated, and volB had tracks 2, 7, 9 updated, then tracks 1, 2, 3, 5, 7, 9 would be copied from volA to volB. In other words, all tracks necessary to make volB look like volA at the time of the flash will be identified to be copied from volA to volB.
4. Finally, the metadata structures are reset to start tracking changes from this point, if requested.

The direction of the refresh can also be reversed, in which case the volume previously defined as the target becomes the source for the volume previously defined as the source (and is now the target). Again, only the data required to bring the target current to the source's point-in-time is copied. If no updates were made to the target since the last refresh, the direction change could be used to restore the source back to the previous point-in-time state.

**Note:** The background copy must complete before the source and target volumes may be reversed.

Figure 2-13 shows an example of Reverse Restore from the target to the source. In the example we are discussing, when a Reverse Restore is done, tracks 1, 2, 3, 5, 7, 9 will be copied from volB to volA. That is, all tracks necessary to make volA look like volB at the time of the flash will be identified to be copied from volB to volA — now volB becomes the source of the FlashCopy pair, while volA becomes the target volume.

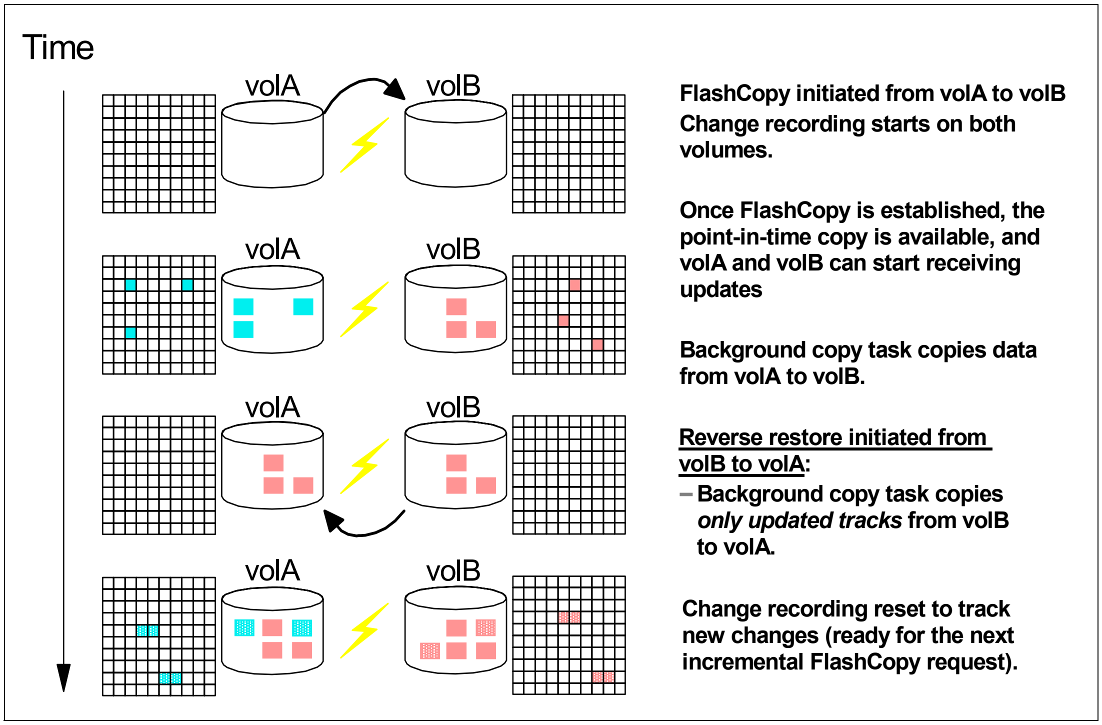



*Figure 2-13 Incremental FlashCopy - Reverse Restore*

Incremental FlashCopy can only be initiated through the ESS Copy Services Web User Interface. For more information on how to invoke this function, refer to "Incremental FlashCopy" on page 56. For detailed information, refer to the publication *IBM TotalStorage Enterprise Storage Server Web Interface User's Guide,* SC26-7448.

## 2.2.6 FlashCopy Consistency Group

When FlashCopy is established for a large number of source volumes, there is a finite amount of time between the first and last establish, so that copies will not be created at a consistent point-in-time.

With the **Freeze FlashCopy Consistency Group** option (Figure 2-14), the ESS will hold off I/O activity to a volume for a time period by putting the source volume in extended long busy state. Thus, a window can be created during which dependent write updates will not occur, and FlashCopy will use that window to obtain a consistent point-in-time copy of the related volumes. I/O activity resumes when a FlashCopy Consistency Group Created task is issued, or when the extended long busy (ELB) window expires (the ELB window is 2 minutes by default); see also "Consistency Group" on page 59.

Consistency groups can be used to help create a consistent point-in-time copy across multiple volumes, and even across multiple ESSs, thus managing the consistency of dependent writes.

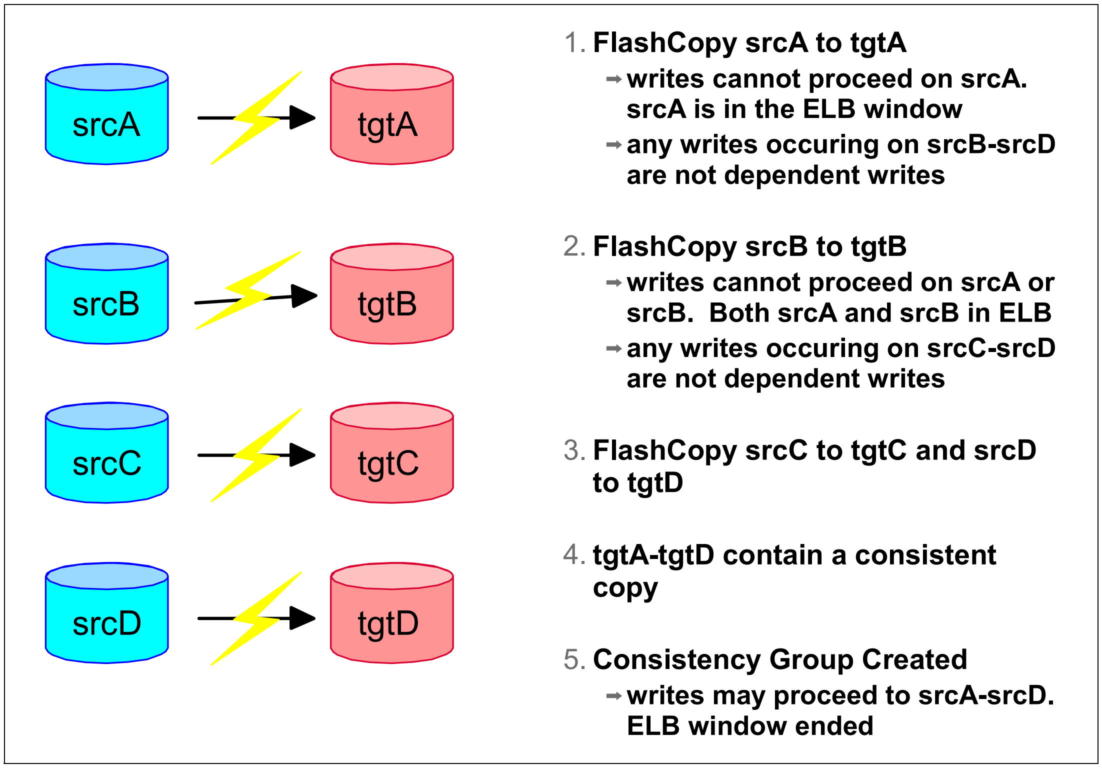



*Figure 2-14 FlashCopy Consistency Group*

Figure 2-14 illustrates how FlashCopy Consistency Groups can be used to manage the consistency of dependent writes:

1. FlashCopy is established from srcA to tgtA with the *Freeze FlashCopy Consistency Group* option enabled. Volume srcA is placed in extended long busy (ELB) state and thus, all the I/O activity to this volume is enqueued. Because writes on srcA cannot proceed, neither will its dependent writes on the other volumes (srcB, srcC, srcD) proceed. This ensures the logical integrity of the related data spanning the whole set of volumes. Independent writes to other volumes (srcB, srcC, srcD) are not affected.
2. FlashCopy is then established from srcB to tgtB with the Consistency Group function enabled. Volume srcB is placed in ELB state — as srcA currently is. Thus, volumes srcA and srcB are not receiving updates. Dependent writes on srcC and srcD waiting on write completions on srcA or srcB cannot proceed. This ensures the integrity of the data over the whole set of related volumes (srcA, srcB, srcC, srcD). Independent writes to volumes srcC and srcD are not affected.
3. As FlashCopy is established from srcC to tgtC, then from srcD to tgtD, each source volume is placed in ELB state and dependent writes held.
4. When all FlashCopy pairs have been established, volumes tgtA, tgtB, tgtC and tgtD contain a consistent copy of data (the order of dependent writes is preserved).
5. The **`Consistency Group Created`** command can be issued to remove the source volumes from ELB state, so that updates can resume.

The FlashCopy Consistency Group function can only be activated through the ESS Copy Services Web User Interface. For more information on how to invoke this function, refer to “Consistency Group” on page 59.

For detailed information, refer to the publication, *IBM TotalStorage Enterprise Storage Server Web Interface User’s Guide,* SC26-7448.

## 2.2.7 Inband command over PPRC links

In a PPRC environment, commands to manage FlashCopy at the remote site can now be issued from the local or intermediate site and transmitted over the PPRC links.

Inband commands are issued to a primary volume of a PPRC pair at a local ESS and sent across PPRC paths (acting as a conduit) to a remote ESS to enable a FlashCopy pair to be established at the remote site. The source volume selected for the FlashCopy at the remote site must be the secondary PPRC volume of the PPRC pair. The PPRC secondary volume becomes the source volume in a FlashCopy pair, as illustrated in Figure 2-15.

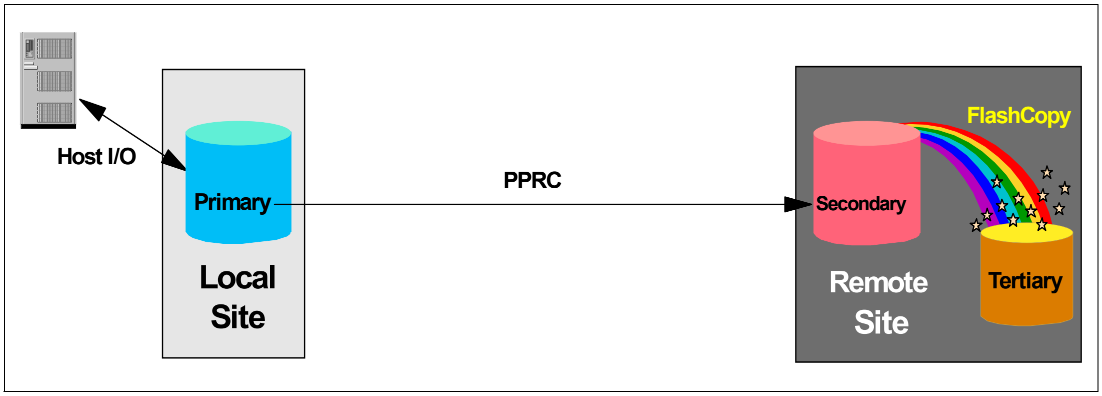



*Figure 2-15 Inband FlashCopy*

This new function eliminates the need for a network connection to the remote site solely for the management of FlashCopy.

Inband commands can only be invoked through the ESS Copy Services Web User Interface. For more information on how to invoke this option, refer to "Inband command" on page 62. For detailed information, refer to the publication *IBM TotalStorage Enterprise Storage Server Web Interface User's Guide,* SC26-7448.

## 2.2.8 Combinations of FlashCopy options

Figure 2-16 serves as a quick reference of valid combinations of FlashCopy options. Refer to relevant sections in this chapter and in 2.1, “FlashCopy Version 1” on page 14 for information on the available options.

| | Fast Revers Restore | Revers estore | Consistency Group | Inband Command | Incremental FlashCopy | Persistent FlashCopy | Multiple Relations | Background No Copy | Background Copy | Data set FlashCopy |
|---|---|---|---|---|---|---|---|---|---|---|
| Volume FlashCopy | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | NA |
| Data set FlashCopy | ✗ | ✗ | ✗ 2 | ✗ 2 | ✗ | ✗ 2 | ✓ | ✓ | ✓ | |
| Background Copy | ✗ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | NA | | |
| Background No Copy | ✓ | ✓ | ✓ | ✓ | ✓ 1 | ✓ | ✓ | | | |
| Multiple Relations | ✓ | ✓ | ✓ | ✓ | ✓ 3 | ✓ | | | | |
| Persistent FlashCopy | ✓ | ✓ | ✓ | ✓ | ✓ | | | | | |
| Incremental FlashCopy | ✓ | ✓ | ✓ | ✓ | | | | | | |
| Inband Command | ✓ | ✓ | ✓ | | | | | | | |
| Revers Restore | NA | NA | | | | | | | | |
| Fast Reverse Restore | NA | | | | | | | | | |

*Figure 2-16 Combinations of FlashCopy options*

Figure 2-16 notes:

1. Incremental FlashCopy and no-background copy can be combined, but provides no benefit. The objective of Incremental FlashCopy is to minimize the amount of data transfer needed when a FlashCopy pair is refreshed. No savings is realized when no-background copy option is in effect. Still, you may start an incremental relation (or do an increment with no-background copy) and start the background copy later.
2. Currently, the software does not provide an interface to exploit these combinations.
3. Multiple non-incremental relationships can coexist with one incremental relationship (in other words, there can be at most one incremental relationship on a volume).

## 2.2.9 Terminating a FlashCopy relationship

Withdrawing a FlashCopy relationship differs widely between FlashCopy V1 and FlashCopy V2, with FlashCopy V2 accommodating more options.

With FlashCopy V2, all FlashCopy withdraws are considered extent withdraws. Withdraw processing, within the specified extents, removes the active FlashCopy relationship between source extents and target extents. Withdraw is also allowed for part of a relationship, resulting in the splitting of the original relationship, as illustrated in Figure 2-17.

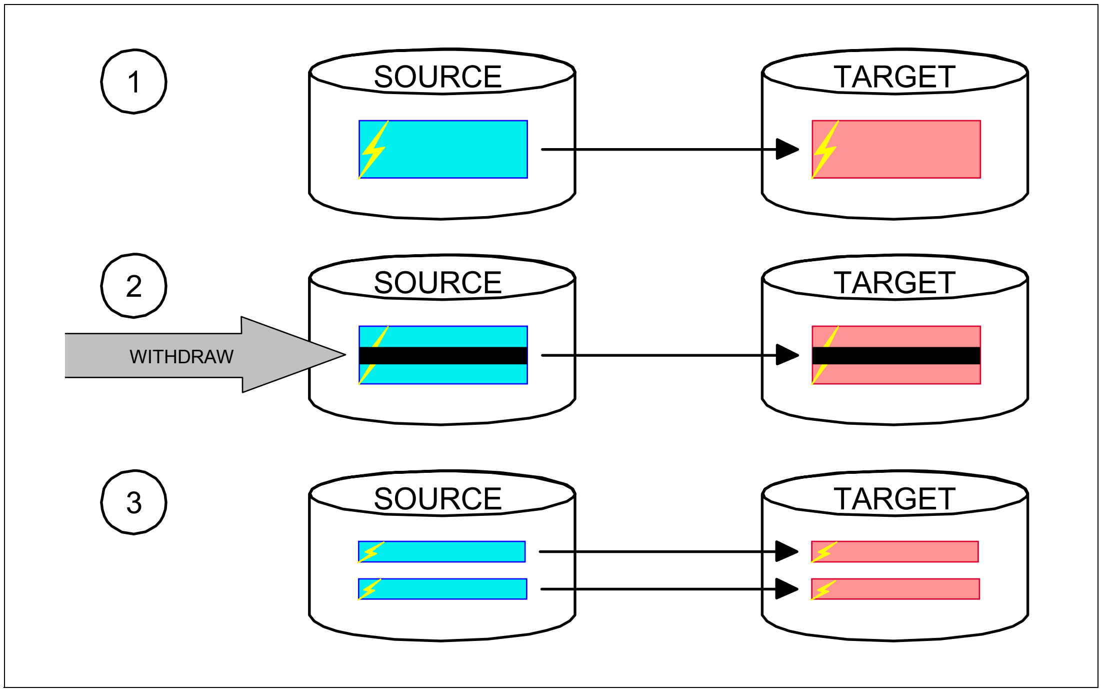



*Figure 2-17 Partial withdraw*

**Note:** While it is possible to withdraw by specifying extents, users generally work with FlashCopy at the volume level or data set level.

### Withdraw to target

Specifying the target device without the source device on a withdraw operation allows the user to withdraw all FlashCopy relationships which have target extents on the specified target volume, regardless of what source volumes are involved in those FlashCopy relationships. In this case ESS withdraw processing identifies where the source extents are (volume and track location) and removes those relationships.

If a TSO FCWITHDR command or the ANTRQST API macro is used to initiate the withdraw operation, and extents are specified on the command, only FlashCopy relationships within the specified extents are removed.

Figure 2-18 illustrates the effect of withdrawing to target device VOL003. All FlashCopy relationships with target extents in VOL003 are terminated. FlashCopy relationships with source extents in VOL003 are not affected.

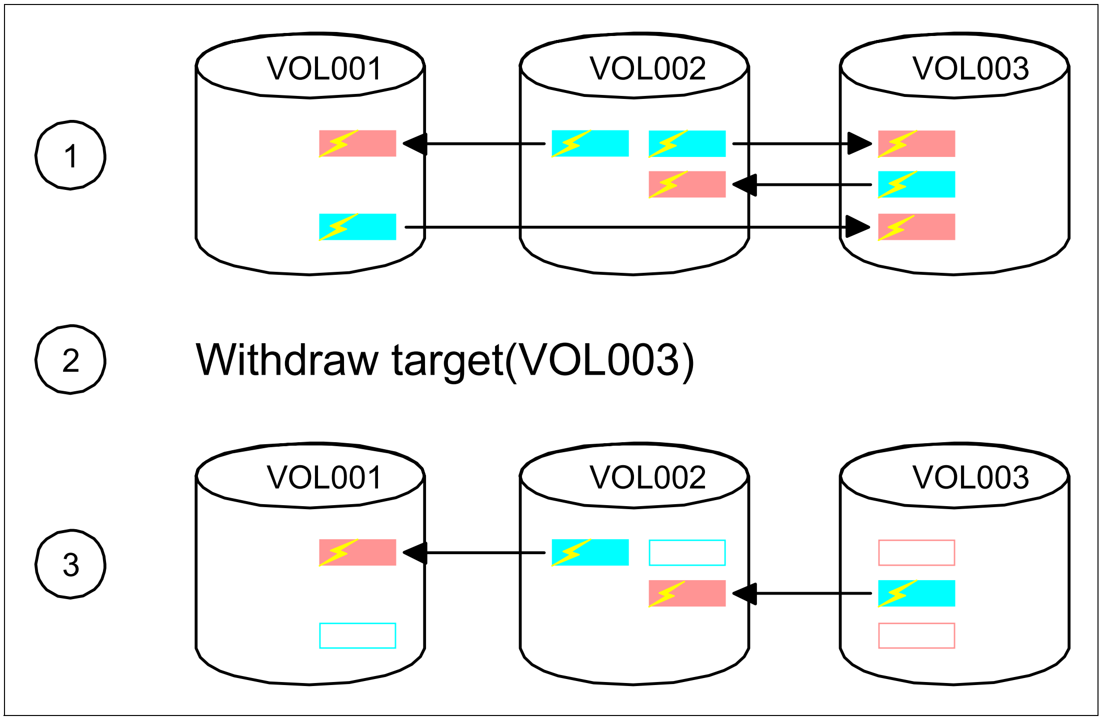



*Figure 2-18 Withdraw to target*

### Withdraw source and target

Specifying both the source and target devices allows the user to limit the scope of the withdraw to those FlashCopy relationships, which have source extents on the specified source device and corresponding target extents on the specified target device, regardless of whatever source and target relationships with other devices may exist.

Figure 2-19 illustrates the effect of withdraw specifying both source device (VOL002) and target device (VOL003). Only relationships with source extents in VOL002 and target extents in VOL003 are terminated.

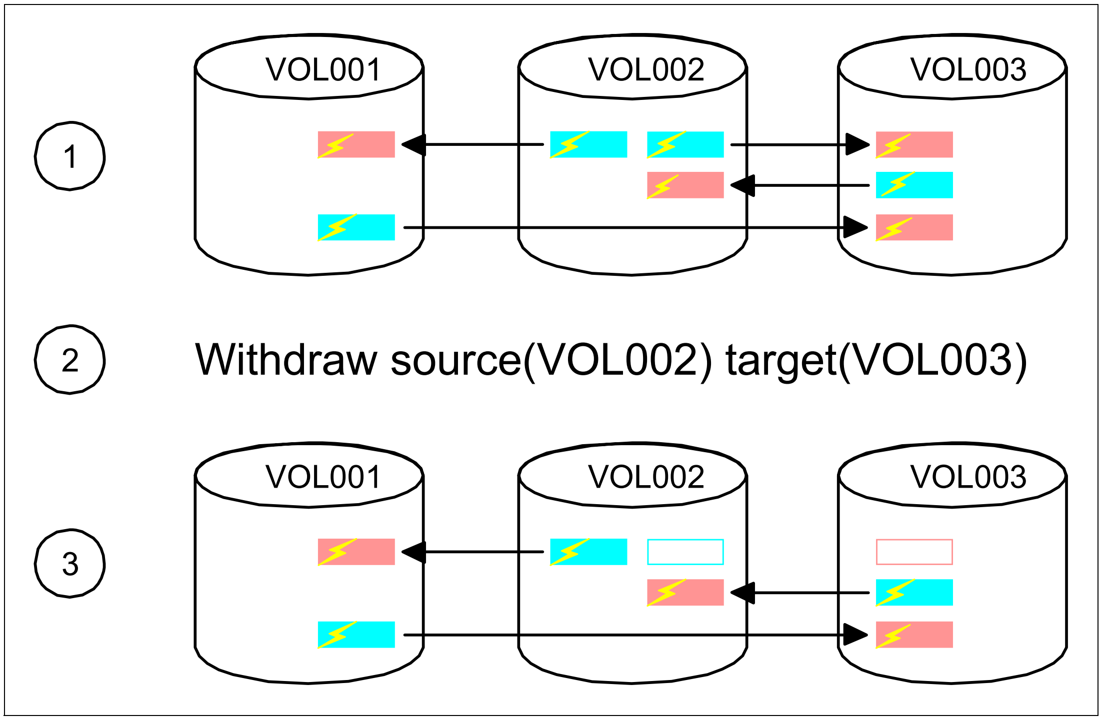


*Figure 2-19 Withdraw source and target*

## Using DDSW(YES)

The *deleted data space withdraw* (DDSW) parameter is available when the withdraw operation is invoked from a TSO command or the ANTRQST macro.

When a withdraw operation issued to a source volume with the DDSW(YES) parameter specified, the following events occur:

- All FlashCopy relationships with target extents on the specified volume are withdrawn.
- All FlashCopy relationships with source extents on the specified volume, if established with background copy, are allowed to complete. The FlashCopy relationships are removed automatically when the copy completes.
- All FlashCopy relationships with source extents on the specified volume, if established with no-background copy, are changed to background copy. This causes all source tracks to be copied to their respective target tracks, the relationships are withdrawn when the copy completes.

The **`DDSW(YES)`** parameter may be used to make sure that all relationships have been cleaned up on the source volume prior to starting a backup cycle using DFSMSdss.

Figure 2-20 illustrates the effect of a withdraw to source with DDSW(YES). There are two active FlashCopy relationships with source extents on VOL002, and one active relationship with target extents on VOL002. As a result of the withdraw:

- FlashCopy relationship VOL002->VOL001 is automatically withdrawn after background copy completes.
- FlashCopy relationship VOL002->VOL003 is changed to background copy mode and withdrawn after background copy completes.
- FlashCopy relationship VOL003->VOL002 is withdrawn immediately even if background copy has not completed.

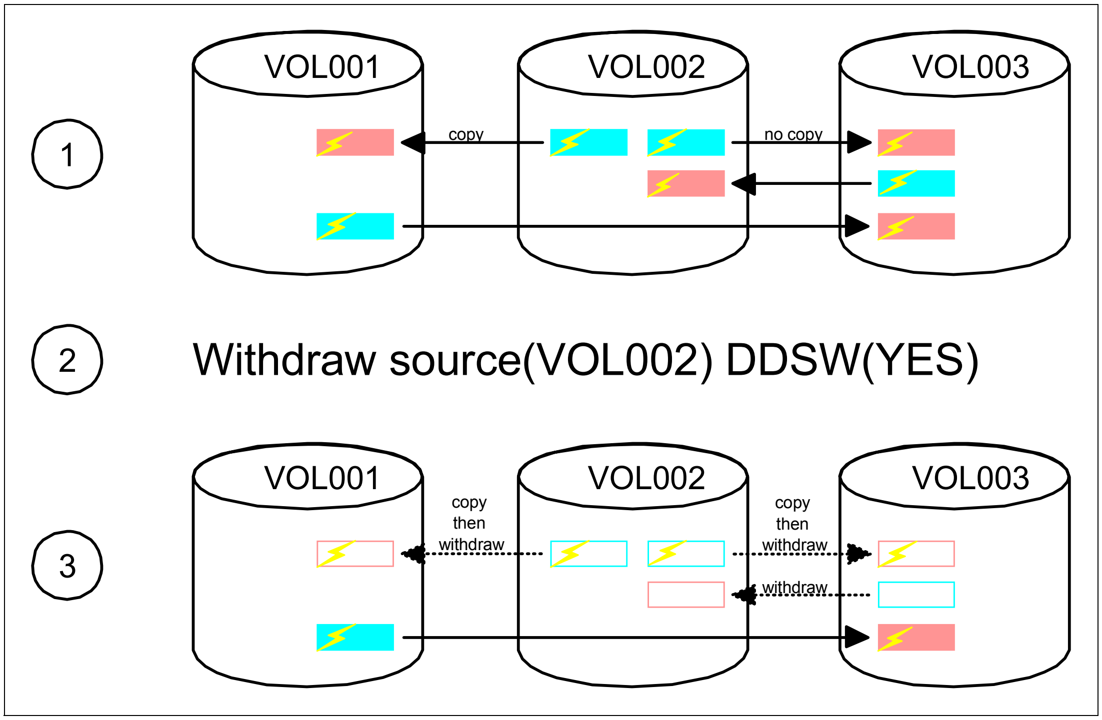



*Figure 2-20 Withdraw to source with DDSW(YES)*

For additional information about how to restrict DDSW processing to selected extents in a volume, see the publication *z/OS DFSMS Advanced Copy Services,* SC35-0428.

## 2.2.10 Invoking FlashCopy V2 new functions

For z/OS and OS/390, FlashCopy can be invoked in five different ways:

- DFSMSdss utility
- TSO commands
- ANTRQST macro
- ICKDSF
- ESS Copy Services Web User Interface

General FlashCopy invocation is discussed in 2.1.3, “How to invoke FlashCopy” on page 17. Refer to that section to complement the descriptions presented in this section.

Specifically for FlashCopy Version 2, Figure 2-21 summarizes what interfaces have options to invoke the new functions.

| FUNCTION \ INTERFACE | DFSMSdss | TSO | ANTRQST macro | Web user interface / CLI | ICKDSF |
|---|---|---|---|---|---|
| Multiple relations ① | ✓ | ✓ | ✓ | ✓ | ✓ |
| FlashCopy across LSSs & across clusters ② | ✓ | ✓ | ✓ | ✓ | ✓ |
| Incremental FlashCopy | ✗ | ✗ | ✗ | ✓ | ✓ |
| Inband commands | ✗ | ✗ | ✗ | ✓ | ✓ |
| FlashCopy consistency groups | ✗ | ✗ | ✗ | ✓ | ✗ |
| Data set FlashCopy | ✓ | ✓ ③ | ✓ ③ | ✗ | ✗ |
| FlashCopy to PPRC Primary volume | ✓ | ✗ | ✓ | ✓ | ✓ |
| Reverse Restore/ Fast Reverse Restore | ✗ | ✗ | ✗ | ✓ | ✓ |

*Figure 2-21 Invoking FlashCopy functions*

Figure 2-21 notes:

1. The ability to establish multiple relations over an extent is transparent to users, that is, there is no keyword associated with this function.
2. The ability to establish FlashCopy relationships across LSSs and across clusters is transparent to users — it is just a matter of choosing a different source and target LSS.
3. Extents can be specified on TSO commands and ANTRQST macro calls, but the VTOC and the catalogs are not updated.

In the following sections we describe the options and parameters, which are used for invoking the new functions that FlashCopy Version 2 provides.

## 2.2.11 DFSMSdss utility

This section discusses how DFSMSdss uses FlashCopy for volume copies and data set copies. For detailed information on DFSMSdss, refer to the IBM publication *z/OS DFSMSdss Storage Administration Reference,* SC35-0424.

### Full volume copy

The DFSMSdss COPY FULL command can use FlashCopy during a full volume copy if the following requirements are met:

- The source and target are on an ESS with the optional FlashCopy feature enabled.
- The volumes are in the same ESS.
- The FASTREPLICATION(NONE) keyword must ***not*** be specified.

Not all tracks on the volume are copied when DFSMSdss invokes FlashCopy for a full volume copy — DFSMSdss requests FlashCopy for allocated extents only. Refer to “DFSMSdss copy volume” on page 32 for an example of DFSMSdss full volume copy using FlashCopy.

In order to balance between excluding free space and saving the number of FlashCopy relationships, up to 255 relations will be created for each full volume copy. If there are more than 255 extents on the volume, extents will be merged (to reduce the number of extents) resulting in some free space being copied.

For the best performance during COPY FULL operations, specify the following keywords:

- `ADMINISTRATOR` — Allows a DFSMSdss-authorized storage administrator to bypass access checking to data sets and catalogs.
- `ALLDATA(*)` — Specifies for all data sets whose data set organization is PS (physical sequential), PSU (physical sequential unmovable), PO (partitioned organization), POU (partitioned organization unmovable), or null, that all of the allocated space for the source data set is copied to the target volume.
- `ALLEXCP` — Specifies for all data sets whose data set organization is PS, PSU, PO, POU, or null and are empty, which all of the allocated space for the source data set is copied to the target volume.
- `PURGE` — Specifies that unexpired data sets, which reside on the target volume, can be overlaid for a full or track copy operation.

## Data set copy

The DFSMSdss COPY DATASET command can use FlashCopy for data set copy when the following conditions are met:

- The source and target device types are the same.
- The source devices and the target devices are in the same ESS.
- The FlashCopy Version 2 optional feature is enabled.
- The FASTREPLICATION(NONE) keyword is *not* specified.
- The following types of processing are *not* required:
  - **Reblocking:** Reblocking occurs when you specify the REBLOCK keyword or when the VTOC indicates that the data set can be reblocked.
  - **PDS compression:** DFSMSdss compresses a PDS data set during copy processing, by default. You can specify the NOPACKING keyword to prevent DFSMSdss from compressing the PDS, thereby allowing the use of FlashCopy.
  - **Changing stripe counts:** The source stripe count must be the same as the target stripe count for a striped extended format data set.
  - **An individual stripe extending to more than one volume:** A single-striped sequential-extended format data set cannot use FlashCopy if either the source data set or the target data set is multivolume.
  - **PDS or PDSE conversion:** Conversion occurs when you specify the CONVERT keyword with these data sets.
  - **Block-by-block processing of direct access data sets:** Block-by-block processing occurs when you specify the REBLOCKADDRESS or the AUTOREBLOCKADDRESS keyword.
  - **Utilities:** FlashCopy cannot be used if your data must be moved with the use of a utility. For example, if DFSMSdss invokes a utility such as IDCAMS REPRO for a data set copy operation, FlashCopy is not used for those data sets. Table 2-2 shows in which situations DFSMSdss invokes utilities for data set copy operations.

*Table 2-2 Data mover for data set copy (to device of same geometry)*

| Data set type | Data mover | Notes |
|---|---|---|
| Sequential | DFSMSdss | |
| Partitioned (not PDSE) | DFSMSdss / IEBCOPY | All partitioned data sets that are not load modules are compressed during a copy. Specify NOPACKING if FlashCopy is to be used. |

| Data set type | Data mover | Notes |
|---|---|---|
| Partitioned (not PDSE) load modules | DFSMSdss / IEBCOPY | If copying partitioned load modules with REBLOCK, DFSMSdss calls IEBCOPY to copy the data set to a like device. |
| Partitioned data set extended (PDSE) | DFSMSdss / IGWFAMS | DFSMSdss calls the IGWFAMS utility when you are converting a PDS to a PDSE. |
| Direct nonrelative block address mode | DFSMSdss | |
| Direct relative block address mode | DFSMSdss | Specify the DFSMSdss RELBLOCKADDRESS parameter. |
| Indexed sequential to same track. Target volume uses VTOC index. Target space available. | DFSMSdss | |
| Indexed sequential (all other cases) | IEBISAM | |
| ESDS | DFSMSdss / IDCAMS | DFSMSdss calls IDCAMS if the target CISIZE, CASIZE, physical record size, or physical block size of the target is different from that of the source. |
| RRDS | DFSMSdss / IDCAMS | DFSMSdss calls IDCAMS if the target CISIZE, CASIZE, physical record size, or physical block size of the target is different from that of the source. |
| LDS | DFSMSdss/IDCAMS | DFSMSdss calls IDCAMS if the target CISIZE, CASIZE, physical record size, or physical block size of the target is different from that of the source. |
| KSDS or VRRDS | DFSMSdss / IDCAMS | DFSMSdss calls IDCAMS if any of the following is true:<br>► The CISIZE, CASIZE, physical record size, physical block size, imbed, or span attributes of the target are different from that of the source.<br>► The target data set is SMS and has an imbedded index or has key ranges, and the target volume count is greater than one.<br>► The target data set is non-SMS, the source component or components span multiple volumes, and there is not enough space on one target volume to contain the entire data set. |
| Key range data set | DFSMSdss / IDCAMS | DFSMSdss calls IDCAMS if the source and target CASIZE, physical record size, or physical block size are different; if the components span multiple volumes; for a KSDS with IMBED and either the source HURBA=HARBA or it has extended indexes. |
| Extended-format VSAM | DFSMSdss / IDCAMS | DFSMSdss calls IDCAMS if the target CISIZE, CASIZE, physical record size, or physical block size of the target is different from that of the source. |
| Integrated catalog facility user catalogs | IDCAMS (EXPORT/IMPORT) | |
| Undefined DSORG | DFSMSdss | |

### *Copy data set job example*

Example 2-11 shows a COPY DATASET operation against a partitioned data set. The NOPACKING option specifies that the data set is not to be compressed during copy, this allows DFSMSdss to invoke FlashCopy for the copy operation. Two copies of SRC.PDS are requested; two FlashCopy relationships will be created from the same source data sets.

*Example 2-11 Copy partitioned data set*

```
//DSSCOPY JOB .....
//STEP1    EXEC PGM=ADRDSSU
//SYSPRINT  DD  SYSOUT=*
//OUTDD     DD  UNIT=3390,VOL=SER=RS1510,DISP=SHR
//SYSIN     DD  *
  COPY DS(INCL(SRC.PDS))                         -
       OUTDD(OUTDD)                              -
       NOPACKING(SRC.PDS)                        -
       RENAMEU((SRC.**,TRG.**))                  -
       CATALOG
  COPY DS(INCL(SRC.PDS))                         -
       OUTDD(OUTDD)                              -
       NOPACKING(SRC.PDS)                        -
       RENAMEU((SRC.**,NEW.**))                  -
       CATALOG
/*
```

With SMS managed data, DFSMSdss attempts to allocate the target data set on the same device type in the same ESS, if the source is in an ESS. This increases the probability that FlashCopy can be used to copy the data. However, FlashCopy cannot be used if the source data set is multivolume and is not contained entirely in one ESS. These data sets will not be processed with FlashCopy and will be allocated to whatever volumes are available, irrespective of their FlashCopy capability.

## FCNOCOPY and FCWITHDRAW parameters

The FCNOCOPY parameter (for the COPY command) and FCWITHDRAW parameter (for the DUMP command) provide the capability of using FlashCopy with the no-background copy option during a DFSMSdss COPY operation and to subsequently withdraw the FlashCopy relationship at the completion of a DFSMSdss DUMP operation.

The FCNOCOPY parameter is enhanced for use with COPY DATASET and COPY TRACKS operations, in addition to COPY FULL operations.

Similarly, the FCWITHDRAW parameter is enhanced for use with DUMP TRACKS and DUMP DATASET (physical and logical) operations, in addition to DUMP FULL operations.

Example 2-12 shows an example using COPY DATASET FCNOCOPY and DUMP DATASET FCWITHDRAW commands. Step1 uses FlashCopy to copy the data set SRC.DATA to the target data set TGT.DATA, no-background copy is specified with the FCNOCOPY keyword. The DUMP DATASET command with FCWITHDRAW keyword withdraws the FlashCopy relationship when the subsequent dump operation is complete.

*Example 2-12 Data set copy and dump job stream - FCNOCOPY and FCWITHDRAW*

```
//DSNOCOPY JOB .....
//*
//* STEP1 - COPY DATASET WITH FCNOCOPY
//*
//STEP1    EXEC PGM=ADRDSSU
//SYSPRINT  DD  SYSOUT=*
```

```
//OUTDD     DD  UNIT=3390,VOL=SER=RS1510,DISP=SHR
//SYSIN     DD  *
  COPY DS(INCL(SRC.DATA))                         -
       OUTDD(OUTDD)                               -
       RENAMEU((SRC.**,TGT.**))                   -
       FCNOCOPY                                   -
       CATALOG
/*
//*
//* STEP2 - DUMP DATASET WITH FCWITHDRAW
//*
//STEP2    EXEC PGM=ADRDSSU
//SYSPRINT  DD  SYSOUT=*
//OUTDD     DD  DSN=BACKUP.TGT.DATA,DISP=(,KEEP),LABEL=(1,SL),
//          UNIT=3490,VOL=SER=TAPE01
//SYSIN     DD  *
  DUMP DS(INCL(TGT.DATA))                         -
       OUTDD(OUTDD)                               -
       FCWITHDRAW
/*
```

## DEFRAG command

Because of the nature of allocation algorithms and the frequent creation, extension, and deletion of data sets, free space on DASD volumes becomes fragmented. This results in:

- Inefficient use of DASD storage space
- An increase in space-related abends (abnormal endings)

The DFSMSdss DEFRAG command relocates data set extents on a DASD volume to reduce or eliminate free-space fragmentation. DFSMSdss can use FlashCopy during a DEFRAG operation if the device is in an ESS that supports data set FlashCopy. DEFRAG will use FlashCopy to move extents if conditions are met. Example 2-13 illustrates a DEFRAG operation issued against volume RS1510.

*Example 2-13 DFSMSdss DEFRAG*

```
//DEFRAG JOB .....
//STEP1    EXEC PGM=ADRDSSU
//SYSPRINT  DD  SYSOUT=*
//INDD      DD  UNIT=3390,VOL=SER=RS1510,DISP=SHR
//SYSIN     DD  *
  DEFRAG DDNAME(INDD)
/*
```

## FASTREPLICATION parameter

The FASTREPLICATION parameter on COPY and DEFRAG commands specifies whether the use of fast replication methods is required, preferred, or not desired. Fast replication methods are FlashCopy in the ESS, and SnapShot in the RVA. It does not affect Concurrent Copy or virtual Concurrent Copy processing.

**FASTREPLICATION(PREFERRED)** specifies that you want DFSMSdss to use FlashCopy before any other method to move data. If FlashCopy cannot be used and you have specified the CONCURRENT keyword, DFSMSdss attempts to use Concurrent Copy. If you have not specified the CONCURRENT keyword or if Concurrent Copy fails, DFSMSdss uses traditional data movement methods to copy the data. FASTREPLICATION(PREFERRED) is the default.

**FASTREPLICATION(REQUIRED)** specifies that fast replication must be used. For full volume or track processing, DFSMSdss fails the operation if fast replication cannot be used. For data set processing when more than one data set is copied, DFSMSdss stops processing the current data set if fast replication cannot be used, but continues processing the rest of the data sets using fast replication. If FlashCopy cannot be used, DFSMSdss issues error message ADR938E, which indicates the processing of the current data set or that the entire COPY task failed.

**FASTREPLICATION(NONE)** specifies that DFSMSdss will not attempt to use FlashCopy to copy data.

### DEBUG(FRMSG(..)) parameter

The optional DEBUG(FRMSG(..)) parameter on the COPY and DEFRAG commands lets the user designate the amount of diagnostic information that DFSMSdss provides when a fast replication method such as FlashCopy or SnapShot cannot be used. This parameter overrides the optional DEBUG=FRMSG parameter, which can be specified in the JCL EXEC statement.

**DEBUG(FRMSG(MINIMAL))** specifies that DFSMSdss issue an informational message with a minimal level of information about why DFSMSdss could not use a fast replication method.

**DEBUG(FRMSG(SUMMARIZED))** specifies that DFSMSdss issue an informational message with summarized information that explains why a fast replication method could not be used. When applicable, summarized information regarding ineligible volumes is provided in the message text. DEBUG(FRMSG(SUMMARIZED)) is equivalent to DEBUG=FRMSG in the JCL EXEC statement.

**DEBUG(FRMSG(DETAILED))** specifies that DFSMSdss issue an informational message with details for why a fast replication method could not be used. When applicable, detailed information regarding ineligible volumes is provided in the message text.

## 2.2.12 TSO commands

This section discusses the TSO commands changed and the new parameters, for using the new FlashCopy V2 functions. The information presented in this section should be complemented with the information in 2.1.5, “TSO commands” on page 22.

For detailed information on FlashCopy TSO commands, refer to the IBM publication *z/OS DFSMS Advanced Copy Services,* SC35-0428.

### FCESTABL command

The FCESTABL TSO command is used to establish a FlashCopy relationship between the source and target devices. Example 2-14 establishes a FlashCopy relationship from source device 9400 to target device 9470 with background copy.

*Example 2-14 FCESTABL command*

```
FCESTABL SDEVN(9400) TDEVN(9470) MODE(COPY)
```

#### *Changed parameters:*

**SDEVN** Specifies the four-digit hexadecimal device number of the source device. This number can be entered without the ‘X’ designation.

**TDEVN** Specifies the four-digit hexadecimal device number of the target device. This number can be entered without the ‘X’ designation.

**EXTENTS** Specifies the source volume begin and end track extents that are copied to the target volume begin and end track extents at the same location. This allows for a greater number of extents to be specified. You can specify as many extents as allowed by the TSO command line limitation.

### *New parameters:*

**XTNTLST** Specifies a list of source and target extent pairs that are in individual FlashCopy relationships. You can specify as many extent pairs as allowed by the TSO command line limitation. Each extent pair contains the source volume begin and end track extents, and the corresponding target volumes begin and end track extents.

Specify either EXTENTS or XTNTLST parameter, not both. If neither is specified, the FlashCopy FCESTABL command defaults to all tracks of the source and target volumes (full volume). See Example 2-15.

*Example 2-15 FCESTABL commands with EXTENTS / XTNTLST*

```
FCESTABL SDEVN(9400) TDEVN(9470) MODE(NOCOPY)
     EXTENTS (01000004 0357000A, 12AC0006 12FF000C)
FCESTABL SDEVN(9400) TDEVN(9470) MODE(NOCOPY)
     XTNTLST (01000004 0357000A 13000001 15570007, 02AC0006 02FF000C 18A00000 18F30006)
```

In Example 2-15 the first command establishes two FlashCopy relationships as follows:

1. Between the source device 9400 extent range 01000004-0357000A and the offline target device 9470 at the same location, that is extent range 01000004-0357000A.
2. Between the source device 9400 extent range 12AC0006-12FF000C and the offline target device 9470 at the same location, that is extent range 12AC0006-12FF000C.

The second command establishes two FlashCopy relationships as follows:

1. Between the source device 9400 extent range 01000004-0357000A and the offline target device 9470 extent range 13000001-15570007.
2. Between the source device 9400 extent range 02AC0006-02FF000C and the offline target device 9470 extent range 18A00000-18F30006.

For both commands, no background copy is initiated.

## FCQUERY command

The FCQUERY TSO command is used to determine the copy status of an ESS device available to the host system. Example 2-16 queries the status of device 9400.

*Example 2-16 FCQUERY command*

```
FCQUERY DEVN(9400)
```

### *Changed parameters:*

DEVN Specifies the four-digit hexadecimal device number of the device. This number can be entered without the ‘X’ designation.

Figure 2-22 shows the FCQUERY command output for FlashCopy V2.

```
ANTF0420I FCQUERY Formatted
 DEVN SSID LSS CCA  CU     SERIAL       ACT    MAX XC PC CC
 1415 1134  04  15 2105 000000018767      0   2719  N  N  N
```

*Figure 2-22 FCQUERY command output*

This new format reports how many relations a volume has, not which data sets are in a relationship. The Copy Services status of the volume is shown under the respective columns:

- `ACT` — Shows the number of active FlashCopy relationships (extent) on the volume. This refers to the number of track extents that are in FlashCopy relationships. For example, if a FlashCopy establish command were issued, requesting FlashCopy for a data set with 5 extents, the output of FCQUERY will show ACT=5.
- `MAX` — Shows the maximum number of FlashCopy relationships (extent) allowed on the volume.
- `XC` — Indicates whether the volume is in an XRC pair.
- `PC` — Indicates whether the volume is in a PPRC pair.
- `CC` — Indicates whether the volume is in Concurrent Copy mode.

## FCWITHDR command

The FCWITHDR TSO command is used to end a FlashCopy relationship. The command has been changed to accommodate more withdraw options.

Example 2-6 shows a situation where all the FlashCopy relationships with target extents in device number 9470 are withdrawn.

*Example 2-17 FCWITHDR command*

```
FCWITHDR TDEVN(9470)
```

### *Changed parameters:*

**SDEVN** Specifies the four-digit hexadecimal device number of the source device. This number can be entered without the 'X' designation. If SDEVN is specified, TDEVN must also be specified, unless DDSW(YES) is specified.

**TDEVN** Specifies the four-digit hexadecimal device number of the target device. This number can be entered without the 'X' designation. TDEVN is required except when SDEVN is specified with DDSW(YES).

### *New parameters:*

**DDSW** Specifies whether you want the deleted *data space withdraw* (DDSW) function to be performed on the device that is specified with the SDEVN parameter.
DDSW(YES) specifies to withdraw all eligible relationships on the device specified with the SDEVN parameter. Refer to , "Using DDSW(YES)" on page 44 for details on this parameter.
DDSW(NO) specifies that withdraw processing be performed without deleted data space withdraw considerations. This is the default.

**XTNTLST** Specifies that FlashCopy withdraw applies to the ranges of source and target extents specified within the list. If not specified, the FlashCopy FCWITHDR command defaults to processing all tracks of the specified devices.

*Example 2-18 FCWITHDR command with XTNTLST*

```
FCWITHDR SDEVN(9400) TDEVN(9470)
     XTNTLST(00000000 0100000E 20000000 2100000E)
```

The command shown in Example 2-18 withdraws all FlashCopy relationships with source tracks on device 9400 in extent range 00000000-0100000E and corresponding target tracks on device 9470 in extent range 20000000-2100000E.

For any other situation other than a request syntax problem (that is, parameters in error), the FCWITHDR request will be reported as successful regardless of the status of the specified tracks.

### 2.2.13 ANTRQST API interface

The QFRVOLS request of the ANTRQST macro can be used to determine if volumes are eligible to participate in a fast replication operation. The QFRVOLS query returns information about the type of fast replication capability for each volume or extent in a user-supplied volume list. If a fast replication operation cannot be performed, the QFRVOLS query information indicates why the volumes or extents are not eligible for fast replication operations.

In addition, the SCREXTENTS and TGTEXTENTS parameters of the FCESTABLISH and FCWITHDRAW requests allow target extent relocation. The specified target extent does not have to match the corresponding source extent. DDSW(YES | NO) has also been added to the FCWITHDRAW request, which allows users to specify if the deleted data space withdraw function is required.

For more information on the ANTRQST macro, see the IBM publication *z/OS DFSMSdfp Advanced Services,* SC26-7400.

### 2.2.14 ESS Copy Services Web User Interface

Some FlashCopy V2 functions, such as Incremental FlashCopy, FlashCopy Consistency Group, and inband FlashCopy commands, can only be invoked using the ESS Copy Services Web User Interface (WUI).

This section discusses the options available with the ESS Copy Services Web User Interface in support of the FlashCopy V2 functions and presents examples on how to invoke them.

When planning to use the ESS Copy Services Web User Interface for FlashCopy operation, refer to the following publication: *IBM TotalStorage Enterprise Storage Server Web Interface User's Guide,* SC26-7448.

### FlashCopy copy options

Figure 2-23 shows all the copy options (old and new) now available when a FlashCopy relationship is established through the ESS Copy Services Web User Interface.

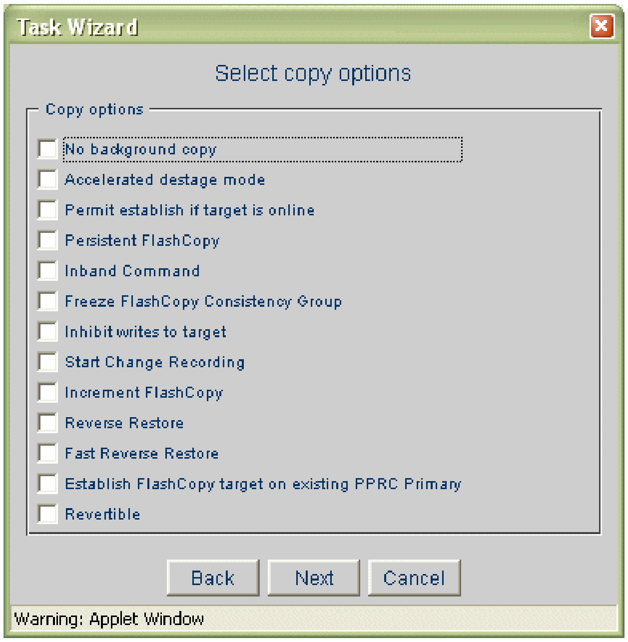



*Figure 2-23 Select copy options*

Here is an explanation of the copy options shown in Figure 2-23:

- **No background copy:** Select this option if you do not want a background copy task to be executed.
- **Accelerated destage mode:** Select this option to cause a FlashCopy source volume track which is being modified to be destaged from cache sooner than it would be if normal cache algorithms were applied. This feature minimizes (for the source volume) the number of modified tracks that are resident in cache.

  With FlashCopy V1, this option might (under specific circumstances) speed up the completion of the background copy process. This feature has no effect with FlashCopy V2 as the cache algorithms have changed.
- **Permit establish if target is online:** Select this option to allow a FlashCopy relationship to be established even if the target is online to the zSeries host.
- **Persistent FlashCopy:** Select this option if you want the FlashCopy relationship to remain even after the FlashCopy operation completes. The FlashCopy relationship between source and target volumes remains until it is explicitly withdrawn. See “Persistent FlashCopy relationship” on page 16 for information.
- **Inband Command:** Select this option to allow a FlashCopy relationship between two volumes to be established at a remote ESS using inband commands. The FlashCopy source volume selected must be the secondary PPRC volume of the PPRC pair. See 2.2.7, “Inband command over PPRC links” on page 40 for more information.
- **Freeze FlashCopy Consistency Group:** Select this option to freeze the write activity to a FlashCopy source volume. See 2.2.6, “FlashCopy Consistency Group” on page 38 for more information.
- **Inhibit writes to target:** Select this option to ensure that writes are inhibited on the target volume until an Incremental FlashCopy operation is complete.

- **Start Change Recording:** Select this option to activate change recording on the volume pair participating in a FlashCopy relationship. This enables a subsequent incremental copy.
- **Increment FlashCopy:** Select this option to create a new point-in-time copy of the data without copying all tracks from the source volume to the target volume. See 2.2.5, "Incremental FlashCopy" on page 36 for more information.
- **Reverse Restore:** Select this option to reverse the FlashCopy relationship and copy over modified tracks from the target volume to the source volume. The background copy process must complete before you can reverse the order of the FlashCopy relationship to its original source and target relationship. See 2.2.5, "Incremental FlashCopy" on page 36 for more information.
- **Fast Reverse Restore:** This option is used with Asynchronous PPRC. If you specify this option, you can reverse the FlashCopy relationship without waiting for the finish of the background copy of the previous FlashCopy.
- **Establish FlashCopy on existing PPRC Primary:** This option allows you to establish a FlashCopy pair with the target volume on a PPRC primary volume. For more information see "FlashCopy to PPRC primary volumes" on page 64.
- **Revertible:** This option is included on this panel for compliance with Asynchronous PPRC only. It is not a standalone option and should not be selected by the user. If you wish to use this option, call your IBM representative for technical support.

## Incremental FlashCopy

Incremental FlashCopy provides the capability to refresh a FlashCopy relationship. With Incremental FlashCopy, the initial relationship between a source and target is maintained. When a subsequent FlashCopy establish is initiated, only the data required to bring the target to the source's newly established point-in-time is copied.

Refer to the discussion in 2.2.5, "Incremental FlashCopy" on page 36 for an explanation of Incremental FlashCopy.

### *Initial establish*

Before an Incremental FlashCopy can be performed, a FlashCopy relationship with the **Start Change Recording** option enabled must be established (selected), as shown in Figure 2-24. The **Persistent FlashCopy** option is automatically enabled when change recording is in effect, so the relationship will remain until it is explicitly withdrawn.

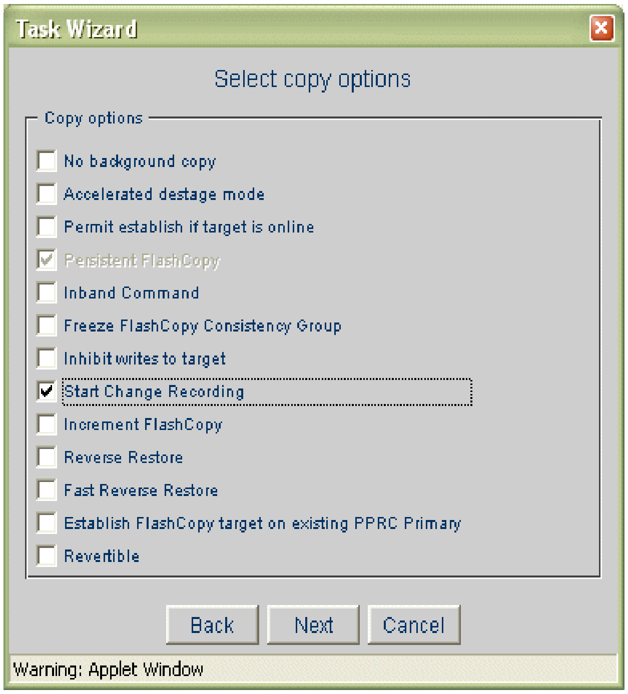



*Figure 2-24 FlashCopy with change recording*

Once the FlashCopy relationship is established the point-in-time copy is ready, and the background copy starts. While application updates may proceed on both the source and target volumes, FlashCopy will keep track of them because the *Change Recording* option has been enabled.

### *Incremental FlashCopy*

When you need to refresh the target copy (get a more recent point-in-time copy), you request an Incremental FlashCopy by selecting the **Increment FlashCopy** option (see Figure 2-25) on the establish operation. The Incremental FlashCopy operation copies only the out-of-sync tracks to the target volume because change recording was maintained for the volume pair.

When you perform an Incremental FlashCopy operation, change recording is not re-enabled by default. You must re-select the **Start Change Recording** option each time you establish an Incremental FlashCopy operation to ensure that change recording is maintained for the volume pair.

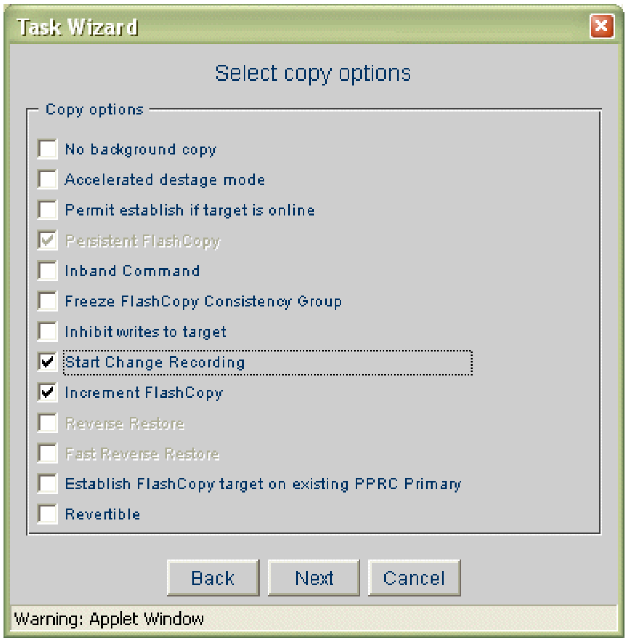



*Figure 2-25 Specifying Incremental FlashCopy option*

### *Reverse Restore*

At times you may wish to copy the target volume image onto the source volume. For example, if the target volume has not been changed since the last FlashCopy, it may be used to restore the source volume to the previous point-in-time state. This can be achieved using Reverse Restore to copy only the out-of-sync tracks from the target volume to the source volume.

Reverse Restore can only be performed after the background copy has completed. To initiate a Reverse Restore operation, establish FlashCopy between the source and target volume, select the **Reverse Restore** option as shown in Figure 2-26. If you wish to maintain change recording, the **Start Change Recording** option must be checked.

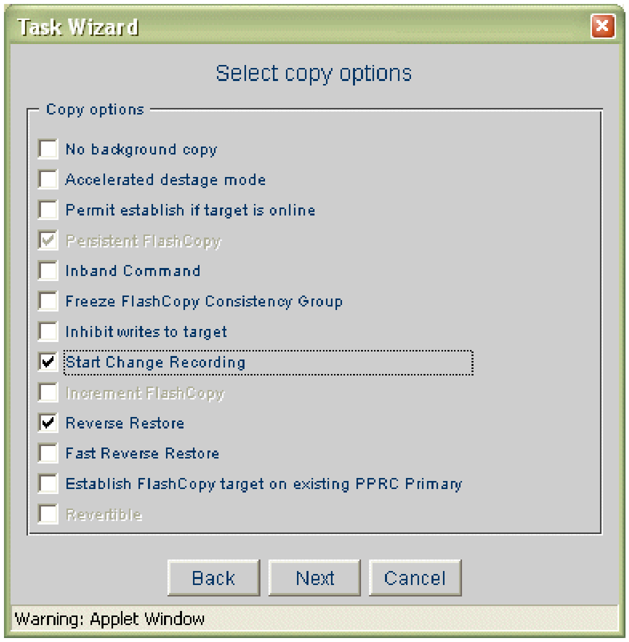



*Figure 2-26 Specifying Reverse Restore option*

After the Reverse Restore operation, the status of the initial FlashCopy pair is reversed; that is, the original source volume is now the target, while the original target volume becomes the source of the FlashCopy relation. This situation remains until another Reverse Restore operation is performed.

### Consistency Group

A Consistency Group is a group of volumes participating in FlashCopy relationships that need to keep the consistency of the logically related data that spans across them. FlashCopy Consistency Groups can be used to get a consistent point-in-time copy for an application or a database spanning multiple volumes.

To use the FlashCopy Consistency Group function, you should first define tasks to establish the FlashCopy pairs, and then define tasks to resume I/O activity after the Consistency Group is created. The defined tasks can be executed from the ESS Copy Services Web User Interface or from Command Line Interface (CLI).

### Defining tasks

A FlashCopy Consistency Group can span multiple volumes, multiple LSSs, even multiple ESSs. The tasks needed to create the Consistency Group, and the tasks needed to release the volumes I/O activity after the Consistency Group creation, should all be pre-defined.

To define the FlashCopy Consistency Group creation task, follow these steps:

1. In the ESS Copy Services Volumes panel, select the FlashCopy source volume.
2. Then right-click to select the corresponding FlashCopy target volume.
3. Right-click again to bring up the Select Task Type panel.
4. Select the **Establish FlashCopy pair** option and click **Next**.
5. At the Select Copy Options panel, select the **Freeze FlashCopy Consistency Group** option (as shown in Figure 2-27) and click **Next**.

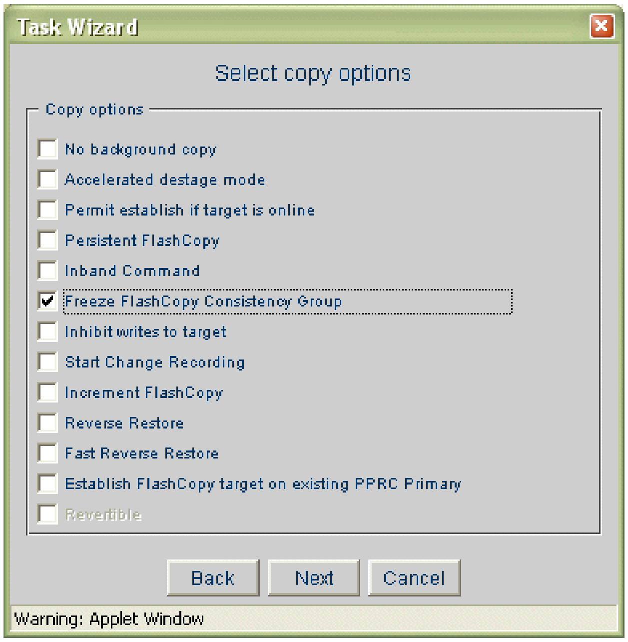



*Figure 2-27 Specifying FlashCopy Consistency Group*

6. Enter the name and description of the task at the Define Task panel, and select **Save** to save the defined task.
7. Repeat steps 1 through 6 for each source volume that will be part of the Consistency Group.
8. When the establish tasks have been created for all volumes in the Consistency Group, click **Tasks** in the navigation frame to go the ESS Copy Services Tasks panel.
9. Click to highlight the task defined in step 6.
10. Press the Ctrl key and click to highlight each task defined in step 7.
11. When all required tasks have been highlighted, click **Group** to define a composite task that comprises all highlighted tasks. We will refer to this task as the Freeze_CG_task.

To define the FlashCopy Consistency Group created task:

1. From the ESS Copy Services Logical Subsystems panel, click to select an LSS which contains source volumes from the FlashCopy Consistency Group defined.
2. Right-click the source LSS (do not select target LSS).
3. Right-click again to bring up the Select Task Type panel.
4. Select **Consistency Created** (as shown in Figure 2-28) and click **Next**.

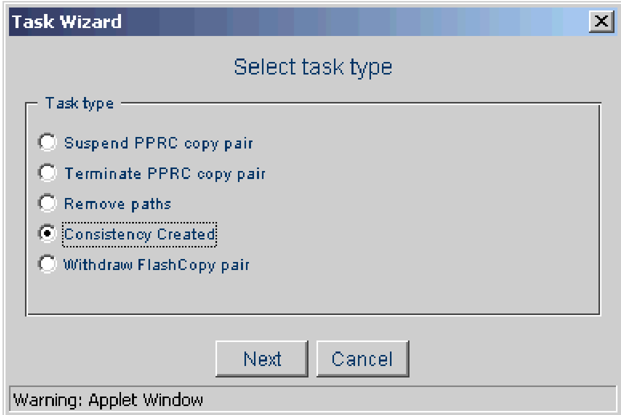



*Figure 2-28 Defining Consistency Created task*

5. From the Select Copy Options panel, select **FlashCopy Consistency Group (One LSS selected)** as shown in Figure 2-29, and click **Next**.

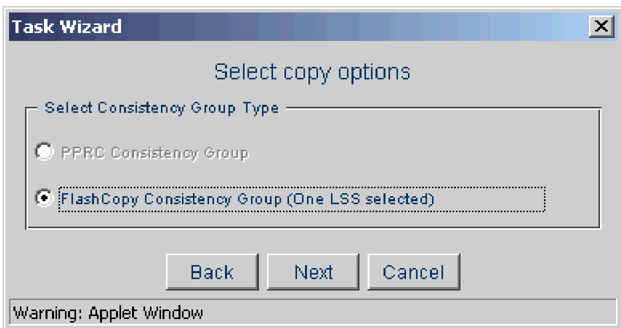



*Figure 2-29 Select Consistency Group Type*

6. Enter the name and description of the task at the Define Task panel, and select **Save** to save the defined task.
7. Repeat steps 1 through 6 for each LSS that contains source volumes in the Consistency Group.
8. When tasks have been created for all volumes in the Consistency Group, click **Tasks** in the navigation frame to go the ESS Copy Services Tasks panel.
9. Click to highlight the task defined in step 6.
10. Press the Ctrl key and click to highlight each task defined in step 7.
11. When all required tasks have been highlighted, click **Group** to define a composite task that comprises all highlighted tasks. We will refer to this task as the CG_Created_task.

> **Tip:** In the previous procedures, if you are going to select several volume pairs, you have the alternative of initially clicking the **Enter Multiple Selection Mode** button and then iterate through steps 1 and 2 until you select all the volumes. Once all the volumes have been selected you then proceed with steps 3 through 6, with the procedure ending at step 6.

### *Create consistent copy*

To create a consistent copy of all volumes in the Consistency Group, do the following steps:

1. Run the Freeze_CG_task. The freeze option in the defined task causes volumes to remain in a long-busy condition until the condition is reset or the time-out value expires (default is two minutes).
2. Run the CG_Created task. This task resets the long-busy condition, making the volumes available for normal I/O activity.

## Inband command

Inband commands are pre-defined on an ESS Copy Services server that has communication to the FlashCopy source and target volumes at the remote site. Once the task has been defined, it can be invoked from the local ESS Copy Services server even if LAN communication between the local ESS Copy Services server and the remote ESS is not available.

This is an example of a procedure to define an inband FlashCopy establish task that will be executed at a remote PPRC site:

1. In the ESS Copy Services Volumes panel, left-click to select the FlashCopy source volume.
2. Then right-click to select the FlashCopy target volume.
3. Right-click again to bring up the Select Task Type panel.
4. Select **Establish FlashCopy pair** and click **Next**.
5. At the Select Copy Options panel, select the desired copy options. On the same panel, select **Inband Command** (as shown in Figure 2-30) and click **Next**.

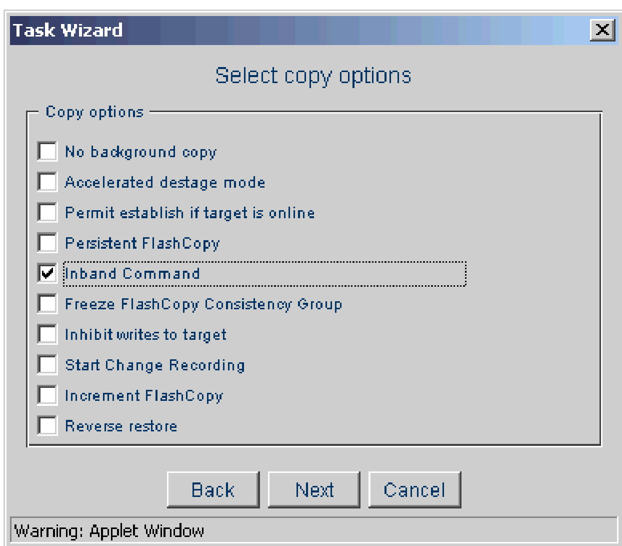



*Figure 2-30 Specifying inband command*

6. At the Choose LSS panel, select the LSS that contains the primary PPRC volume as shown in Figure 2-31.

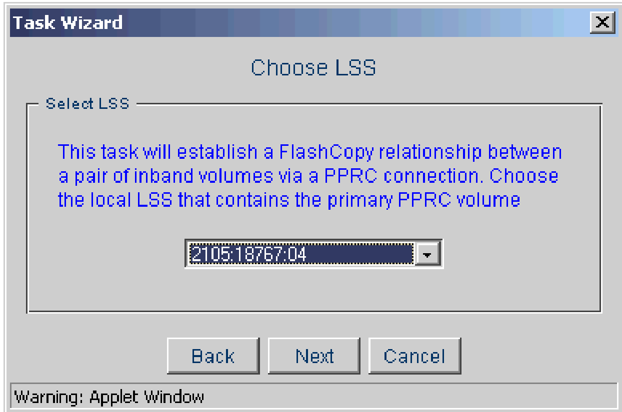



*Figure 2-31 Select LSS*

7. Enter the name and description of the task at the Define Task panel, and select **Save** to save the defined task.

To use inband commands, the FlashCopy source volume must be a PPRC secondary volume. Inband commands can be created for other FlashCopy functions, such as withdrawing a FlashCopy relationship.

## 2.2.15 Support for other zSeries operating systems

This section discusses the FlashCopy support available for VSE/ESA and VM operating environments.

### VSE/ESA support

ICKDSF now supports FlashCopy for VSE/ESA. Not all of the FlashCopy V2 functions are supported. See Figure 2-21 on page 46 to compare the FlashCopy options you can use.

### z/VM support

The CP FLASHCOPY command is enhanced to support the following FlashCopy V2 functions:

- Data set relocation: Allows relocation of source and target cylinder extents
- Multiple relationships: Allows more than one FlashCopy to be active on a volume
- Cross LSS copy: Allows source and target volumes to be on different logical controllers

There is no change to the syntax of the CP FLASHCOPY command.

The following FlashCopy Version 2 enhancements can be invoked by means of the ESS Copy Services Web User Interface:

- FlashCopy Consistency Groups
- Incremental FlashCopy

For additional information, refer to the IBM publication *z/VM CP Command and Utility Reference*, SC24-5967.

You can also use ICKDSF for FlashCopy with VM. Not all of the FlashCopy V2 functions are supported. See Figure 2-21 on page 46 to compare the FlashCopy options you can use.

## 2.2.16 FlashCopy to PPRC primary volumes

This section describes the new enhancement of having the FlashCopy target be a PPRC primary volume.

It was possible to establish a FlashCopy relation between two volumes and then create a PPRC copy pair by using the FlashCopy target as the PPRC primary volumes. But it was not permitted to use an existing PPRC primary volume as a FlashCopy target.

With ESS LIC 2.4.0 and higher this is now possible. Clients that are frequently creating new FlashCopy targets, that immediately serve as production volumes and therefore have to be mirrored by PPRC to a remote site, can do this easily now using the Web User Interface, ICKDSF or DFDSS with a new parameter. It is not yet available for TSO commands.

This ability can enable the implementation of a multi-hop remote copy solution as an alternative to other vendors' solutions.

FlashCopy can have a full or incremental point-in-time copy to an existing PPRC primary volume. FlashCopy Consistency Groups can be used, the *Nocopy* option can be specified and the relation may be persistent. Data set level FlashCopy in z/OS systems is also permitted. PPRC can run either in Synchronous or in XD mode. The PPRC primary may also be a Synchronous PPRC suspended primary. The only exception is the use of Asynchronous PPRC with this new function at the primary site.

When the FlashCopy command is submitted, and the logical copy has finished, the corresponding PPRC volume pairs turn to *duplex pending* status as long as the flashed data is being transferred to the PPRC secondary site. At this time the secondary site will be inconsistent, if a disaster occurs. Therefore clients should carefully decide if the FlashCopy target volumes can temporarily have no mirror at the remote site.

See Figure 2-32 for an example of the FlashCopy to PPRC primary.

The time it takes after the logical end of the FlashCopy until all PPRC volumes are synchronous again depends on the amount of data to be transferred and the available bandwidth to the secondary site. This time should be as short as possible and it may be important for clients as they need a full set of production data mirrored in synchronous mode.

If the parameter FULLCOPY is used for FlashCopy, the entire PPRC primary volume has to be copied to the secondary site, as if a new pair would be established.

If in contrast, Incremental FlashCopy or Data set level FlashCopy with DFSMS is used, no full volume PPRC copy is required, and only the changes have to be transferred to the PPRC secondary site. Therefore it may take less time for PPRC to be synchronized again.

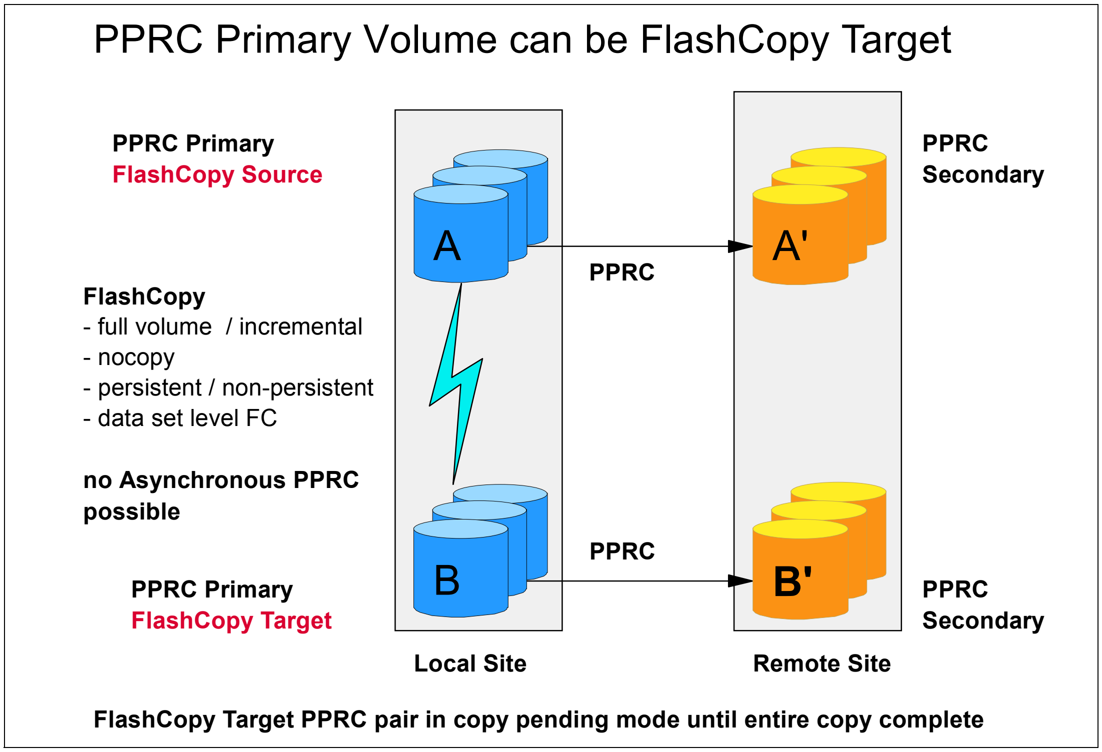



*Figure 2-32 FlashCopy to PPRC primary volumes*

DFSMSdss APAR OA06196 provides a new keyword:

```
FCTOPPRCPRIMARY or FCTOPPRCP
```

This permits the FlashCopy to the PPRC primary, otherwise traditional I/O with physical data transfer will be used. Refer to Example 2-19 and see an example of using DFDSS.

*Example 2-19 FlashCopy DFDSS job with full copy and FCTOPPRCPRIMARY*

```
//COPYFULL JOB .....
//*
//INSTIMG EXEC PGM=ADRDSSU
//SYSPRINT DD SYSOUT
//SYSUDUMP DD SYSOUT,OUTLIM=3000
//SYSIN DD *
      COPY FULL INDYNAM ((VOLA)) OUTDYNAM ((VOLB)) -
      FCTOPPRCPRIMARY
```

With ICKDSF you can use this function for full volume copies by specifying the parameter TGTOKASPPRCPRIM(YES). See Example 2-20. Data set level FlashCopies are only possible with DFDSS.

*Example 2-20 Establish FlashCopy Pair with TGTOKASPPRCPRIM(YES)*

```
   PPRC      ESTABLISH -
             DDNAME(VLB000)                     -
             SOURCEVOL(X'00',X'00',X'1070',24663) -
             TARGETVOL(X'01',X'00',3100)         -
             TGTOKASPPRCPRIM(YES) -
             CHANGERECORDING(YES) -
             MODE(NOCOPY)
```

## Using the Web User Interface

There is a new option in the Copy Services Web User Interface. After selecting the FlashCopy source, click the target that is already a PPRC primary volume. Select **Establish FlashCopy target on existing PPRC primary** and the other features of your choice and start the FlashCopy; see Figure 2-33.

You can use the WUI to start the FlashCopy to PPRC primary or to simply monitor the progress of the FlashCopy you started with ICKDSF. There are no restrictions to the options used with FlashCopy.

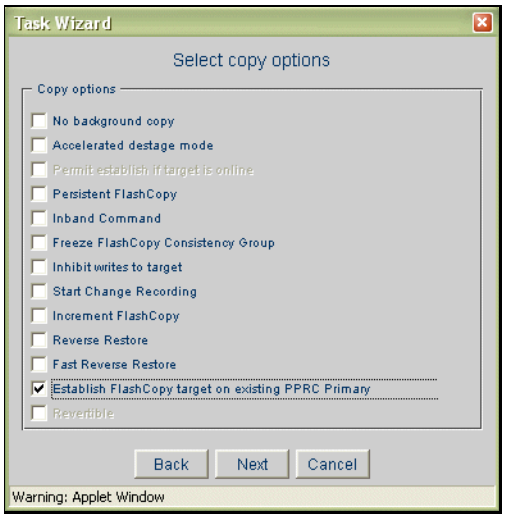



*Figure 2-33 New Option for FlashCopy*

If you start the FlashCopy to a Synchronous PPRC pair, the volumes become *duplex pending* as long as the flashed data is being received at the PPRC Target. See the status of the volumes in Figure 2-34.

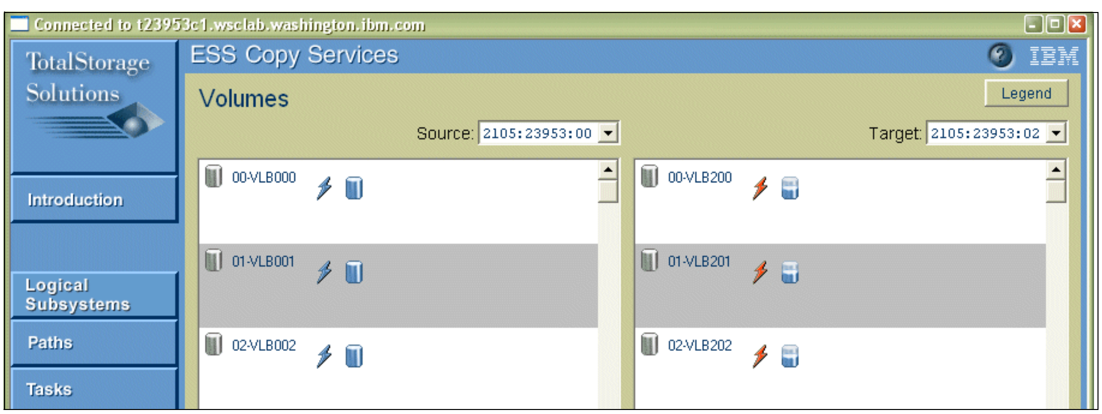



*Figure 2-34 Starting a full volume FlashCopy to a PPRC primary*

Some of the FlashCopy background tasks may end before the copied data is sent to the PPRC secondary site; see Figure 2-35.

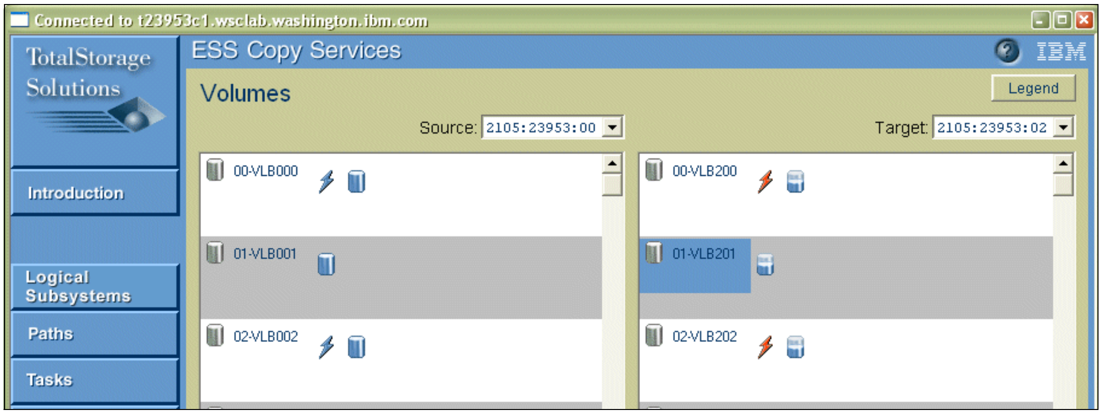



*Figure 2-35 FlashCopy process ended before Volumes synchronized*

Sometime later, other FlashCopies are still in progress while the PPRC pairs are already in synchronous mode; refer to Figure 2-36.

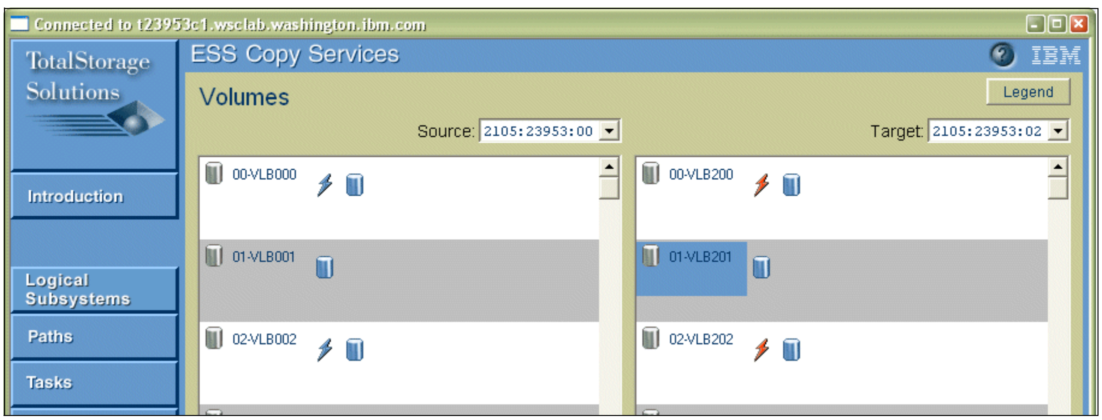



*Figure 2-36 Synchronous PPRC again before FlashCopy background process ended*

The complete full volume FlashCopy for three 3390 model 3 volumes as shown in the example took only some minutes to complete and returned the PPRC volumes to the synchronous state. Incremental FlashCopy can be used to significantly reduce the time for the copy process.

Notice that the FlashCopy of a huge amount of data may raise your workload at the PPRC links.

If the FlashCopy is done to a PPRC Extended Distance primary volume, the pair remains in *duplex pending* state as usual for PPRC-XD; see Figure 2-37.

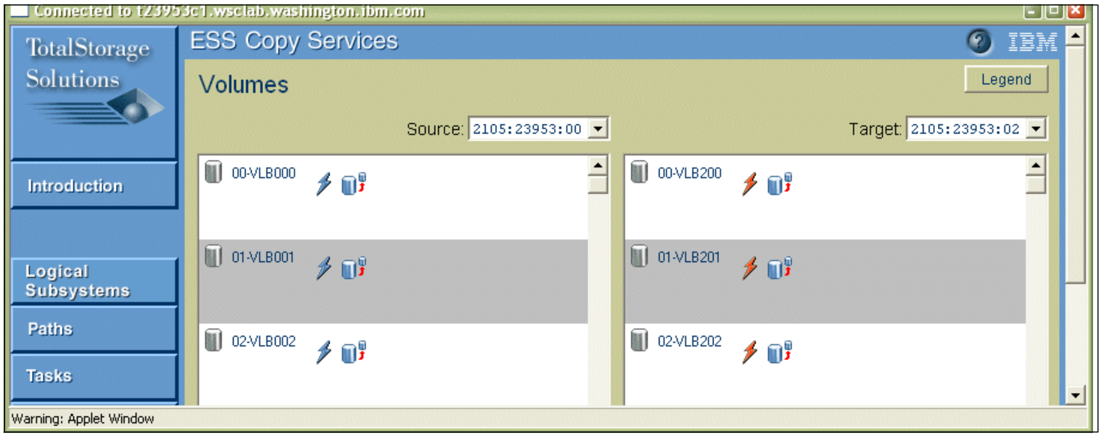


*Figure 2-37 FlashCopy to PPRC-XD primary volume*

The following components are planned to support this function:

- Multiple Device Manager Replication Manager
- CLI, API
- GDPS®, TSO, ANTRQST

Prerequisites:

- ESS Model 800, 750, ESS LIC level 2.4.0
- FlashCopy V2
- APAR OA06196 for z/OS

## 2.2.17 Final considerations

This section reviews some final considerations useful when planning the FlashCopy Version 2 implementation. The information in this section should be complemented with the general recommendations discussed in 2.1.10, “Guidelines and recommendations” on page 29.

Performance improvements in FlashCopy Version 2 provide up to a 10 times reduction in the time required to complete a FlashCopy establish command. With this significant reduction in establish time, operational interruption is further minimized and the benefits of FlashCopy can be extended to new application environments

The following are considerations when planning to use FlashCopy functions:

- In general, if you want a temporary copy of the data, specify no-background copy, and then withdraw the FlashCopy relationship when you no longer need the copy. If you want a permanent copy, do not specify no-background copy.
- With FlashCopy, TSO or API usage does not provide any data management services, such as allocation or cataloging. Data sets that are copied using these functions are not accessible from the target volume without the user manually performing these data management tasks. You can use a data set copy program that provides these data management services as part of the copy process, such as DFSMSdss.

For the allowable combinations of FlashCopy with the other ESS copy functions, refer to 1.8, “Combinations of copy services” on page 10.

3

# Peer-to-Peer Remote Copy

In this chapter, the characteristics and operation of ESS Copy Services Peer-to-Peer Remote Copy (PPRC) are described. This chapter also discusses considerations useful when planning the implementation of PPRC.

PPRC Version 1:

- Synchronous PPRC, also known as IBM TotalStorage Metro Mirror
- PPRC-XD, also known as IBM TotalStorage Global Copy

PPRC Version 2:

- Asynchronous Cascading PPRC, also known as IBM TotalStorage Metro/Global Copy
- Asynchronous PPRC, also known as IBM TotalStorage Global Mirror

PPRC Pathing:

- ESCON
- PPRC over Fibre Channel links

When planning to implement and use PPRC, refer to the following publications, which complement the information presented in this chapter:

- *IBM TotalStorage Solutions for Disaster Recovery,* SG24-6547
- *z/OS DFSMS Advanced Copy Services,* SC35-0428
- *z/OS DFSMSdfp Advanced Services,* SC26-7400
- *IBM TotalStorage Enterprise Storage Server Web Interface User's Guide,* SC26-7448
- *IBM TotalStorage Enterprise Storage Server PPRC Extended Distance*, SG24-6568
- *Device Support Facilities, User's Guide and Reference,* GC35-0033
- *IBM TotalStorage Enterprise Storage Server Command-Line Interfaces User's Guide,* SC26-7494

## 3.1 Overview and introduction

Following is a summary of the functions available with PPRC Copy Services and the topics discussed in this chapter.

### 3.1.1 Overview

Peer-to-Peer Remote Copy (PPRC) is a proven remote data mirroring technique. It is used primarily as part of the business continuance solution for protecting an organization's data against disk storage subsystem loss or complete site failure.

PPRC is a hardware-based mirroring and remote copying solution of the IBM TotalStorage Enterprise Storage Server (ESS), that can be used in environments operating with the IBM @server zSeries, iSeries, pSeries, and xSeries, as well as the non-IBM Intel-based and UNIX-based supported servers.

PPRC can be used not only for application data recovery, but also for failover to remote sites for Disaster Recovery, remote migration of data and application workloads, as well as offsite backups.

The ESS Copy Services function Peer-to-Peer Remote Copy can operate in different modes:

- **Synchronous PPRC operation:** In this mode of operation, the ESS synchronously mirrors updates done to the primary volumes. It can be used over distances of up to 300 km depending on the link type used. In this chapter, we refer to it as Synchronous PPRC.
- **Extended Distance PPRC operation:** This technique operates non-synchronously and can be used over continental distances (using channel extenders or SAN) with excellent application performance. We refer to it with the acronym PPRC-XD.
- **Asynchronous Cascading PPRC operation:** A three site solution that synchronously mirrors from the primary site to the secondary site over metro distance, then asynchronously mirrors the volumes to the backup site over continental distances. We refer to this as Asynchronous Cascading PPRC.
- **Asynchronous PPRC operation:** By combining the speed of PPRC-XD with the point-in-time advantages of FlashCopy, repeated consistent copies are made at the remote site for recovery. This is referred to as Asynchronous PPRC.

### 3.1.2 Introduction

Typically, an installation must make copies of its data for the following reasons:

- Protection against hardware failure
- Protection against software failure
- Protection against natural disasters
- Protection against loss or loss of access to the production site
- Creation of test systems
- Creation of quality assurance systems
- Creation of systems for data mining

Also, consideration needs to be given to the status of the copied data with relation to both the consistency of the data across volumes as well as the asynchronous copy delay. These considerations then need to be assessed against the risk and impact of data loss along with the cost.

Peer-to-Peer Remote Copy is available in a number of different configurations to meet these different requirements.

## Synchronous PPRC

Synchronous PPRC ensures that, once the volume pair has been established and synchronized, the secondary volume will always contain the same data as the primary. The ESS ensures that a write complete response is not returned to the host I/O until the write operation has been completed on both the local and remote volumes.

This guaranteed write capability is typically suited to applications requiring the remote site to have all I/O in sync. It can also be used to copy data to a remote site.

These are some typical implementations:

- Critical applications requiring up-to-date backups where the recovery point is near or equal to zero
- Online applications requiring fast failover to a backup site with minimum or no loss of transactions
- Applications and processes where there is a high impact in the event of data loss

The I/O response time is directly affected by the distance between the sites and bandwidth of the path. For this reason, this is a short distance (metro) solution.

Synchronous PPRC requires a minimum feature of PPRC Version 1.

## PPRC-XD

PPRC Extended Distance (PPRC-XD) is a non-synchronous version of PPRC. This means that host updates to the primary volumes are not delayed by waiting for the update to be confirmed by the secondary ESS. It also means that the sequence of updates on the secondary volume is not guaranteed to be the same as on the primary volume.

PPRC-XD is an excellent solution for:

- Remote data copy
- Remote data migration
- Off-site backup
- Transmission of inactive database logs
- Application disaster recovery solutions based on periodic point-in-time copies of the data, if the application tolerates the quiesce of its I/O activity for a short period

PPRC-XD can operate at very long distances (continental), well beyond the distance supported for Synchronous PPRC transmissions, and with minimal impact on the application. The distance is limited only by the network and channel extender technology capabilities (see Figure 3-1).

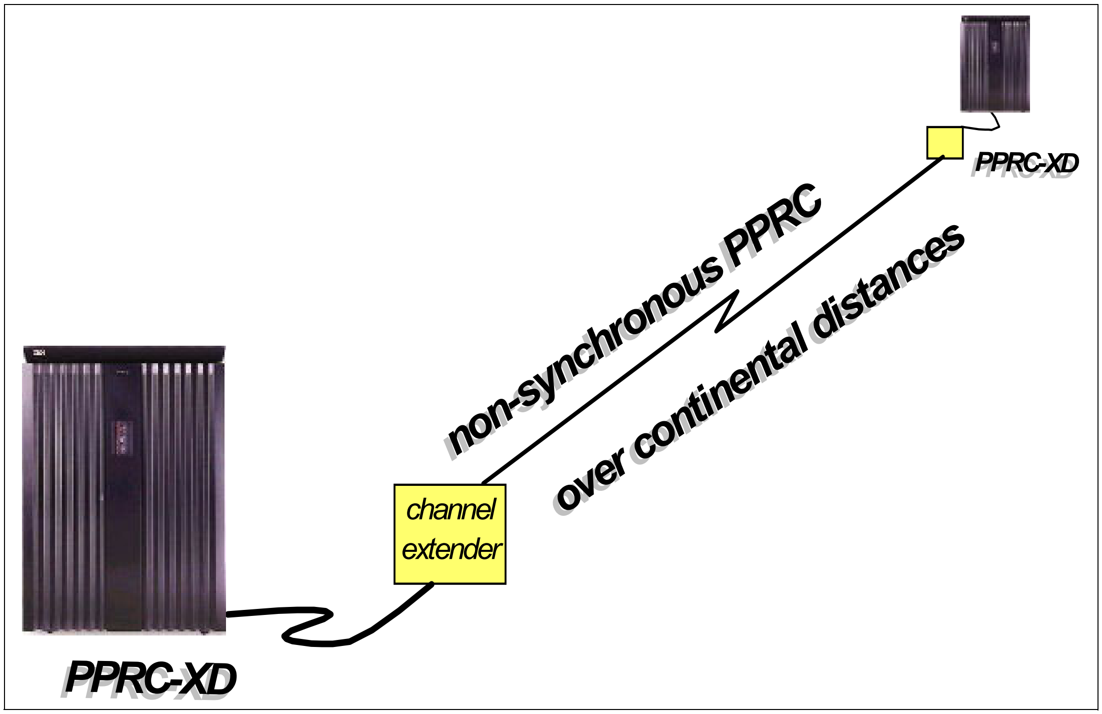



*Figure 3-1 ESS Copy Services: Peer-to-Peer Remote Copy - Extended Distance*

PPRC-XD requires a minimum feature of PPRC Version 1.

### Asynchronous Cascading PPRC

PPRC Version 2 was introduced with ESS LIC level 2.2.0, which included support for Asynchronous Cascading PPRC. Further enhancements were introduced with ESS LIC level 2.3.0 enabling PPRC Failover/Failback modes with Asynchronous Cascading PPRC.

- Asynchronous Cascading PPRC provides a long-distance remote copy solution for zSeries and open systems environments by allowing a PPRC secondary volume (involved in a Synchronous PPRC relationship) to also simultaneously serve as a PPRC primary volume in a PPRC Extended Distance (PPRC-XD) relationship to the remote site. This new capability enables the creation of three-site or two-site Asynchronous Cascading PPRC configurations.
- PPRC Failover and Failback modes are supported for Asynchronous Cascading PPRC. This helps reduce the time required to synchronize PPRC volumes after switching between sites during a planned or unplanned outage.

Asynchronous Cascading PPRC requires a minimum feature of PPRC Version 2.

### Asynchronous PPRC

Asynchronous PPRC was introduced with ESS LIC level 2.4.0. Asynchronous PPRC utilizes the low overhead response time of PPRC-XD. Periodically invoking a point-in-time FlashCopy at the remote site at regular intervals without disrupting the I/O to the primary volume, gives a continuously updating, nearly up-to-date backup. By joining many volumes into a session which is managed by the Asynchronous PPRC Master, multiple volumes can be copied to the remote site simultaneously while maintaining point-in-time consistency across those volumes. The time between the formation of these Consistency Groups is set by the user and the allowable range is 0 to 64K seconds. For most applications the time would be typically set in the order of a few seconds but can be set to many hours depending on the recovery requirements.

Applications for Asynchronous PPRC include:

- Maintaining a recoverable copy of critical volumes in an online or database environment at global distances
- Automation of consistency
- A continuous, consistent copy solution

Asynchronous PPRC requires a minimum feature of PPRC Version 2.

### PPRC paths

Before any PPRC pair relationship can be established between volumes, a PPRC path (or group of paths) must be established. The path(s) are uni-directional and go from the LSS containing the primary volume to the LSS containing the secondary, typically in another ESS in a remote site. These paths are explained and the setup discussed in 3.10, “PPRC links” on page 125.

These logical paths are created using physical paths, which require available host adapters at each end on the relative ESSs as well as the associated cabling, switches, directors and extenders as required. The following physical transport methods are supported:

- **ESCON:** Can carry up to 64 PPRC logical paths and can only be used in one direction.

  ESCON PPRC paths require a minimum feature of PPRC Version 1.
- **PPRC over FCP:** Fibre Channel Protocol can be used as the communications link between PPRC primary LSSs and secondary LSSs. Fibre Channel reduces the link infrastructure by at least 4 to 1 when compared to ESCON PPRC, and relieves logical and physical path constraints. These physical links can be bidirectional and can also share I/O with hosts using FCP.

  PPRC over FCP paths require a minimum feature of PPRC Version 2 and ESS LIC 2.3.0 or higher.

## 3.1.3 Synchronous versus non-synchronous

As the distance between ESSs increases, the Synchronous PPRC response time overhead increases proportionally, and this affects the application I/O performance. When implementations over extended distances are needed, a non-synchronous solution becomes an excellent trade-off. Business and recovery requirements need to be assessed to decide which non-synchronous solution provides the best fit, PPRC-XD, Asynchronous Cascading PPRC or Asynchronous PPRC.

You can estimate the impact of PPRC-XD on the application as being similar to working with PPRC volumes in suspended state. For the ESS there is some more work to do with the XD volumes as compared to the suspended volumes, because with PPRC-XD the changes have to be sent to the remote ESS, but this is not a significant overhead for the application.

We can look at the distance factor to illustrate how the different PPRC methods are positioned. PPRC-XD, Asynchronous Cascading PPRC and Asynchronous PPRC are non-synchronous solutions for implementations over long distances or continental distances well beyond the synchronous 103 km for ESCON (300 km for FCP) supported distance. As the distance increases there is an increasing lag in the time it takes for the data to be written to the remote site. This can be referred to as *drain delay*. As the copy is non-synchronous, this delay does not substantially affect the currency of the remote copy and has no effect on the local response time.

With PPRC-XD and Asynchronous Cascading PPRC, in order to take a consistent point-in-time copy of the data, you must do a catch-up transition or perform a Freeze/Run. Asynchronous PPRC eliminates the need to manually take consistent copies at the remote site or to interrupt the I/O activity at the primary site.

Synchronous PPRC (where no catch-ups are needed) is a synchronous solution for implementations within metropolitan areas.

### 3.1.4 Terminology

Here is some basic terminology used throughout this chapter:

- A site where the production applications run is referred to as either the *primary* site, *application* site*, production* site or *local* site.
- A site that has the mirrored data of the application site is referred to as the *secondary* site, *recovery* site, *backup* site or *remote* site.
- An IBM TotalStorage Enterprise Storage Server where the production data resides and which has a set of volumes in a PPRC relationship, is referred to as either the *primary* ESS, *application* site ESS or *local* site ESS.
- An IBM TotalStorage Enterprise Storage Server onto which the production data is mirrored by PPRC, is referred to as either the *secondary* ESS, *recovery* site ESS, or *remote* site ESS.
- In Asynchronous Cascading PPRC, the following terms apply:
  - The site where the application site data is first mirrored is referred to as the *intermediate site* or *bunker site*. This will be the first hop of the cascading PPRC.
  - The PPRC volume pairs established between the local site and the intermediate site are called the *local PPRC pairs.*
  - The PPRC volume pairs established between the intermediate site and the remote site are called the *remote PPRC pairs.*

### 3.1.5 Feature dependencies

PPRC is an optional chargeable feature of the ESS product and is available in 2 versions. PPRC Version 1 provides real-time mirroring and extended distance:

- Synchronous PPRC
- PPRC-XD

PPRC Version 2 further supports business continuance solutions with the following additional functions:

- PPRC over Fibre Channel links
- Asynchronous PPRC
- Asynchronous Cascading PPRC

Note that PPRC Version 2 includes all PPRC Version 1 features. Only one of the PPRC features (V1 or V2) can be ordered on the same machine. Also, any other ESS that will contain the PPRC pairs must also have the minimum PPRC feature version required to support the type of Copy Services being used. Also note that Flash Copy Version 1 cannot be ordered with PPRC Version 2 on the same machine.

For more detail, refer to Appendix D, “ESS Copy Services feature codes” on page 383. See Table 3-1 for the minimum allowable LIC levels for the various PPRC features.

*Table 3-1 Minimum requirements for PPRC features*

| PPRC Feature | Minimum PPRC License | Minimum ESS Licensed Internal Code | | |
|---|---|---|---|---|
| | | Fxx | 800 | 750 |
| Synchronous PPRC | PPRC-V1 | 1.3.0 | 2.0.0 | 2.3.1 |
| PPRC-XD | PPRC-V1 | 1.5.2 | 2.0.0 | 2.3.1 |
| Asynchronous Cascading PPRC | PPRC-V2 | 2.2.0 | 2.2.0 | 2.3.1 |
| Asynchronous PPRC | PPRC-V2 | no support | 2.4.0 | 2.4.0 |
| PPRC Paths over ESCON | PPRC-V1 | 1.3.0 | 2.0.0 | 2.3.1 |
| PPRC paths over FCP | PPRC-V2 | no support | 2.3.0 | 2.3.1 |

## 3.2 Choosing the best method

Each of the PPRC Copy Services features has different characteristics designed to cover a wide range of client needs. Every business has its own special and unique requirements, and within a business there is usually a diverse range of applications. Some applications may require fast recovery due to critical online requirements, others cannot tolerate the loss of a single transaction.

For each of the different business applications, the storage management requirements are unique, but can generally be categorized under the following general points (examples are given to illustrate):

- Data loss (the amount of data that is lost after recovery):
  - Transaction based applications, financials, ordering: Little or no data loss is tolerable.
  - Database applications with logging: Some data loss is tolerable.
  - Batch, since the application can be rerun: Data loss can be recovered.
- Consistency (writes to all volumes are in the correct time sequence)
  - Database applications: Each transaction is logged when started and again when completed to allow correct recovery. Log and data must be in step.
  - General applications: Preserves the order of dependent writes.
  - Operating systems: File system recovery (journaling or VTOC consistency), catalogs.
- Outage time (time to recover):
  - In the event of a disaster or failure, the time it will take to bring backup systems up on the mirrored volumes.
  - Businesses with customer facing online systems may not be able to tolerate more than minutes of down time.
  - Most online database systems are required to be back online within an hour.
  - Many businesses, using risk management of non-critical processes, accept recovery times in hours or days depending on the application requirements.
- Distance between production and recovery sites:
  - Same site
  - Recovery or backup site in same city, or nearby - Metro, 0 - 300 km
  - Another city, country or continent - Global, greater than 300 km

- Copying requirements:
  - Point-in-time mirrored copies for disaster recovery testing, Quality Assurance testing, archive backups
  - Single volume mirrored copies, locally or at another site
  - Multiple volume mirrored copies
  - Continuous copies of data from a live system without disruption

While recovery remains with the systems and the applications, the use of the appropriate Copy Services' feature will ensure that the data is copied in a suitable state for recovery depending on the business and application requirements.

There are four PPRC features designed to meet the different needs of client's businesses and within their enterprise, the different needs of individual applications:

- **Synchronous PPRC:** Metro, zero data loss
- **PPRC-XD:** Global
- **Asynchronous Cascading PPRC:** Global, zero data loss
- **Asynchronous PPRC:** Global, point-in-time consistent

## 3.2.1 Disaster Recovery

In the introductory chapter we introduced Disaster Recovery and discussed the main objectives; refer to 1.6, "Disaster Recovery" on page 7. Now let's introduce some important terms used in Disaster Recovery and look at their levels. It is not the intention of this book to discuss Disaster Recovery in detail. Refer to the redbook, *IBM TotalStorage Solutions for Disaster Recovery*, SG24-6547, for an in-depth review and discussion on this subject.

### Business objectives for Disaster Recovery

As shown in Figure 3-2, to design a cost-effective solution, start determining the following objectives by application or business line:

- **Recovery Time Objective (RTO):** What is the business cost-justified elapsed time to recovery?
- **Recovery Point Objective (RPO):** When the RTO is met, how much data can you afford to recreate or lose?
- **Network Recovery Objective (NRO):** How long does it take to switch the entire network over to the backup data center?

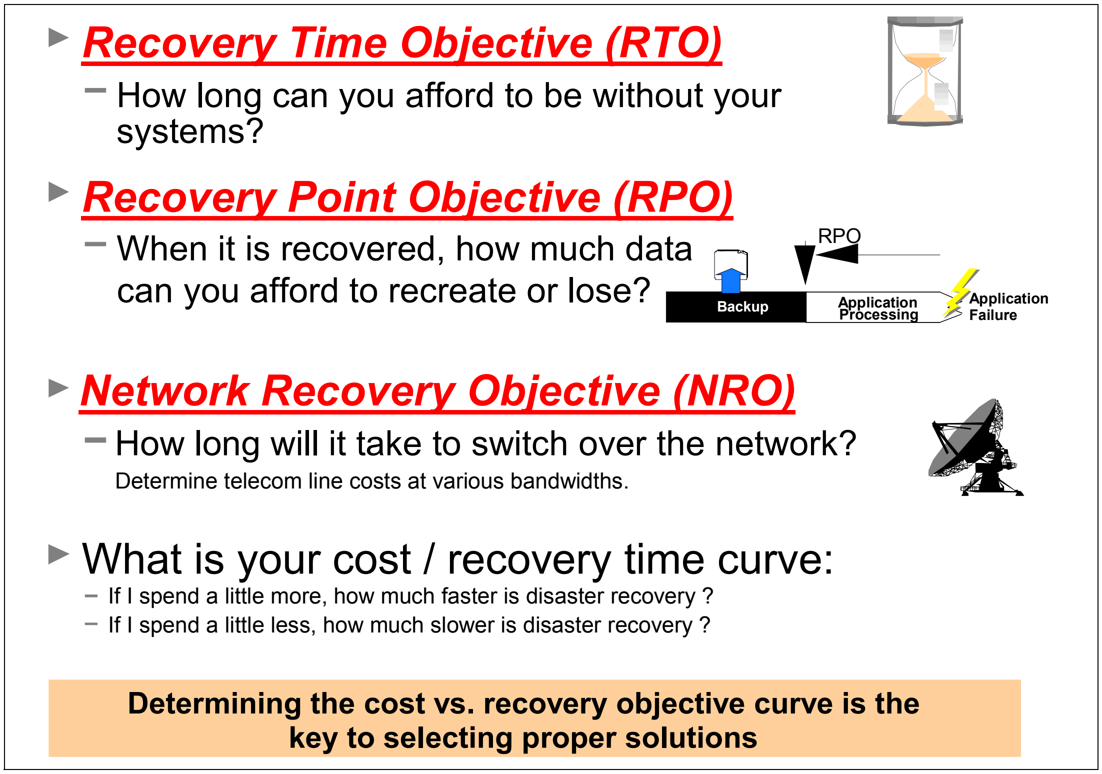



*Figure 3-2 Business objectives for Disaster Recovery*

The first two objectives (RTO and RPO) can often be balanced against the cost of the solution. When the third objective (NRO) comes into play, networking issues come into consideration. For example, there is no need to purchase a 30 minute RTO solution if the network provider requires 2 hours to switch the network.

### Planned versus unplanned outages

Planned outages are just as effective at removing service from the end users as unplanned outages — and they are much more frequent. Yet typically, disaster recovery solution cost justification is attempted based on the unplanned outage cost alone. So, a realistic return on investment analysis should always consider the planned outages' effects.

## 3.2.2 Tiers of Disaster Recovery

When looking for a solution, you will always want to position it according to its effectiveness and the corresponding investment that must be done. This leads us to the tier considerations (refer to Figure 3-3).

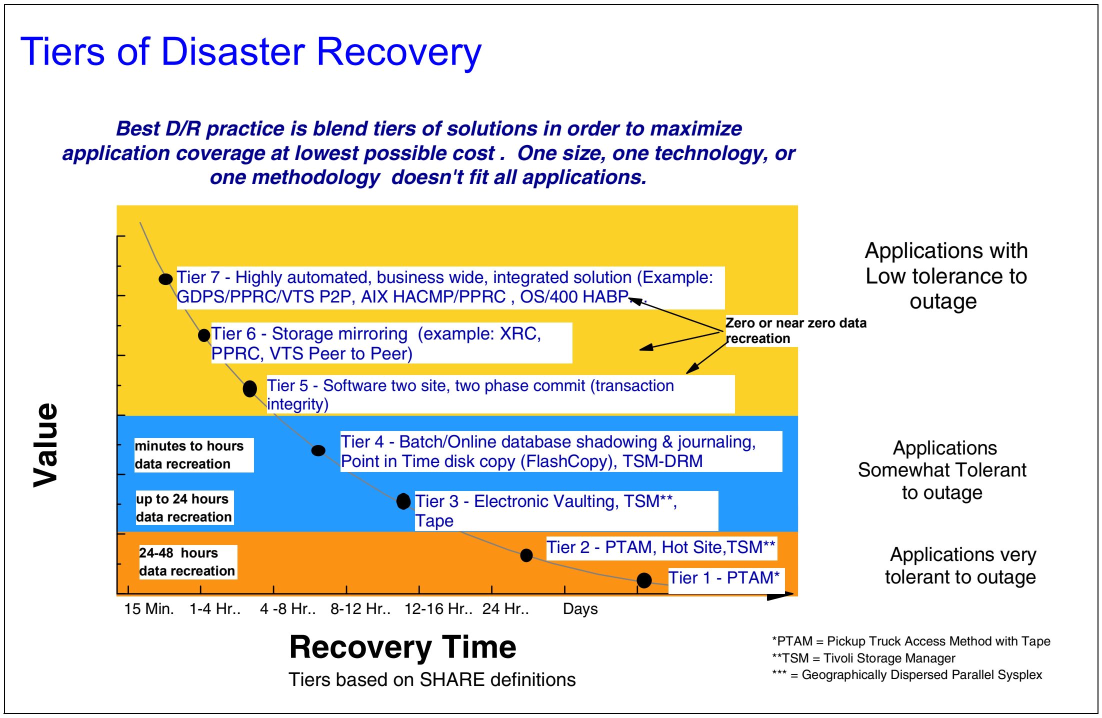



*Figure 3-3 Tiers for Disaster Recovery*

The chart in Figure 3-3 is a standard disaster recovery industry tier chart, showing the different solutions that can be implemented with their different recoverability characteristics.

In general, there are three main bands. Within each band, there are tiers, which vary as follows:

- Tier 0 (no recovery capability)
- Tier 2, 3 (tape intensive recovery methods)
- Tier 4 (disk mirroring or disk point-in-time facilities)
- Tier 5 (software/ database specific recovery)
- Tier 6 (near zero or zero data loss disk mirroring)
- And finally, tier 7, which is a completely automated self-managed failover of servers, storage, software, and automation, networking all and integrated with services into the user's application.

### Power failure data consistency versus transaction data consistency

There are two levels of data consistency to be considered:

- **Power failure data consistency:** An example might be if the primary machine room were to experience an instantaneous power failure.
- **Transaction consistency:** This is achieved when the application is in a known state in relation to the storage.

Therefore, software recovery on top of hardware recovery is essential and cannot be avoided.

## 3.2.3 Comparing PPRC features

In Figure 3-4 we have shown a comparison of the main attributes of each PPRC feature. The following discussion explains this diagram in more detail.

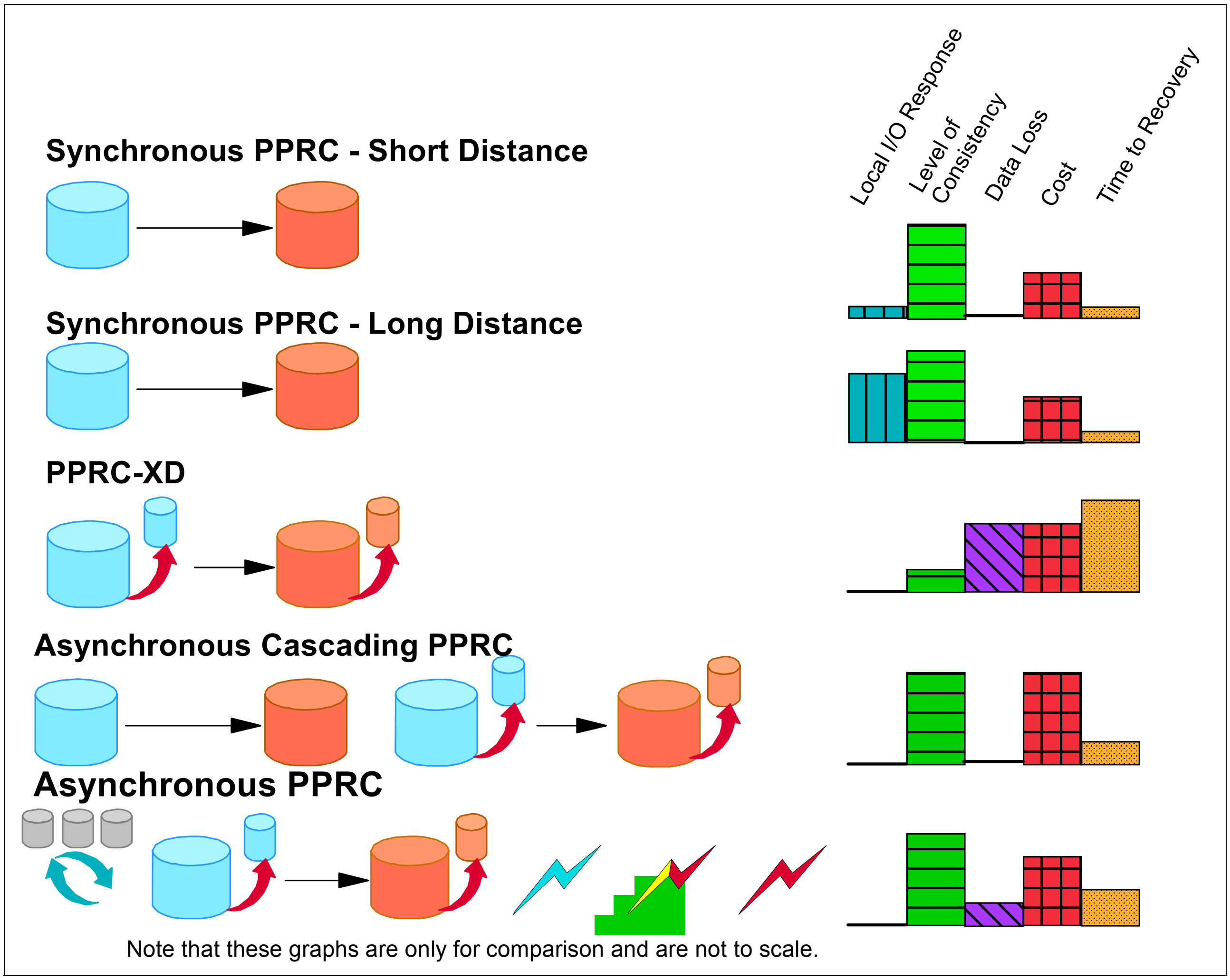



*Figure 3-4 PPRC feature comparisons*

## Terms

Here is an explanation of some terms used in the following discussions:

- **Local I/O response:** This is the response time to the host I/O at the local or primary site from I/O start to completion notification. A low value is a short (good) response time.
- **Level of consistency:** This is the degree of consistency of the data after the copy operation and recovery. The data across all volumes is in sequence and the last writes were at the same point-in-time. A high value indicates good consistency.
- **Data loss:** This refers to the amount of data that was written at the primary site after the copy operation or that cannot be recovered at the remote site. A low value is a small amount of data loss.
- **Cost:** This is the comparative cost of ESS resources (capacity, ESS subsystems), site resources and recovery workload. A higher value means a higher cost.
- **Time to recovery:** This is the comparative time it takes from a disaster event occurring to having only the data ready to use at the recovery site. This does not represent the recovery of the entire application and all business processes. A low value is a short time.

Refer to the diagram in Figure 3-4 for the following discussions.

## Synchronous PPRC - Short Distance

Consider short distances as meters up to 20 kilometers. The remote ESS may be in the same building, but will most likely be at another site in the same city. Typical links would be via DWDM or SAN.

### *Local I/O response*

The response time for the I/Os will be very good, but noticeably higher than running without PPRC. This is usually acceptable. I/Os are accepted, written to the local ESS and also sent to the remote. Successful I/O completion is not returned to the local host until the remote has responded that the write operation is also complete.

### *Consistency*

This mode of operation gives continuous consistency. As completion of the write operation is not reported to the host until the remote site also has written the data, then data is always consistent in terms of the applications.

### *Data loss*

After return of I/O completion to the application, the data at the remote site is the same as the data at the local site. From this point of view, there is no data loss. Incomplete operations may have to be discarded or recovered. There may be some further application data loss after the recovery of the file systems or data base recovery, for example in-flight transactions.

### *Cost*

For full coverage, a second site is required. This will require the installation of ESS subsystems at the remote site with sufficient capacity to create a volume for every volume at the primary site that needs to be mirrored. ESCON or SAN connectivity between the sites is also required with sufficient bandwidth to accommodate all traffic at peak loads.

### *Time to recovery*

As the remote data is always consistent, recovery of the storage is very fast. As soon as the PPRC relationships have been suspended (which can usually be done with FREEZE commands) the data can be accessed and used at the remote site.

### *Applications*

- Critical online systems.
- Financial and legal transaction processing.
- Customer facing online systems.
- Copying.
- Site failover requirements
- Entire system mirroring

**Reference:** Synchronous PPRC is discussed in 3.6, “Synchronous PPRC” on page 91.

## Synchronous PPRC - Long Distance

The same as the short distance description above, only over local or metro distances, considered to be in the same city or nearby. The behavior is much the same, only the local response time is adversely affected as the distance increases.

### *Local I/O response*

As the distance increases the response time to I/O at the local sites increases proportionally. For continuous operation a maximum distance of 103 km over ESCON and 300 km over FCP is supported. It is possible to use synchronous mode over longer distances during low or zero I/O rates as part of recovery using one of the non-synchronous modes.

This distance factor needs to be carefully considered, as the degradation in response time may exceed application tolerance long before the maximum distance is reached. Your IBM Representative will be able to assist with planning, but it ultimately depends on your business requirements and applications.

#### *Consistency*

As above, for short distance.

#### *Data loss*

As above, for short distance.

#### *Cost*

Because the distance is increasing, so also does the cost. For example, Channel Extenders or DWDMs add to the cost. It is necessary to maintain the PPRC links continuously with sufficient bandwidth to cope with the maximum workload.

#### *Time to recovery*

As above, for short distance.

#### *Applications*

- As above, where geographic separation of sites is required within metro distances.

**Reference:** Synchronous PPRC is discussed in 3.6, "Synchronous PPRC" on page 91.

### PPRC-XD

Designed for global distances, the effect on local response times is negligible, while asynchronously mirroring to the remote site. Very good as part of a solution to copy data remotely with little impact, to maintain an almost up-to-date copy of data or as used in conjunction with other ESS Copy Services features.

#### *Local I/O response*

This is excellent and almost the same as running native without PPRC. Because the write to the remote ESS is batched and sent asynchronously, there is no impact to the host. I/O complete is returned as soon as the write is complete in the local ESS.

#### *Consistency*

Poor. As the load on the paths increases the I/O transfers can become more out-of-sync between volumes. When the I/O rate decreases, the copy process is able to catch-up, so the level of consistency will vary depending on load.

Provided applications are able to recover from a checkpoint, then this still provides a very useful and low impact solution. A good solution for databases is to send the data via PPRC-XD and send the transaction logs with Synchronous PPRC.

If the primary site ESS is still available, then there is the ability to go-to-sync. This changes the PPRC-XD relationship to Synchronous PPRC mode. There will be a delay while the remote copies catch up, then the rules are the same as for Synchronous PPRC above. Also included in this feature is the ability to freeze the primary site, which creates consistency across multiple volumes.

#### *Data loss*

Depending on application checkpoints this can be high. Generally applications will have to go back to a known good point and recover forward. The amount of data loss will depend on the level of recovery but will generally be worse than other methods of PPRC.

### *Cost*

ESS resources are the same as with Synchronous PPRC, although there is usually a need to have volumes available for making FlashCopies at the remote site. Path requirements are less for two main reasons: 1. We only have to allow for the average throughput. This also needs to be carefully sized to ensure we don't get too far behind. 2. Because the PPRC process is batching the copies to the remote, there is more efficient use of the available bandwidth.

### *Time to recovery*

This can be very long, depending on the application's ability to recover. Database recovery and rebuild can take many hours or may not be possible, requiring backup restore. It may be required to go back to a checkpoint and migrate forward to a point close to the failure. For some applications where writes are not frequent (such as web hosting) recovery time may be short and systems can be brought online at the remote site reasonably quickly. Generally, the recovery time from this inconsistent state should be considered unpredictable.

### *Applications*

- Low impact database and batch recovery
- Copying to test site for testing or planned maintenance
- Low cost solution for global distance requirements where inconsistency and lag can be tolerated

**Reference:** PPRC-XD is discussed in 3.7, "PPRC-XD" on page 94

## Asynchronous Cascading PPRC

A very good solution where data is mirrored synchronously to a nearby intermediate site, then asynchronously at global distances to the remote site. This gives full protection against loss of any one of the three sites, while still maintaining good response times at the primary site and zero RPO with no data loss at the remote site. Also very fast recovery times.

### *Local I/O response*

Because the first link of the PPRC copy is synchronous, this will depend on the distance. Use the same rules as Synchronous PPRC to determine how far away the *bunker* (intermediate) site can be and still meet the response time requirements of the applications.

### *Consistency*

Again, because the second link is running PPRC-XD, the consistency while running is poor. But, with the use of an intermediate site (and assuming the risk of losing the primary and intermediate sites simultaneously is very low) the data can later be caught up, even after the primary site has failed. By issuing commands to go-to-sync or simply waiting for the PPRC-XD to catch-up, then this gives excellent consistency at the remote site.

Depending on the implementation, there may be varying degrees of risk when using this method. If a two-site implementation is used, then the risk increases.

### *Data loss*

Again, provided there is only one site lost in a three-site implementation, then there is no data loss and the recovery is the same as for Synchronous PPRC.

### *Cost*

The need for an additional site and replication of the equipment of the remote site at the intermediate site makes this the most costly solution. Also the need for two sets of links between sites. This can be done with 2 sites, but this compromises the protection against primary site loss.

### *Time to recovery*

Time to recovery is very good. Once the PPRC-XD on the second link has drained all the changes (or been forced to synchronous mode), then the data at the recovery site is ready for use.

### *Applications*

- Critical applications requiring quick recovery times and near zero data loss over global distances

**Reference:** Asynchronous Cascading PPRC is discussed in 3.8, “Asynchronous Cascading PPRC” on page 108

## Asynchronous PPRC

Continuously updated Consistency Groups are being created at the remote site on an additional set of volumes. These are managed and controlled by the *Master*, running in one of the ESSs in the enterprise. This PPRC relationship consists of two parts, a PPRC-XD pair that can be over global distances and a FlashCopy pair on the remote site used to make consistent copies.

### *Local I/O response*

Because we are using PPRC-XD, this is excellent, with negligible difference from native non-PPRC writes.

### *Consistency*

Excellent consistency. The FlashCopies at the remote site are synchronized in time with the data, therefore the order of dependent writes is preserved. Another Consistency Group is not started unless the previous one completed, or timed out. If the FlashCopy is not completed on all volumes, there is the ability to revert to the previous FlashCopy.

### *Data loss*

The creation of Consistency Groups can be on the order of seconds apart. Five seconds or less is quite achievable. Tuning and application requirements (see “Start/resume a session” on page 124) will dictate the best values, part of which are user setable. Also the available bandwidth is a factor. The recovery process is to recover to the last successful FlashCopy.

### *Cost*

Requires the remote site and additional ESS hardware with capacity to mirror every volume that needs to be mirrored times two, one for the PPRC target and one for the FlashCopy.

### *Time to recovery*

The consistency of the FlashCopies must be assessed and then the data recovered back to the remote volumes for use by the applications. IBM has tools available to automate and manage this process. See Chapter 4, “Managing Copy Services” on page 145.

### *Applications*

- Most high availability business applications where a minimum data loss can be tolerated
- Systems requiring fast recovery at the remote recovery site
- Global distances

**Reference:** Asynchronous PPRC is discussed in 3.9, “Asynchronous PPRC” on page 116.

## 3.3 Invocation and control

In this section we introduce the tools that the zSeries and S/390 systems users have to invoke and control Synchronous PPRC and PPRC-XD. These will be discussed in detail in Chapter 4, “Managing Copy Services” on page 145.

### *z/OS and OS/390*

For z/OS and OS/390 systems users, Synchronous PPRC and PPRC-XD can be managed using the following facilities:

- TSO commands
- ESS Copy Services Web User Interface (WUI)
- ICKDSF utility
- ANTRQST application programming interface (API)

### *z/VM*

They can be invoked and controlled in the VM environment using:

- ESS Copy Services Web User Interface
- ICKDSF utility

### *VSE/ESA*

They can be invoked and controlled in the VSE environment using:

- ESS Copy Services Web User Interface
- ICKDSF utility

### *TPF*

They can be invoked and controlled in the TPF environment using:

- ESS Copy Services Web User Interface
- ICKDSF utility
- TPF itself
- Other attached servers that support control of PPRC

In addition, the ESS Copy Services Command Line Interface (CLI) can be used to invoke a previously saved task created from the Web interface if a supported server is connected (refer to 4.2, “TSO commands for PPRC” on page 147 for details).

Details and usage of the TSO commands, the ESS Copy Services WUI, the ICKDSF utility, and the ANTRQST API for PPRC-XD invocation and control are discussed in Chapter 4, “Managing Copy Services” on page 145.

## 3.4 PPRC configuration guidelines

This section discusses the configuration guidelines that must be considered when planning the PPRC environment.

PPRC Version 1 only supports ESCON links, while PPRC Version 2 in addition supports FCP links. The present section discusses the configuration guidelines when using ESCON links.

> **Note:** If you are planning to use FCP links, the information in this section should be complemented with the information discussed in 3.10, “PPRC links” on page 125.

## CKD logical subsystems

During the ESS installation process, you must have configured the *logical subsystems* (LSSs) that will be available in the ESS. The LSS is a logical structure that is internal to the ESS and groups up to 256 logical volumes. LSSs are identified by a unique ID. The LSS relates directly to the *logical control unit* (LCU) concept of the ESCON and FICON architectures.

The count-key-data (CKD) logical subsystems are configured with the ESS Specialist at the time of configuring the logical control units (LCUs) at S/390 storage allocation time. The device adapter (DA) to LSS mapping is a fixed relationship. Each DA supports two CKD logical subsystems and two FB logical subsystems. So, a DA pair supports four CKD and four FB LSSs as illustrated in Figure 3-5.

When all the eight loops have capacity installed, then there can be up to 16 CKD LSSs, and up to 16 FB LSSs available to support the maximum of 48 RAID ranks. Each LSS supports up to 256 logical devices.

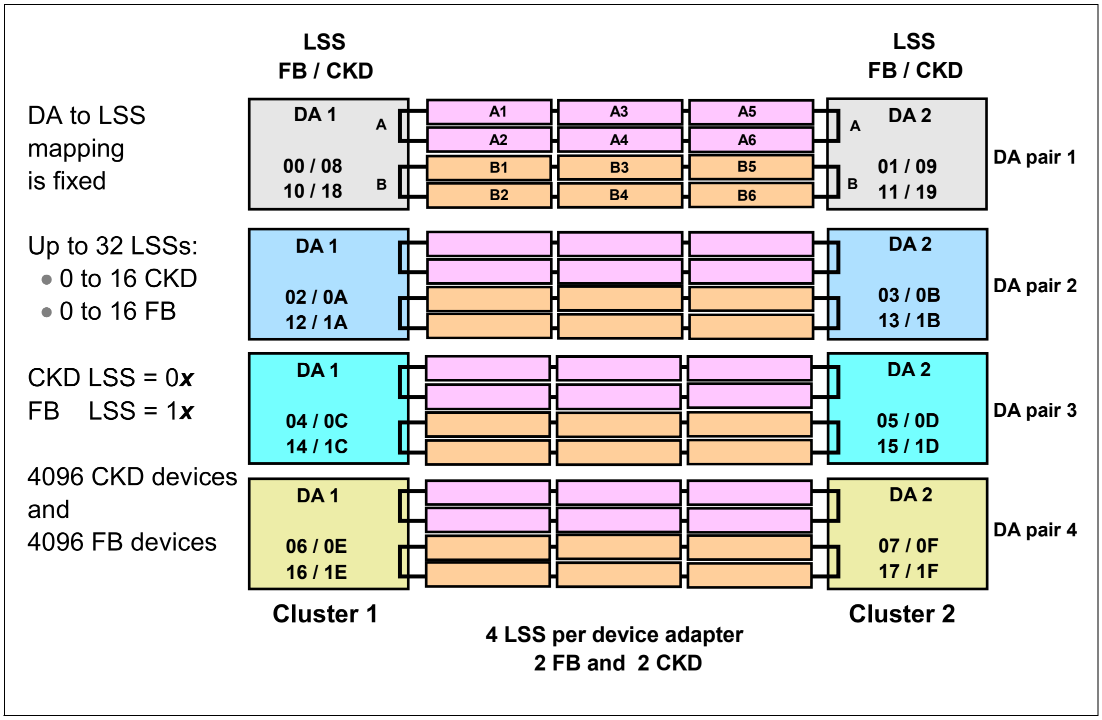



*Figure 3-5 Logical Subsystems and device adapters mappings*

The numbering of the logical subsystems indicates the type of LSS. CKD LSSs are numbered x'00' to x'0F' and the FB LSSs are numbered x'10' to x'1F'. As previously mentioned, for the CKD host view, a LSS is mapped one-to-one to a logical control unit (LCU). These values correspond to the `CUADD` values that are specified in S/390 `CNTLUNIT` macro in the IOCP or HCD.

> **Tip:** As part of the configuration process, you can define the maximum number of LSSs of each type (either CKD or FB) you plan to use. If you plan to use the ESS only for zSeries volume allocation, then you can set the number of FB LSSs to 0. This releases the definition space from cache. But, you must also remember that going from 8 to 16 LSSs is disruptive, so you should decide in advance how many you will need.

## 3.5 Data consistency in a PPRC environment

The following section discusses the PPRC elements that come into play when planning how to ensure data consistency at the recovery site, in the event of unplanned outages affecting components of the PPRC configuration.

Data consistency across all secondary volumes spread across any number of LSSs is essential for logical data integrity, and to enable a normal database restart.

### Dependent writes

Applications where one write is dependent on the completion of another, are said to have *dependent writes*.

### Data consistency on the primary volumes

Using dependent writes, applications are able to manage the consistency of their data on the primary volumes. This way a consistent state of the application data on disk is maintained if a failure occurs in the host processor, software, or storage subsystem.

Some common examples of dependent writes are:

- Databases and associated log files
- Catalogs and VTOCs
- VSAM indexes and data components

Database data sets are related, with values and pointers from indexes to data. Databases (DB) have pointers inside the data sets, in the DB catalog and directory data sets, and in the logs. Data integrity must be kept across these components of the DB.

Figure 3-6 shows a typical DB transaction for updating DB objects:

1. The I/O request is written to the DB log.
2. After the log has been successfully updated, the DB object is updated.
3. After the DB object has been successfully updated, the DB log is again updated to mark the transaction as completed.

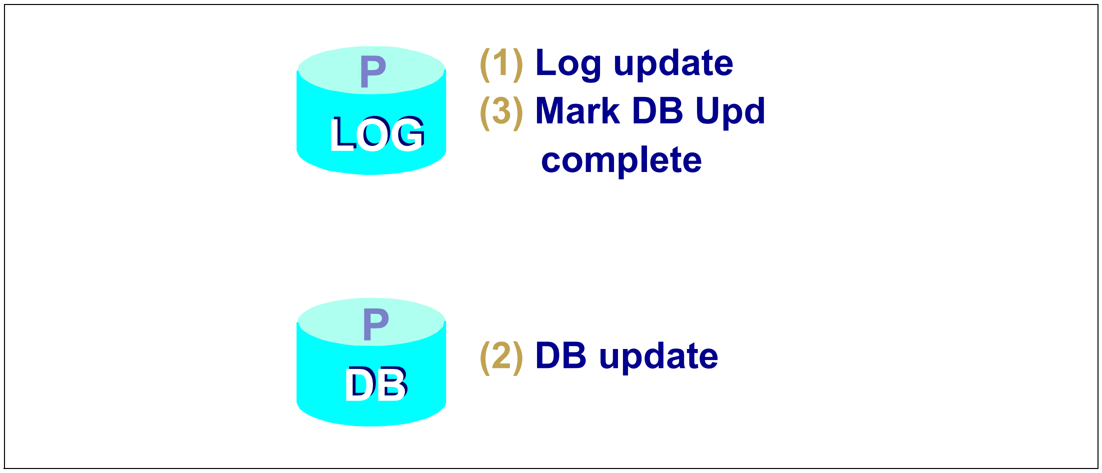



*Figure 3-6 Dependent write*

### Data consistency on the secondary volumes

Data consistency on the secondary volumes must be managed by the PPRC implementation. One very efficient way to do this is using automation procedures, which are able to freeze the activity on the required volumes in the event of an unexpected outage.

When in a mirroring (synchronous) environment, the DB transaction sequence for updating DB objects will have the following characteristics:

1. The I/O request is written to the DB log — on the primary volumes:
   – This DB log update is mirrored on the secondary volumes.
2. After the log has been successfully updated, the DB object is updated:
   – This DB object update is mirrored on the secondary volume.
3. After the DB object has been successfully updated, the DB log is updated to mark that the transaction is completed:
   – This DB log update is mirrored on the secondary volume.

### Rolling disaster

In disaster situations, it is unlikely that the entire complex will fail at the same moment. Failures tend to be intermittent and gradual, and disaster can occur over many seconds, even minutes. Because some data may have been processed and other data lost in this transition, data integrity on the secondary volumes is exposed. This situation is called a *rolling disaster*. The mirrored data at the recovery site must be managed so that cross-volume or LSS data consistency is preserved during the intermittent or gradual failure.

An example based on the write sequence of the DB system of how data consistency on the secondary volumes can be affected when a rolling disaster is progressing, is illustrated in Figure 3-7. In this example, the link between the volume where the DB data resides (DB-P) and its mirrored pair at the recovery site (DB-S) is lost, while the link between the DB log volume (LOG-P) and its related secondary pair (LOG-S) is still available.

With the link between DB-P and DB-S lost, a write sequence of (1) - (2) - (3) was able to be completed on the primary devices. But while the LOG writes (1) and (3) are mirrored on the LOG-S device, the DB write (2) will not be applied on DB-S. If a restart is done at the recovery site, this will cause the database manager to analyze the log and clean up pending transactions, but it will consider this transaction to be complete. So, we will have the DB-S with some updates not applied, which will probably go unnoticed. This poses a serious data integrity problem.

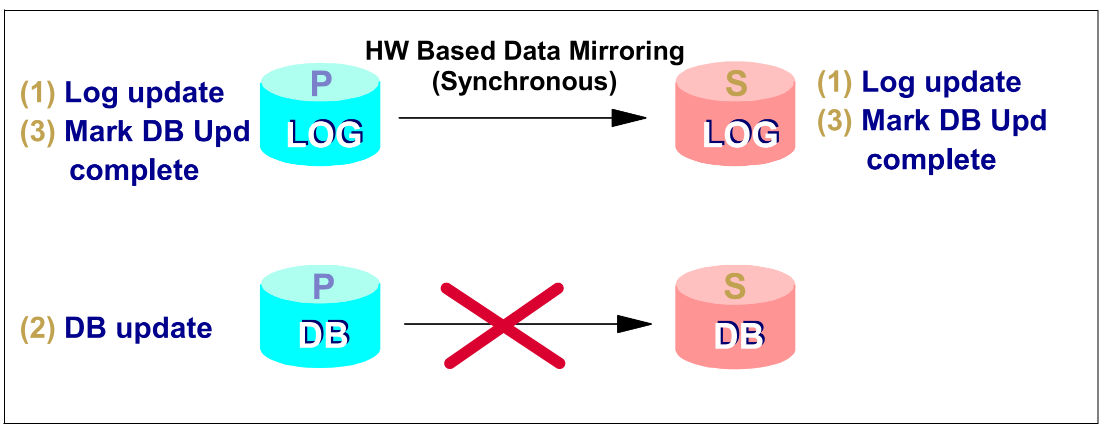



*Figure 3-7 Rolling disaster - Data consistency exposure*

Using Consistency Groups and critical attributes, which are implemented along with appropriate automation solutions, PPRC can be used to manage the data consistency and integrity at the remote site.

## Freeze command

Messages and environmental triggers can be detected by automation and are used to suspend all required secondary PPRC operations by using the CGROUP command with the FREEZE parameter — if you are using TSO commands to invoke PPRC functions.

The FREEZE option removes the paths, and suspends the pairs for a given LSS pair. If used in combination with the CGROUP(Y) option of the CESTPATH command, then an ELB condition will be forced on all the LSSs where the FREEZE is issued, and thus, all the I/Os to the respective primary volumes will be queued while in the ELB window. Once automation detects that the updates can be resumed, it can issue the CGROUP command with the RUN parameter. This command releases all I/O to the primary devices, but still leaves the secondary devices suspended at a point-in-time (with the FREEZE the associated paths are removed).

The default 2 minute timer of the extended long busy state gives automation enough time to issue a CGROUP FREEZE command to the necessary LSSs. I/O resumes after the default 2 minutes if a CGROUP RUN command is not received before.

The ICKDSF PPRCOPY FREEZE and PPRCCOPY RUN commands provide the same functionality as the TSO CGROUP FREEZE and CGROUP RUN commands.

Alternatively, if you are using the ESS Copy Services Web User Interface, starting from the Logical Subsystems panel, selecting the **Freeze PPRC Consistency Group** option and the **Consistency Created** option, will allow you to define the tasks needed to freeze and to thaw the I/O activity at the logical subsystem level.

## Critical attribute

The critical attribute determines what will happen after a failed I/O to a secondary volume occurs. What happens will depend on how this attribute was set when the pair was established, and how the ESS was configured.

The critical attribute is set when establishing the volume pair. If using TSO commands, the CRIT parameter of the CESTPAIR command is used for this. If you are using the ESS Copy Services Web User Interface, starting from the Volumes panel (when selecting the copy options) the **Critical volume mode** option sets this attribute for the volume pair.

If CRIT(NO) was specified in the CESTPAIR command when the pairs were established, then following an I/O error completion to the secondary volume, PPRC will suspend the volume and allow subsequent write requests to the PPRC primary volume. The pair is suspended, the secondary volume does not receive any more updates, and the primary ESS will perform change recording, thus keeping track of the primary updates for subsequent re-synchronization.

If CRIT(YES) was specified with the CESTPAIR command, and if an I/O error to the secondary volume occurs, PPRC will either allow or not subsequent writes to the primary, depending on how the ESS was configured. The IEA491E message will present the reason `SUSPENDED, CRITICAL-STATE, ALL WRITES FAILED`. The PPRC pair then remains in a suspended state until the problem is corrected and a CESTPAIR RESYNC or CDELPAIR command is later issued.

Whether subsequent writes to the primary volumes are allowed or not will depend on how the ESS is configured:

- `CRIT=YES - Paths` (Light version)
  - Suspend the pair and do not accept any further writes if the control units can no longer communicate.

- Suspend the pair and accept further writes to the primary volumes if the LSSs still can communicate with each other. The reason for not being able to copy the data to the remote volume is probably only a device problem on the secondary site, and not a disaster. Therefore, we continue with write operations to the primary volume. The ESS records which tracks have changed. After investigation of the problem and after it has been solved, you can re synchronize source and target volume again.

– `CRIT=YES - All` (Heavy version)

- Suspend the pair and do not accept any further writes to the primary volume if data cannot be sent to the secondary volume.

These attributes are configured on the ESS, on an LSS basis. From the Logical Subsystems panel presented by the ESS Copy Services WUI, after selecting the desired LSS, click the **Properties** button and the window presented in Figure 3-8 will pop up.

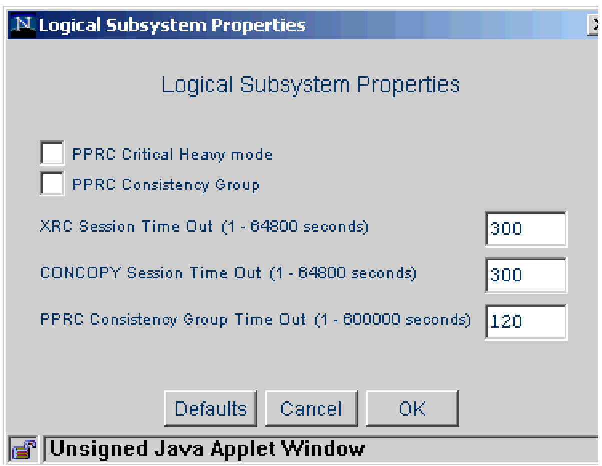



*Figure 3-8 Setting the CRIT attribute on the ESS*

As illustrated in Figure 3-8, the Critical Heavy mode can be set for the selected LSS.

## Consistency Group and critical attribute combination

Table 3-2 shows what happens when the CRIT parameter of the CESTPAIR command, and the CGROUP parameter of CESTPATH command are combined. These options can also be set using the ESS Copy Services WUI, as explained in *IBM TotalStorage Enterprise Storage Server Web Interface User's Guide,* SC26-7448.

*Table 3-2 Interactions of CRIT and CGROUP parameters*

| | **CRIT(Y)** | **CRIT(N)** |
|---|---|---|
| **CGROUP(Y)** | IEA494I with long busy state | IEA494I with long busy state |
| | Pair in error suspended | Pair in error suspended |
| | Unit check on primary device | No unit check on primary device |
| | IEA494I when suspended | IEA494I when suspended |
| **CGROUP(N)** | No long busy state | No long busy state |
| | Pair in error suspended | Pair in error suspended |
| | Unit check on primary device | No unit check on primary device |
| | IEA494I when suspended | IEA494I when suspended |

The IEA494I message with SUSPENDED status is issued for the in error pair to all z/OS images that have the primary volume online. An IEA491E message is returned to the issuing system of the next I/O that is attempted to the volume that is suspended.

For further discussion on the CGROUP and the CRIT parameters, the following publication can be referenced: *z/OS DFSMS Advanced Copy Services,* SC35-0428.

## Alert messages

Messages are issued to alert of any change in the status of the PPRC volumes. These alert messages can be intercepted and acted upon by automation routines like Geographically Dispersed Parallel Sysplex™ (GDPS). GDPS is an IBM service offering described in Appendix B, “Geographically Dispersed Parallel Sysplex (GDPS)” on page 359.

### *Consistency grouping and IEA494I messages*

The CGROUP parameter of the CESTPATH command is used to control error situations and to maintain consistency at the secondary site. The same functionality is provided by the ESS Copy Services Web User Interface, when establishing paths starting from the Paths panel, with the **PPRC Consistency Group** option from the Select path options wizard.

When specifying CGROUP(YES) on the CESTPATH command (or, its equivalent with the ESS Copy Services WUI), the disk storage subsystem ERP (error recovery procedures) will put the primary LSS in an extended long busy (ELB) wait condition if PPRC cannot update a secondary volume. Also, the ELB state condition will be reported on message IEA494I. The IEA494I message is issued whenever volume pairs are changing status.

If a system attempts an I/O to the primary volume while the status is changing, it will receive the IEA494I message with a status of SUSPENDING. When the pairs actually suspend, the IEA494I is broadcast to all of the attached systems where the primary volume is online, informing you that the pairs are now SUSPENDED.

These are the messages issued:

```
IEA494I 1162,LIVI62,PPRC PAIR SUSPENDING,SSID=00B0,CCA=02, EXTENDED LONG BUSY STATE
IEA494I 1162,LIVI62,PPRC PAIR SUSPENDED,SSID=00B0,CCA=02, EXTENDED LONG BUSY STATE
```

### *IEA491E message*

When PPRC detects an error that prevents the update of a secondary volume, the pair is suspended and message IEA491E is issued indicating that an error has occurred, and a possible reason is given. IEA491E is returned to any system attempting an I/O to the failed volume pair.

When the error that caused IEA491I to be issued has been corrected, the volume pair can be re-synchronized. The following is an example of the IEA491E message:

```
IEA491E LIVI62,PPRC SUSPENDED, COMMUNICATION_TO_SECONDARY_FAILURE,
(PRI)SER=0339-35570,CCA=02 (SEC)SER=0339-35416,CCA=03 CONTINUATION OF IEA491E
SNS=101010F0C20000FBF2030C69008A580C69008AF28700
```

## Automation

Alerted by the PPRC messages, the automation functions can detect when outage conditions are occurring. Automation procedures can then perform damage evaluation and limitation, deciding whether a major failure is in progress, or if a transient or an isolated failure has occurred, and then it will proceed to take the necessary actions.

The PPRC Consistency Grouping timer function gives time to the *automation* routines and functions to issue the necessary commands (when a pairing has an error) by putting the primary volume into a long busy state. This state is issued by the ERP and reflected in the `IEA494I` messages. The long busy state temporarily queues I/O to the affected primary volume; thus, dependent writes are held. On the ESS, the creation of Consistency Groups is controlled through the CGROUP parameter of the TSO CESTPATH command, or its equivalent options from the ESS Copy Services Web User Interface.

With the Consistency Groups created and the extended long busy window active, the automation functions can do a freeze to temporarily suspend I/O to all the necessary volume pairs at an LSS level. This ensures a point of consistency at the secondary site, across all necessary volumes.

Automated procedures should be set to ensure the consistency of the data across the application volumes in whatever LSS or ESS they reside. These procedures have to be able to act on the PPRC configuration according to the error.

Automation procedures can have a dedicated device per LSS as a utility device, which is used as the target of the automation triggered PPRC commands. Having a dedicated utility device ensures that the device is immediately available to receive commands, and there will not be reservation conflicts or deadlocks.

GDPS, which is discussed in Appendix B, “Geographically Dispersed Parallel Sysplex (GDPS)” on page 359, implements all necessary functions for an automatic recovery, with PPRC integrated into this solution.

### 3.5.1 PPRC Failover and Failback modes

The PPRC Failover and Failback modes are designed to help reduce the time required to synchronize PPRC volumes after switching between the application and the recovery sites.

In a typical PPRC environment, processing will temporarily failover to the PPRC secondary site upon an outage at the PPRC primary site. When the primary site is capable of resuming production, processing will failback from the PPRC secondary site to the primary site. Previously, synchronization of the PPRC volumes during a failback required a full establish, thereby causing all data to be transmitted from the secondary site to the primary site.

With the PPRC Failover Mode capability, when it is used during the failover to the secondary site, change recording is performed while processing is done on PPRC volumes at the secondary site. When the PPRC Failback Mode is used during the failback to the primary site, only changed data will be sent to the primary site to synchronize the PPRC volumes, thereby helping to reduce the time required to complete the failback.

The PPRC Failover and Failback modes can be invoked from the ESS Copy Services Web User Interface (WUI), the ESS Copy Services Command Line Interface (CLI), or by means of the ICKDSF utility.

An example of a planned outage procedure using the PPRC Failover and Failback modes is described in Chapter 8, “Solutions” on page 409.

## 3.6 Synchronous PPRC

Synchronous PPRC was often referred to as *PPRC* in earlier documentation. To avoid confusion, we now refer to this original mode of operation as Synchronous PPRC, and the term PPRC is used generically for all the methods of PPRC operation, including the asynchronous methods. This section discusses Synchronous PPRC.

## 3.6.1 How Synchronous PPRC works

PPRC is a hardware based solution that enables the mirroring of application system data from one site (the application site) and its associated DASD volumes (the primary volumes) to a second system at another site (the recovery site) and its DASD volumes (the secondary volumes).

Updates made on the primary DASD volumes are synchronously shadowed onto the secondary DASD volumes. The application write operation is only complete when the data is secure on the secondary site ESS. Figure 3-9 illustrates the sequence of a write update with PPRC (synchronous mode).

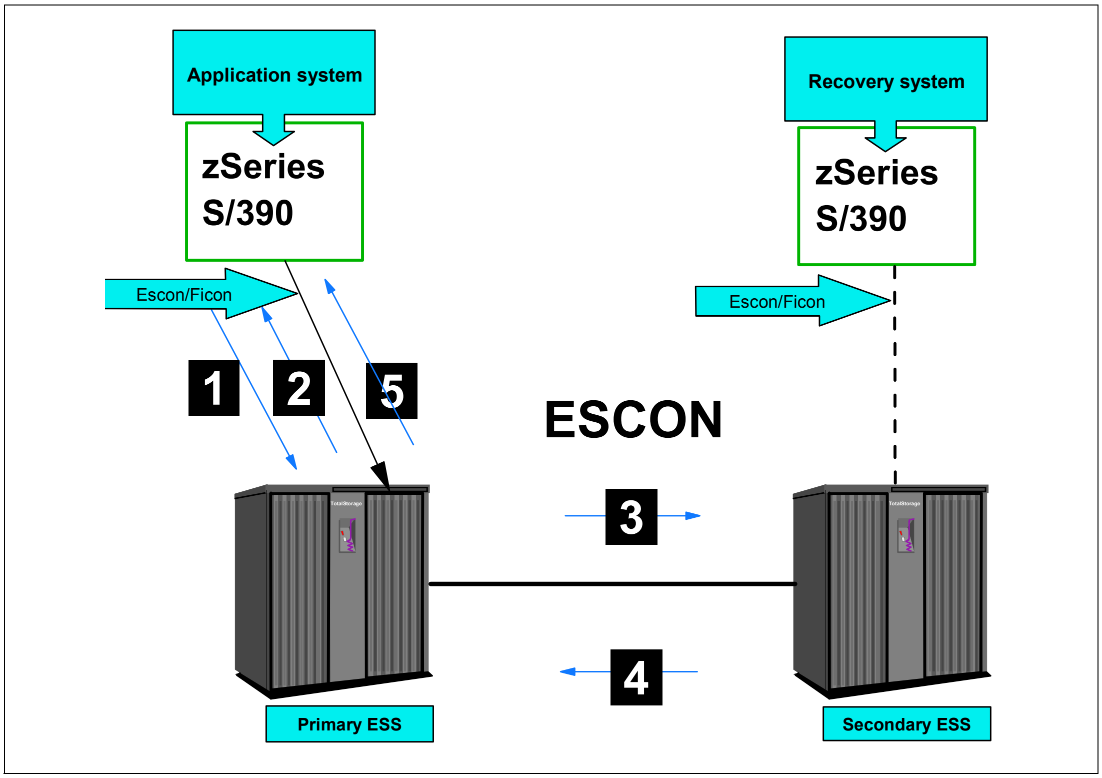



*Figure 3-9 Synchronous PPRC data flow*

When the application performs a write update operation to a primary volume, this is what happens:

1. Write to primary volume (ESS cache and NVS):

   The application system writes data to a primary volume on the ESS, and cache hit occurs.

2. Disconnect from channel (CE):

   Once the data has been transferred to the primary ESS's cache and NVS, the *channel end* status is issued to the application system.

3. Write to secondary (ESS cache and NVS):

   The application site ESS then initiates an I/O channel program to the recovery site ESS to transfer the updated data to the recovery site cache and NVS. This technique ensures that performance for the synchronous write is optimized.

4. Signal write complete on the secondary:

   The recovery site ESS signals write complete to the application site ESS when the updated data is in its cache and NVS.

5. Post I/O complete (DE)

   When the application site ESS receives the *write complete* from the recovery site ESS, it returns *device end* status to the application system.

   > **Note:** Destage (the movement of data from cache to the disk drives) on both the application and recovery site ESSs is performed asynchronously.

If, during the process, an error occurs that prevents the secondary volume from being updated, PPRC automatically suspends the mirroring function on that volume and a set of accompanying actions are triggered to ensure data consistency at the recovery site. The series of resulting actions will depend on certain critical attribute and consistency group options defined at implementation time (these recovery options are discussed later in this chapter).

## PPRC volume states

PPRC environments, as compared to non-PPRC environments, bring a new element into consideration: the *state* in which a volume is. In order to manage PPRC configurations, you need to know the state of the PPRC volumes.

This section reviews the states in which we may find PPRC volume pairs (refer to Figure 3-10) when operating in the Synchronous PPRC environment.

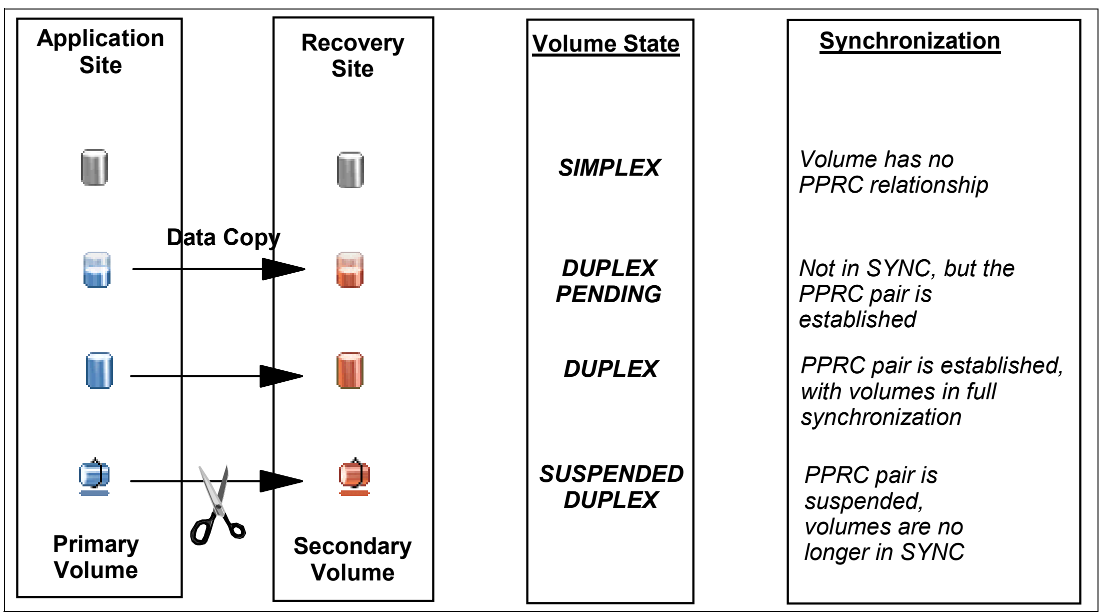



*Figure 3-10 Synchronous PPRC volume states*

As Figure 3-10 illustrates, at any given time a volume can be in any of the following states:

**Simplex** The initial state of a volume. A PPRC volume pair relationship has not been established yet between the primary and the secondary volumes.

**Pending** The initial state of a defined PPRC volume pair relationship, when the initial copy of the primary volume to the secondary volume is progressing. This state also is found when a PPRC volume pair is re-synchronized after it was suspended. During the *pending* period, the volume pair is not in synchronization, and PPRC is copying tracks from the primary to the secondary volume.

**Duplex** The state of a PPRC volume pair after PPRC has fully completed the copy operation of the primary volume onto the secondary volume. At this moment the volume pair is synchronized, and all write updates to the primary volume have been synchronously applied onto the secondary volume as illustrated in Figure 3-9 on page 92.

**Suspended** In this state of the PPRC pair, the writes to the primary volume are not mirrored onto the secondary volume. The secondary volume becomes out-of-sync. During this time, PPRC keeps a bitmap record of the changed tracks in the primary volume. Later, the volume pair can be re-synchronized, and then only the tracks that were updated will be copied.

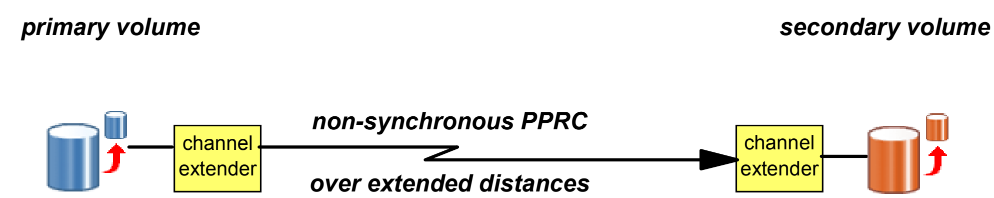
A PPRC volume pair will automatically go into suspended state, for instance, when the primary ESS cannot complete a write operation to the recovery system ESS. Also, the operator can suspend pairs by TSO commands, or by using the ESS Copy Services WUI.

# 3.7 PPRC-XD

PPRC-XD provides a non-synchronous mode of mirroring over extended distances.

## 3.7.1 Characteristics

The basic characteristics of PPRC-XD are summarized in Figure 3-11.




- XD option when establishing a volume pair
  - XD option (extended distance) in addition to the traditional synchronous option (SYNC)
  - CRIT setting not valid with XD option
  - Long distance implementations when using channel extenders
- Non-synchronous copy
  - Write updates to primary receive immediate completion
  - Primary ESS keeps record of updates to primary volumes
  - Incremental copy of primary updates periodically sent to secondary
  - Less demanding bandwidth requirements compared to SYNC option
  - No synchronous-like overhead on applications' response time
  - Potentially unlimited distance with channel extenders
    - Only limited by the channel extender and network technology capabilities
  - Maintains a fuzzy copy on secondary volumes until forced to synchronous
    - No certainty for dependent writes mirroring sequence

*Figure 3-11 Characteristics of PPRC Extended Distance*

These characteristics of PPRC-XD are discussed in the following sections.

## Non-synchronous option of PPRC

When doing the initial copy (establishing a new PPRC pair), or when re-synchronizing (RESYNC) a previously suspended volume pair, there are two options you can specify:

- **SYNC** This option is used for synchronous mirroring up to 103 km over ESCON and up to 300 km over FCP. In this method of PPRC operation, the initial copy of a pair is completed up to the *duplex state*, and from there on updates to the primary volumes are mirrored *synchronously* on the secondary volumes. A complete status is not returned to the application until both writes are done. As a consequence, there is an overhead which becomes progressively more significant at longer distances. At longer distances, the overhead in the I/O response time, which is typical of a synchronous operation, may not be acceptable for some applications.
- **XD** This option is used for *non-synchronous* transfer of updates over extended distances. In this method of PPRC operation, when doing the initial copy of a pair, or when a suspended pair is re-synchronized, the volumes do not reach the *duplex state,* but remain in *duplex pending*-XD state. While in duplex pending-XD state, the application updates to the primary volumes receive a complete status without waiting for the secondary ESS to acknowledge — as opposed to the Synchronous PPRC. The primary ESS keeps record of the updates to the primary volumes, and the updates are periodically sent in batches to the secondary volumes.

## Application backup and recovery

PPRC-XD continuously mirrors the primary volumes onto the recovery site secondary volumes while the application is running. This provides a *fuzzy* copy at the recovery site.

For application recovery implementations based on point-in-time copies of the data, you will have to plan for appropriate checkpoints to briefly *quiesce* the application I/O activity and synchronize the volumes pairs (catch-up). This way you can get a point-in-time copy with logical data consistency across volumes. In the following sections you will find examples for creating these consistent point-in-time copies of data with minimum application impact.

Application recovery implementations based on transmitting inactive database log files will not require application I/O quiescing.

If quiescing the application and then waiting for the volumes to synchronize is not acceptable, then the Asynchronous PPRC mode of operation should be considered. Refer to 3.9, "Asynchronous PPRC" on page 116.

## Write operations: Response time

PPRC-XD uses a non-synchronous mirroring technique, and so it does not incur synchronous-like overheads for the applications' I/O write operations. The application's write operations are finished as soon as the updates are secured in the primary ESS non-volatile storage (NVS) as in normal non-PPRC operation.

This characteristic makes PPRC-XD recommended for solutions needing a minimum impact on the application response time, especially when extended distances are in consideration.

## Distance factor

The non-synchronous technique of PPRC-XD relieves the application write operations from the typical synchronous mirroring overhead, which is directly proportional to the distance between the application and the recovery sites.

This characteristic makes PPRC-XD a recommended solution when planning for implementations over extended distances — continental distances, well beyond the synchronous 103 km ESCON (300 km FCP) supported distance. The distance is limited only by the capabilities of the network and the channel extenders' technologies.

These extended distances can be achieved using channel extenders over WAN lines or DWDMs using dark fiber. ESCON Directors can also be used for shorter distances. Please refer to 3.10, "PPRC links" on page 125 for additional distance and connectivity information.

### Fuzzy copy: Dependent writes

The PPRC-XD non-synchronous mirroring technique does not guarantee that application dependent writes are copied in the same sequence as they have been applied onto the primary volumes; the secondary copy is *fuzzy*. Dependent writes are explained in "Dependent writes" on page 86.

### CRIT attribute

As XD mirroring is done non-synchronously, with the secondary being a fuzzy copy, any CRIT setting is invalid for volume pairs established with the XD option.

### PPRC-XD positioning

The main points of PPRC-XD positioning are listed in Figure 3-12.

**PPRC Extended Distance is a recommended solution for remote data copy, data migration and offsite backup - over continental distances without impacting application performance.**

**It can be used for application recovery implementations if application I/O activity can be quiesced and non-zero data loss RPO is admissible.**

- It can be used over continental distances with excellent application performance
  - The distances only limited by the network and channel extenders capabilities
- Application write operations do not have synchronous-like overheads
- Fuzzy copy of data at the recovery site (sequence of dependent writes may not be respected at the recovery site)
- Recovery data can become a consistent point-in-time copy of the primary data, if appropiate application checkpoints are set to do global catch-ups
  - Pairs are synchronized with application group consistency
- Synchronizations can be done more frequently, because of short catch-ups
  - RPO still not zero, but improves substantially

*Figure 3-12 PPRC-XD positioning*

As summarized in Figure 3-12, PPRC-XD is a recommended solution for remote data copy, data migration, off-site backup, and transmission of inactive database logs, without impacting application performance, which is particularly relevant when implemented over continental distances. PPRC-XD can also be used for application recovery solutions based on periodic point-in-time copies of the data. This requires short quiescings of the application's I/O activity.

## 3.7.2 How PPRC-XD works

When a PPRC-XD relationship is first requested for a volume pair (that is, when the pair is initially established), a sequential copy of all the primary volume tracks is done. Then, with the primary ESS keeping a record of the updates done on the primary volumes, the updated tracks periodically will be sent to the secondary ESS. This is a throughput-oriented and very efficient way of non-synchronous mirroring (see Figure 3-13).

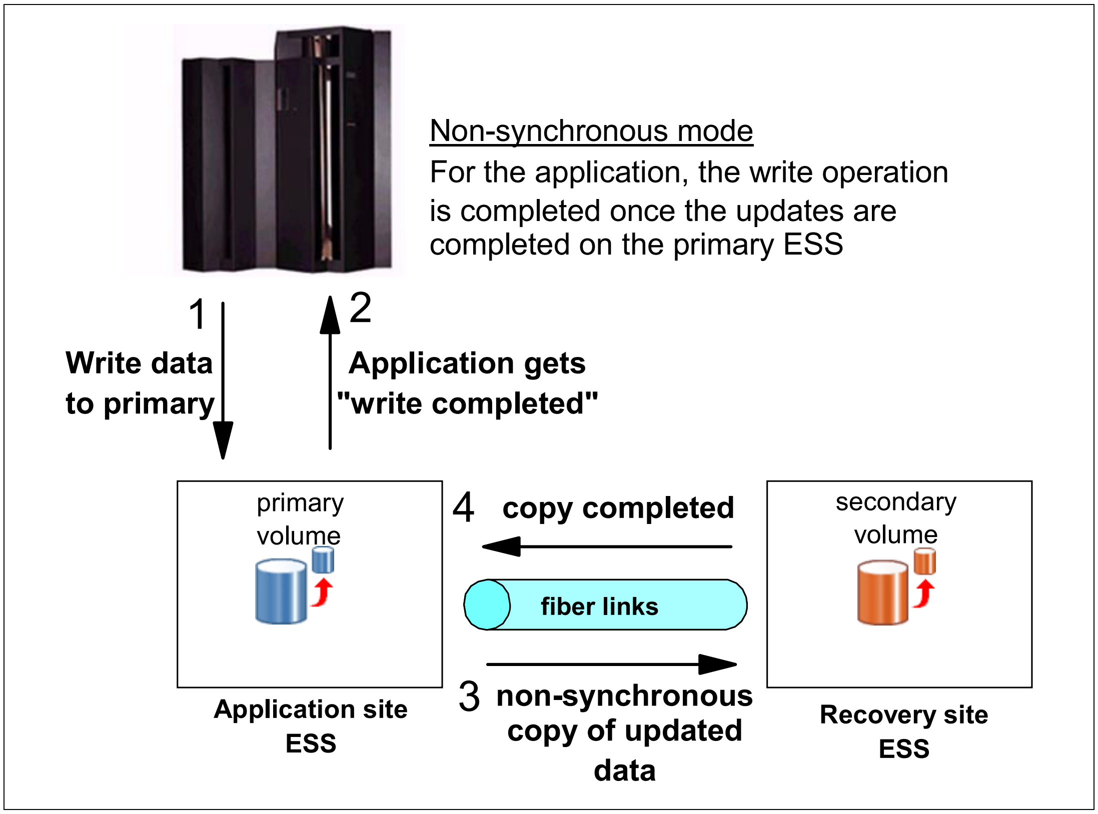



*Figure 3-13 PPRC-XD operation*

Figure 3-13 shows a simplified sequence of a PPRC-XD operation, illustrating its non-synchronous characteristic. The following is the sequence of the operation:

1. Host writes data to the primary volume. Data is secured in the primary ESS cache and NVS. Channel end (CE) is returned to host. The primary ESS keeps record of updated tracks for later transmission to the secondary ESS.
2. The primary ESS returns a *device end* (DE) to the host, so the I/O operation is complete for the application.
3. At some moment later, that is in a non-synchronous manner, the primary ESS will send the necessary data so that the updates are reflected on the secondary volumes. The updates are grouped in batches for efficient transmission.
4. The secondary ESS returns *write completed* to the primary ESS when the updates are secured in the secondary ESS cache and NVS. The primary ESS then resets its change recording information.

> **Note:** The efficient extended distance mirroring technique of PPRC-XD is achieved using sophisticated algorithms. For example, if changed data is in the cache, then PPRC sends only the changed sectors. There are also sophisticated queueing algorithms to schedule each volume for processing of their updated tracks; and then the setting of the batches of updates to be transmitted.

### 3.7.3 PPRC-XD volumes states change logic

PPRC-XD adds a new volume state, duplex pending-XD, and the specific change state logic associated with it. Please refer to Figure 3-14 for the following explanations:

- When you initially establish a PPRC pair relationship, you have the option to request that it become an XD pair (establish XD arrow in Figure 3-14), or a traditional fully duplexed pair (establish SYNC arrow in Figure 3-14).
- Pairs can change from the *duplex pending-XD* state, to the traditional *duplex* (SYNC) state when a go-to-SYNC is commanded (go-to-SYNC arrow in Figure 3-14).
- You can also request that a pair be *suspended* as soon as it reaches the full-duplex state (go-to-SYNC and SUSPEND in Figure 3-14).
- Pairs cannot change directly from SYNC state to XD state. They need to go through an intermediate suspended state.
- You can go from suspended state to XD state doing an incremental copy (copying out-of-sync tracks only). This is a similar process as for the traditional transition from suspended state to SYNC state (RESYNC/copy out-of-sync arrow in Figure 3-14).

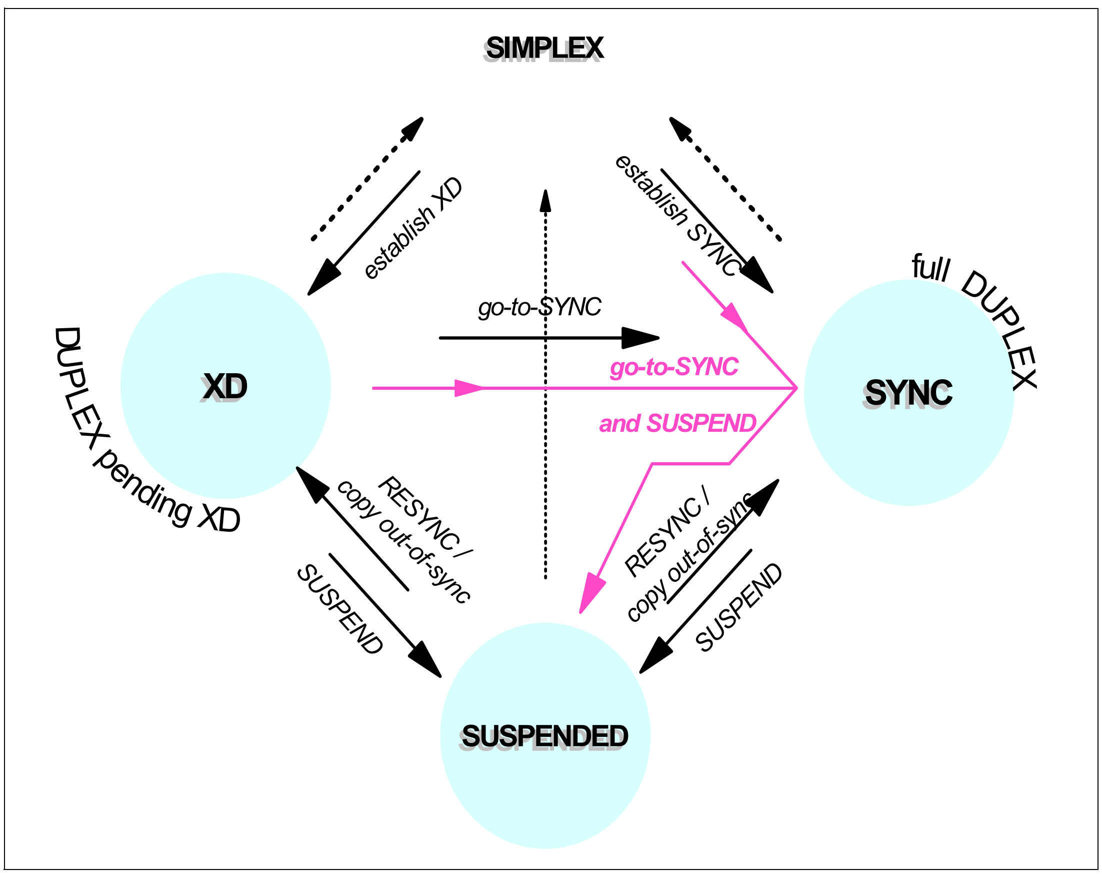



*Figure 3-14 PPRC volume state change logic*

The dotted lines in Figure 3-14 represent the termination of the pairs.

**Tip:** We recommend that when you terminate suspended pairs, always specify the primary LSS. If you specify the secondary LSS, then the primary volumes will remain in suspended state, because this will be interpreted by PPRC as a recovery operation.

When you are querying a primary volume:

- If the pair was established as a SYNC pair, then initially you will be informed of the *duplex-pending* state condition while the pair is synchronizing. Afterwards, when the pair has completed the initial synchronization, you will be informed of the *duplex* state condition (the volumes are fully synchronized). Information about the number of tracks or sectors out of synchronization (out-of-sync) is available while in duplex-pending state until duplex is reached.
- If the pair was established as an XD pair, then you will be informed of the *duplex-pending* XD state condition, and the volume will never reach the *duplex* state condition. The number of tracks or sectors out of synchronization (out-of-sync) is always displayed even if it is zero, because in fact the volumes never leave the duplex pending state.

## 3.7.4 Duplex-pending state

When in *duplex-pending XD* state, the secondary volumes have a *fuzzy* copy of the primary volumes. That is, there is no certainty that the updates on the secondary volumes are applied in the same sequence as on the primary volumes. Additionally, if primary updates are not quiesced then the pairs will typically never reach the 100% synchronization — full synchronization is a characteristic of the *duplex* state. Figure 3-15 summarizes the characteristics of the *duplex-pending XD* volume state.

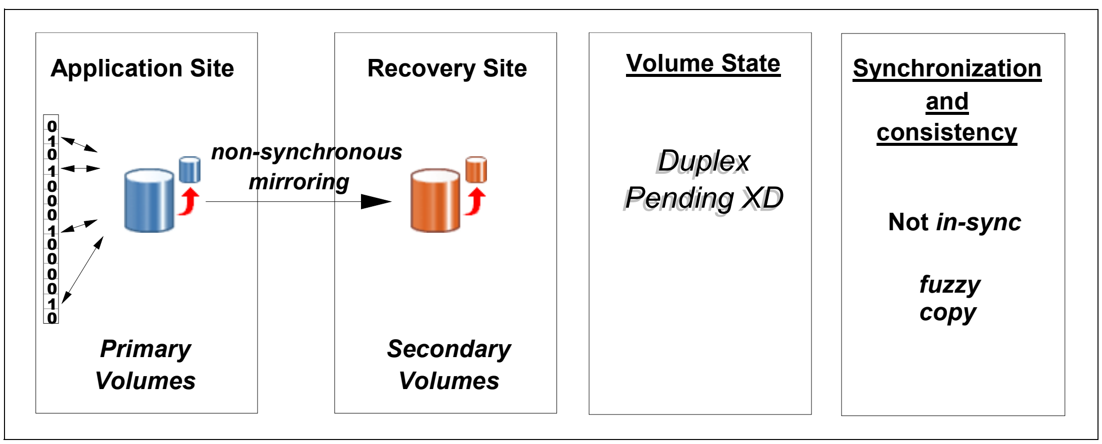



*Figure 3-15 Duplex-pending XD state*

When in a PPRC-XD relationship, the volume pair will remain in the *duplex-pending XD* state indefinitely. A specific command is required to change the state. The duplex-pending XD state can be changed to:

- **Full duplex state:** By commanding PPRC to do so (explained in “Go-to-SYNC” on page 100).
- **Suspended state:** By using the CSUSPEND TSO command, or the PPRCOPY SUSPEND ICKDSF command, or the equivalent ESS Copy Services WUI panel option.
- **Simplex state:** By using the CDELPAIR TSO command, or the PPRCOPY DELPAIR ICKDSF command, or the equivalent ESS Copy Services WUI panel option.

Figure 3-16 illustrates these basic characteristics of PPRC-XD, as well as the volume state transitions possible in a PPRC-XD environment.

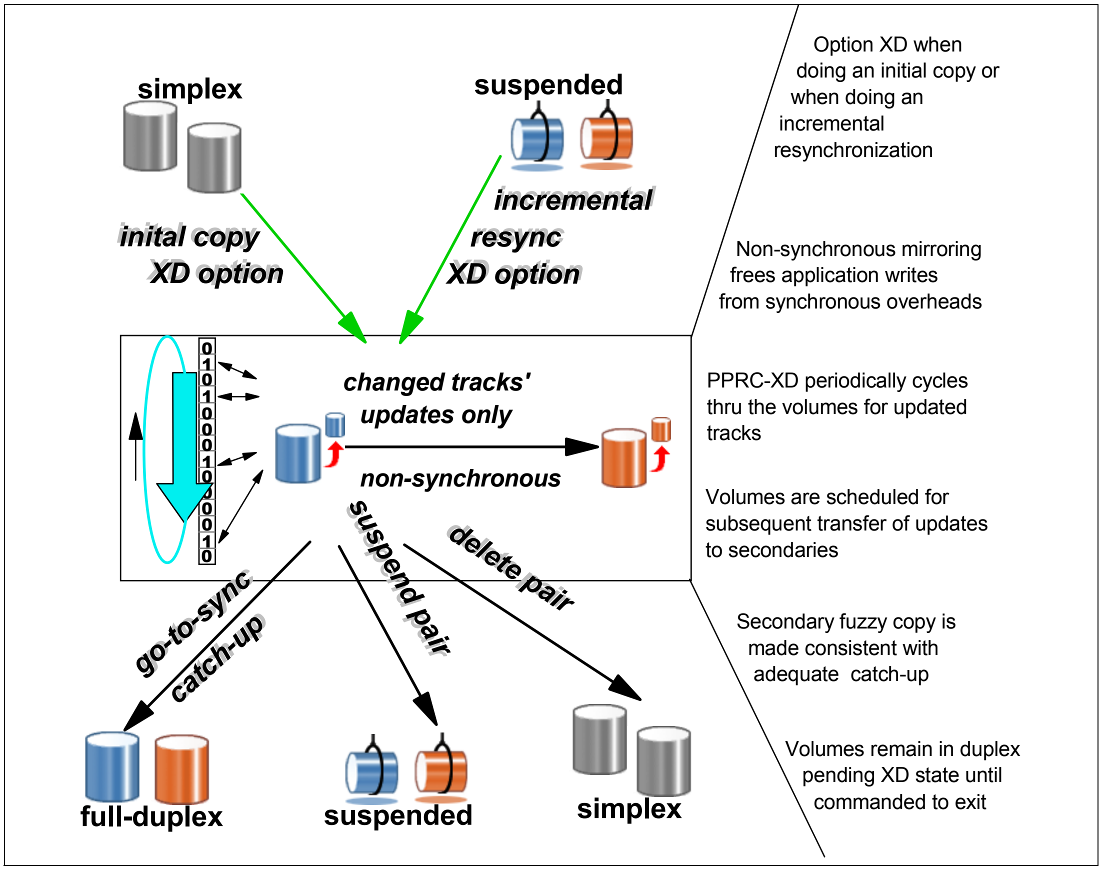



*Figure 3-16 PPRC-XD basic operation characteristics*

In the go-to-SYNC operation, PPRC does an incremental copy of changed track updates. This is a very efficient synchronization technique that minimizes the time needed for the catch-up transition.

Also, the catch-up transition can be accomplished by just temporarily quiescing the application writes to the primary volumes, and waiting for PPRC-XD to synchronize the pairs.

## 3.7.5 Catch-up transition

PPRC-XD *catch-up* is the name of the transition for a PPRC-XD pair when it goes from its normal out-of-sync condition until it reaches a full synchronous condition. In other words, the pair goes from the *duplex pending XD* state to the *duplex* state (*go-to-SYNC* as illustrated in Figure 3-14 on page 98). At the end of this transition, primary and secondary volumes become fully synchronized, with all their respective tracks identical.

### *Go-to-SYNC*

The *catch-up* transition can be achieved in two ways. One is a passive way, where an application quiesce is done, and you wait until there are no more writes onto the primary volumes. After a short time, PPRC-XD by its normal operation will bring primaries and secondaries to a full synchronization point.

The other way of accomplishing a *catch-up* transition is by commanding PPRC-XD to do so. To designate this way of catching up, we use the term *go-to-SYNC*.

When triggering a catch-up by commanding PPRC to go-to-SYNC, the `IEA494I` system message for z/OS and OS/390 systems will allow you to detect the moment when the volume pairs reach the duplex state — that is, when the catch-up is completed.

**Note:** The commanded catch-up transition (go-to-SYNC) should be triggered when the application write activity is low or preferably none. Also, it is better to trigger it with all the volumes at 90% synchronization or beyond.

## Using TSO for a go-to-SYNC

You can initiate a catch-up operation by commanding PPRC to go-to-SYNC using the TSO command CESTPAIR. For the go-to-SYNC, the CESTPAIR command must be used with the OPTION(SYNC) and MODE(RESYNC) parameters.

Table 3-3 summarizes the values of the CESTPAIR command parameters for the go-to-SYNC transition.

*Table 3-3 CESTPAIR parameter values for the go-to-SYNC transition*

| Operation | Volume pair state | | CESTPAIR parameter values | |
|---|---|---|---|---|
| | **from** | **to** | **OPTION** | **MODE** |
| go-to-SYNC catch-up | *duplex pending* XD | *duplex* | SYNC | RESYNC |

**Note:** The CESTPAIR command does not support the go-to-SYNC and suspend operation as the ESS Copy Services WUI does. When not using the Web interface, in order to suspend as soon as the duplex state is reached, you have to automate it by triggering on the state change messages that the system issues.

## Using the ESS Web User Interface for a Go-to-SYNC

You can initiate a catch-up transition by commanding PPRC to go-to-SYNC using the ESS Copy Services WUI panels, or by invoking a saved task using the ESS Copy Services command line interface, if a supported server is attached.

Table 3-4 summarizes the copy options that will be used in the ESS Copy Services WUI panels for doing a go-to-SYNC transition.

*Table 3-4 ESS Copy Services WUI options for the go-to-SYNC transition*

| Operation | Volume pair state | | WUI options | |
|---|---|---|---|---|
| | **from** | **to** | **Select task type** | **Select copy options** |
| go-to-SYNC catch-up and suspend | *duplex pending* XD | *suspended (as soon as the volumes reach duplex state)* | Establish Synchronous PPRC copy pair | Copy out-of-sync cylinders only<br>+<br>Suspend PPRC after establish |

### *Go-to-SYNC and suspend*

When doing the catch-up transition (from duplex-pending XD to duplex) by commanding PPRC go-to-SYNC, you may not want any synchronous copy operations to occur — especially if the mirrored volumes are separated by long distances, beyond 103 km (300 km). For this circumstance, the ESS Copy Services Web User Interface panels provide an option, when you select the copy options, that allows you to ask PPRC to suspend the pair as soon as it is established.

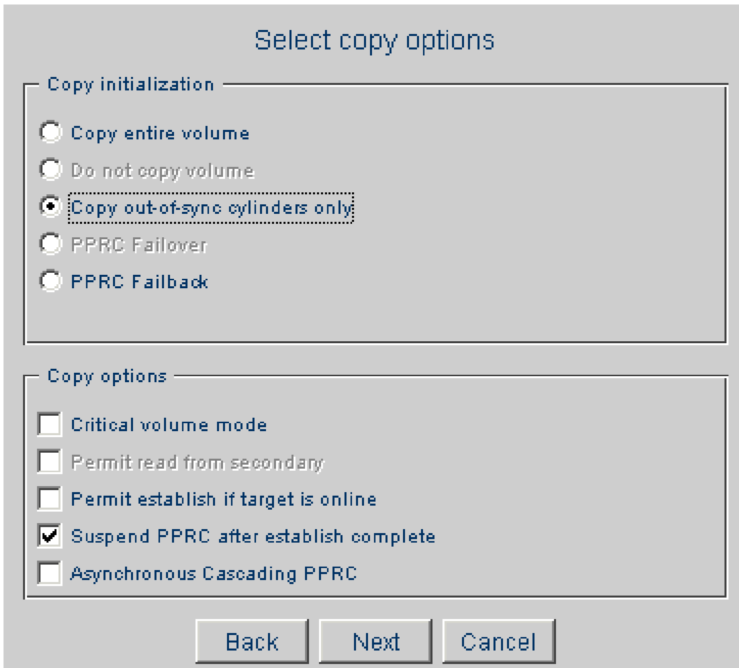



*Figure 3-17 Go-to-SYNC and suspend*

Figure 3-17 shows that the out-of-sync tracks from the primary will be copied to the secondary and the PPRC copy pairs will be suspended as soon as the duplex state is reached.

## Using ICKDSF for a Go-to-SYNC

You can initiate a catch-up operation by commanding PPRC to go-to-SYNC using the ICKDSF PPRCOPY ESTPAIR command. For the go-to-SYNC, the CESTPAIR command must be used with the OPTION(SYNC) and MODE(RESYNC) parameters

Example 3-1 shows the PPRCOPY ESTPAIR command to go from PPRC-XD mode to synchronous mode.

*Example 3-1 ICKDSF initiated go-to-SYNC transition*

```
PPRCOPY ESTPAIR UNIT(4C80) LSS(X'01',X'05') -
PRI(X'2901',FCA29,X'00') SEC(X'2905',FCA29,X'00') -
OPTION(SYNC)
```

## 3.7.6 PPRC-XD implementation considerations

The PPRC Extended Distance option provides additional possibilities for mirroring solutions. This unique technique also highlights some considerations that need to be looked at when planning its implementation. Here are some considerations:

- PPRC-XD provides theoretical unlimited distances with very little impact to the local host.
- Automation is recommended for efficient catch-up windows.
- Bandwidth management is more flexible.

#### *More data protected*

When planning your data integrity requirements, you now can think of more data being protected. PPRC Extended Distance has less bandwidth demand, and minimum application performance impact — independent of the implemented distance. This allows you to include more of your data under the PPRC protection range.

#### *Consistency management*

With PPRC Extended Distance, the management of consistency is done differently than with Synchronous PPRC. With PPRC-XD you now have remote fuzzy copies of the primary volumes, thus the dependent writes on the remote volumes are not necessarily applied in the same sequence as they were done by the application on the primary volumes.

This characteristic makes you plan for the applications recovery implementations in a different manner. For example, you have to plan for checkpoints to build consistency at the recovery site data; whereas in a synchronous implementation, you will not need these checkpoints.

#### *Bandwidth management*

PPRC Extended Distance uses the available bandwidth in an adaptive and flexible way, efficiently exploiting the available resources. You should plan the bandwidth based upon the amount of write operations, and the minimum time needed for the go-to-SYNC transition.

### 3.7.7 Data consistency with PPRC-XD

The XD option brings more functionality into the PPRC environment, changing the paradigm that PPRC is only synchronous. PPRC-XD will be used in different implementation scenarios than those of traditional Synchronous PPRC.

#### *PPRC-XD point-in-time consistency*

With PPRC-XD, the secondary volumes are a *fuzzy copy* of the primary volumes. This fuzzy condition of the secondary volumes demands that in order to use them for application recovery you need to make them point-in-time consistent at both the individual volume level and across the volumes that the application uses.

#### *Global application level consistency*

Consistency at the volume level means that at a selected point-in-time the secondary volume mirrors exactly what the primary volume was at that point-in-time. Consistency at the application level means logical *global consistency* across all the volumes upon which the application does updates. This is to say that for a selected point-in-time, the whole set of application related secondary volumes will mirror exactly what the primary set of application related volumes were at that point-in-time.

#### *Data consistency at the recovery site*

Lets examine the considerations that come into play in a PPRC-XD environment, when the consistency of the secondary data is considered.

While in the *duplex-pending XD* state, and with the applications doing updates on the primary volumes, the recovery site volumes are keeping a *fuzzy copy* of the data. This means that during this state, there is no certainty that the application *dependent writes* will be applied to the secondary volumes in the same sequence as they are written to the primary volumes.

Also, because of the non-synchronous characteristic of PPRC-XD at any time there will be a certain amount of application updates that will not be reflected at the recovery site secondary volumes. These are the out-of-sync tracks.

The number of out-of-sync tracks can be queried with the CQUERY command, or by clicking the **Information Panel** button for a selected pair in the Volumes panel of the Web interface.

The *catch-up* transition allows you to synchronize the volume pairs in a minimum interval of time. Upon reaching duplex state, the volume pairs can be temporarily suspended, followed by a FlashCopy of the secondary volumes onto tertiary volumes, and then resuming the PPRC-XD relationships.

With PPRC-XD you can have application *global consistency* — across the secondary copies of the application set of volumes — if you implement the appropriate checkpoint activities in order to build this logical consistency. The *catch-up* transition will be part of this checkpoint procedure.

When planning the recovery of an application using a point-in-time copy of the data, you must remember that while in an active PPRC-XD relationship, the secondary volumes always have a fuzzy copy of the primary volumes. So, for a valid application recovery you must obtain the tertiary volumes where you did the last FlashCopy from secondary volumes having a global consistent checkpoint. As you can realize, this tertiary copy of the data does not reflect the updates to the primary volumes done after the last global *catch-up* operation.

## 3.7.8 Global catch-up

The following sections present examples on how to use the *catch-up* procedure to get point-in-time copies of the data at the recovery site with consistency across all the volumes (*global catch-ups*) from the application perspective (see Figure 3-18).

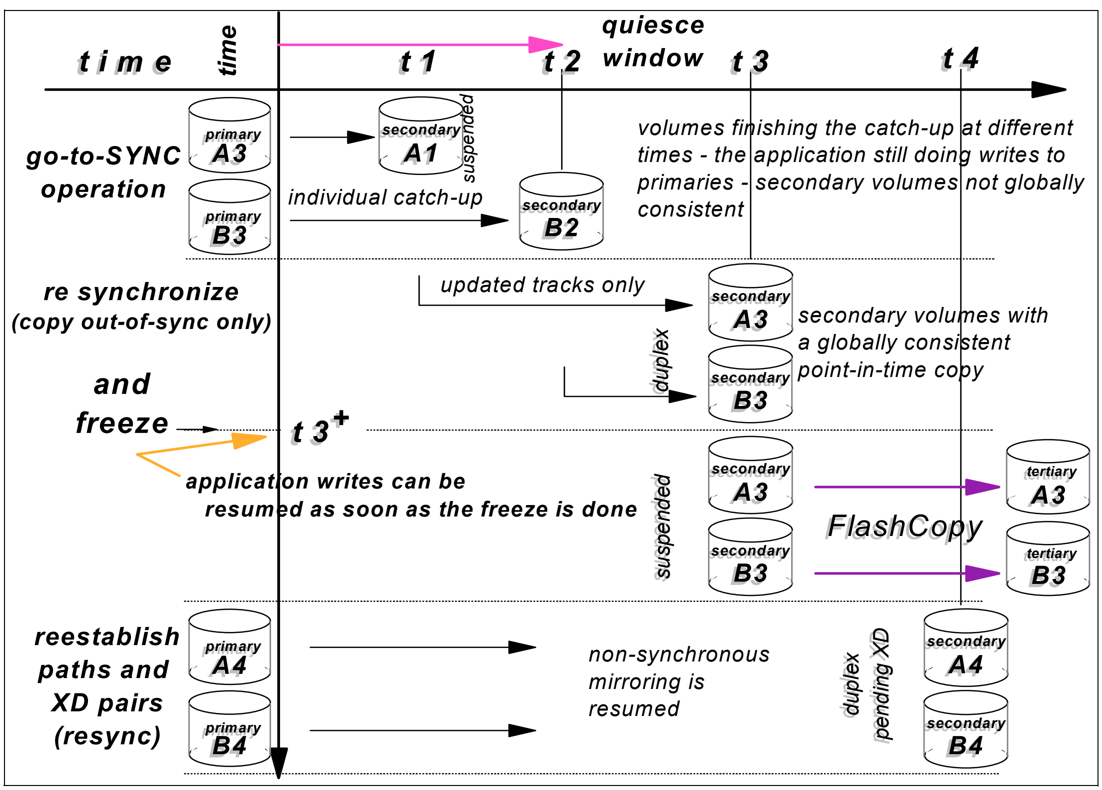



*Figure 3-18 Consistent point-in-time copy - Starting with a go-to-SYNC*

Please refer back to Figure 3-18 on page 104 to follow the explanation presented in this section. As the application is running and making updates onto the primary volumes, you may begin the catch-up by first doing a *go-to-SYNC and then suspend* the set of volumes where the application is updating data. This step should be done during a very low (or none) application write activity window. All the volumes should be stable and not in catch-up mode.

### Go-to-SYNC and suspend

In order to suspend immediately upon reaching the duplex state to avoid any synchronous update over the long distance, you can use the option **Suspend PPRC after establish** if you are working with the Web User Interface.

For TSO users, the *go-to-SYNC and suspend* operation should be automated. Because the volumes will be transitioning — from the duplex pending XD state to the duplex state — this change of status will be notified to the host with the `IEA494I` message. These messages can be intercepted and acted upon by automation routines to suspend the volumes as they reach the duplex state.

Upon finishing this initial *go-to-SYNC and suspend* step, you have a set of suspended secondary volumes, which individually have a consistent copy of their primary suspended counterparts. But as you can see from Figure 3-18 the point-in-time is not the same *across* all the secondary volumes. The application was doing write updates to the primary volumes and some of these updates — as the volumes were individually finishing their catch-up and reaching the suspended state at different moments — were not mirrored, but recorded by the primary ESS for later re-synchronization.

### Quiesce application writes

In order to reach global consistency across the secondary volumes, you should quiesce the application writes. This quiesce will allow for a succeeding re-synchronization of the volumes, but this time reaching a point-in-time with global consistency.

The need of a quiesce, in order to build consistency at the recovery site, is a trade off against the write performance penalty inherent to a long distance synchronous implementation, which does not require the quiesce.

> **Note:** At Synchronous PPRC supported distances, this quiesce could be skipped. Refer to“Point-in-time within synchronous range” on page 107 for further explanation.

### Build consistent point-in-time and restart application writes

With the application not doing any writes, you now re-synchronize (**`copy out-of-sync tracks only`** / **`RESYNC`**) the suspended pairs.

One alternative could be to re-synchronize the pairs, and then suspend them again immediately upon reaching the duplex state. This operation can be implemented by creating a major task that groups several individual suspend tasks, or by creating a TSO command list that executes several individual suspend commands — one for each of the volume pairs in consideration.

A more efficient way of doing this is to re-synchronize the pairs, and once they reach the duplex state, then freeze the respective LSS. This will simultaneously suspend all the volume pairs within the affected LSS, and also terminate the paths between the LSSs.

Either of the two alternatives leaves the secondary volumes with a globally consistent point-in-time copy of the primary volumes, and also, both alternatives leave the volumes suspended. This suspended state enables you to restart the application write activity on the primary volumes, while you FlashCopy the secondary volumes that now have a consistent point-in-time copy.

#### *Restart application writes*

The application write activity can be restarted as soon as the freeze is done. These updates will not be propagated to the secondary volumes — as volume pairs are suspended and LSS paths are discontinued. In the meanwhile, PPRC will keep track of the updated tracks on the primary volumes to be able to do an incremental re-synchronization when the XD pairs are later reestablished.

#### *FlashCopy consistent checkpoint data*

Once the FlashCopy relationship is established, you can bring the primary and secondary volumes out of the suspended state. Bringing them into the duplex-pending XD state — an incremental re-synchronization (copy out-of-sync only) — will resume the mirroring on the secondary volumes.

Before reestablishing the volume pairs' XD relationships, you must redefine the LSS paths that were terminated by the freeze operation.

Now the procedure is complete, the consistent point-in-time copy is on the tertiary volumes, and we are ready for the next checkpoint.

### 3.7.9 Managing PPRC-XD recovery

When PPRC-XD is used for application recovery solutions, automated procedures can be implemented in order to efficiently execute the steps involved in building global application consistency at the recovery site. The following events that occur when a consistent checkpoint must be taken can be automated:

- The quiesce of the application write activity and the following resumption
- Managing the go-to-SYNC transition window, where a quiesce of the application write activity is traded for a slowdown of the application I/O performance — if the ESSs are within the synchronous supported distance

In z/OS and OS/390 environments, when automating, you will find the state change messages that the system issues when volumes transition from one PPRC state to a different one very useful. The **IEA494I** and **IEA491E** messages can be detected by automation routines and used to suspend secondary PPRC operations.

When scheduling the quiesce of your application writes you will also need to consider some lead time to build consistency of the recovery data. This planned outage can be minimized when properly automated, and the whole process can prove to be very efficient when recovering an application after its outage.

When making your secondary volumes fully synchronous with their respective primaries, the commanded go-to-SYNC approach is not the only one you can use. As already mentioned, you can just quiesce the application writes and let the PPRC-XD normal operation do the catch-up, while you query PPRC to check for no more out-of-sync tracks.

**Note:** The following sections present possible setups for getting a consistent point-in-time copy of your data at the recovery site. Nevertheless, keep in mind that the optimum setup (variations of steps, sequence and timings) will be application dependent, and you will best be able to determine them by testing when doing your actual implementation.

### Point-in-time within synchronous range

At Synchronous PPRC supported distances, the previous procedure can be done differently, skipping the quiesce and the re-synchronization steps. If you temporarily trade-off some application write performance, then the procedure would be limited to the following actions:

1. Trigger a catch-up commanding PPRC-XD to go-to-SYNC but *not suspending*. This should be done when the application write activity is very low, and volumes are at least 90% in-sync; the catch-up will be short and the application write performance will have less impact.
2. Detect when *all* volumes are in duplex state — secondary copies become globally consistent. Queries and detecting `IEA494I` messages will give this information.
3. When the volume pairs reach the duplex state, do a freeze.
4. Now you have a secondary consistent point-in-time copy, that you can FlashCopy. Also, at this moment, the application is already relieved of the write penalty because volumes are suspended because of the freeze.
5. Finally, you can reestablish the paths, reestablish the XD pairs, and the non-synchronous mirroring starts again until the next catch-up.

For this particular implementation, the distance must be less than 103 km ESCON (300 km for FCP).

### Initial quiesce of the application

As already mentioned, you can set up different procedures in order to build global consistency in the data at the recovery site. At the time of actual implementation at your installation, you will be testing to determine which is the most efficient and the less disruptive implementation.

The example presented in Figure 3-19 illustrates an alternate procedure where you first start by waiting for the application writes to quiesce completely, and then let PPRC-XD complete the catch-up of the volumes on its own.

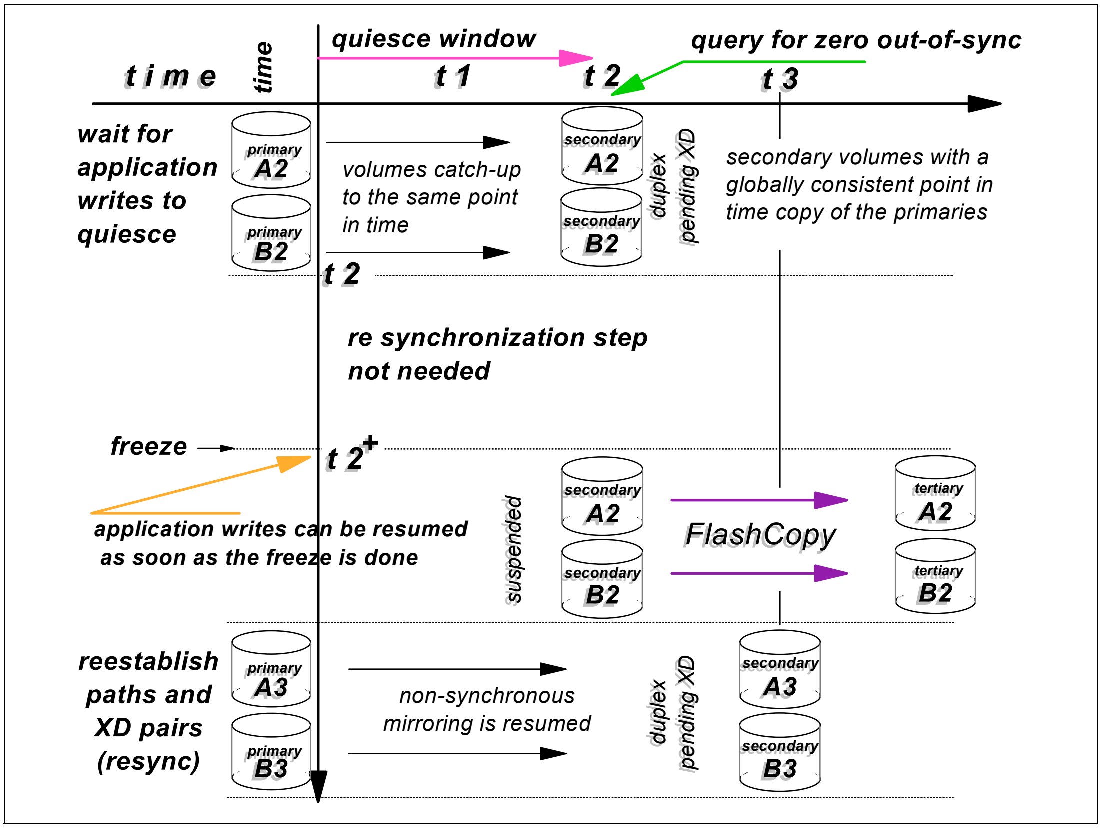



*Figure 3-19 Consistent point-in-time copy - Waiting for application writes to quiesce*

**Tip:** There will be no `IEA494I` messages when the volumes reach 100% synchronization because you have not requested any volume state change.

The more granular PPRC queries will allow you to determine with more accuracy, when there are no more out-of-sync tracks. At this moment, your secondary volumes hold a point-in-time copy of the primary application volumes with global consistency.

As you see in this example, because PPRC-XD puts the volumes in a 100% synchronization state, there is no need to re-synchronize the pairs, as in the previous example. You proceed to freeze the LSSs, which still have the volumes in duplex pending XD state. Also, upon completing the freeze, you proceed to do a FlashCopy onto the tertiary volumes.

As soon as the FlashCopy relationships are established, you can now re-initiate the XD mirroring onto the secondary volumes. To do so, you reestablish the paths and the pairs.

## 3.8 Asynchronous Cascading PPRC

This section introduces and describes the characteristics of the Asynchronous Cascading PPRC implementation that became available with LIC level 2.2.0 and requires the PPRC V2 feature. This information should be read in conjunction with the information in 3.6, “Synchronous PPRC” on page 91, and 3.7, “PPRC-XD” on page 94.

## 3.8.1 How Asynchronous Cascading PPRC works

PPRC Version 2 allows a PPRC secondary volume to become also a PPRC primary of another PPRC relationship, thus enabling an *Asynchronous Cascading PPRC* implementation as illustrated in Figure 3-20.

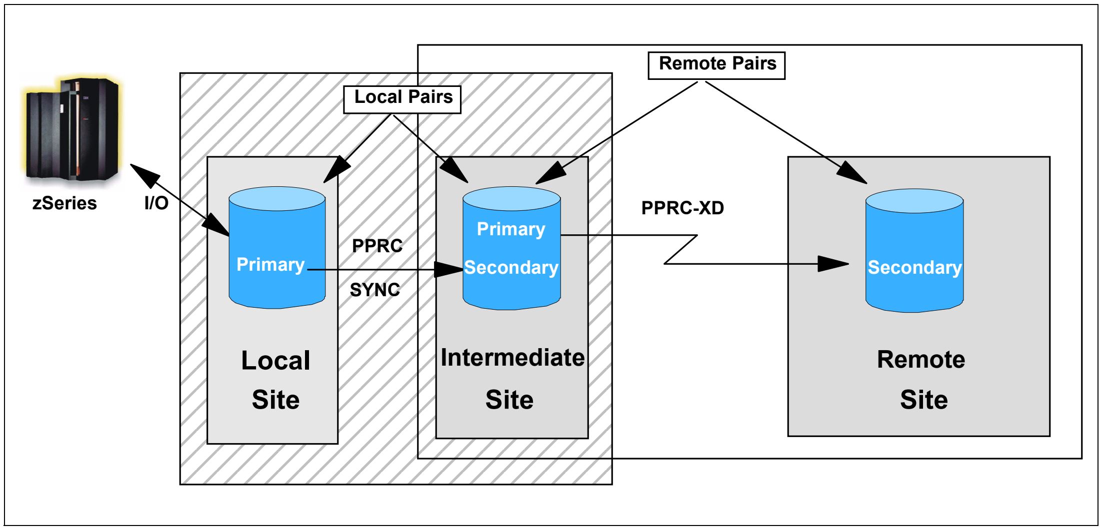



*Figure 3-20 Asynchronous Cascading PPRC - topology*

This is a data replication solution that will typically copy data synchronously (PPRC-SYNC) from the local site volumes to the intermediate site volumes, and from there the data is copied non-synchronously (PPRC-XD) to the remote site volumes. Asynchronous Cascading PPRC enables long distance disaster recovery solutions.

PPRC Version 2 enables the creation of two-site or three-site Asynchronous Cascading PPRC configurations.

In a three-site configuration, the Synchronous PPRC relationship is maintained between an ESS at the local site and an ESS at an intermediate site (located within 103 km for ESCON or 300 km for FCP). A PPRC-XD relationship is simultaneously maintained between the ESS at the intermediate site and an ESS at the remote site. The remote site can be located at continental distances well beyond the metro distances required for the local to the intermediate connection, with the distance typically limited only by the capabilities of the network and channel extension technologies.

When used with specific operational procedures, a three-site configuration can be designed to provide a data protection solution in the event of an unplanned outage at any one of the three sites. Additionally, the creation of a safety copy at the remote site using FlashCopy can be used to design a solution that provides a consistent copy of the data at the remote site in the event of a regional outage affecting both the local and intermediate site.

In a two-site configuration, the Synchronous PPRC relationship is maintained between two ESSs at the local site. As for the three-site configuration a PPRC-XD relationship is simultaneously maintained to an ESS at the remote site. While this does not offer the same level of protection against a total primary site failure, it does protect against failure of the primary or intermediate ESSs and with the automated use of FlashCopy at the remote site, point-in-time recovery can be achieved to recover from a total primary site loss. The advantage is the elimination of the need for the intermediate site.

## Operational characteristics

In a typical implementation, the local pairs are in a PPRC-SYNC relationship, while the remote pairs are in a PPRC-XD relationship. As in a normal PPRC-SYNC operation (described in “How Synchronous PPRC works” on page 92), the intermediate site ESS signals write complete to the local site ESS when the data is in its cache and NVS.

The updates on the intermediate volumes, which are also the primary volumes in a PPRC-XD relationship with the remote site volumes, are then automatically mirrored non-synchronously to the remote site ESS. Since this is done non-synchronously, the application I/O performance is not affected at all, it is independent of the distance.

The local and intermediate volumes can reside in the same physical location on different ESSs or even in the same ESS. However, a better solution is to have the intermediate site ESS within Synchronous PPRC distance of the application site ESS, such that a disaster at the local site ESS would be unlikely to affect the intermediate site ESS. Examples of Asynchronous Cascading PPRC solutions are discussed in Chapter 8, “Solutions” on page 409.

## Volume states

Asynchronous Cascading PPRC environments introduce a new state in which we may find a volume. In this section we review the PPRC volume states that we find in the Asynchronous Cascading PPRC environments.

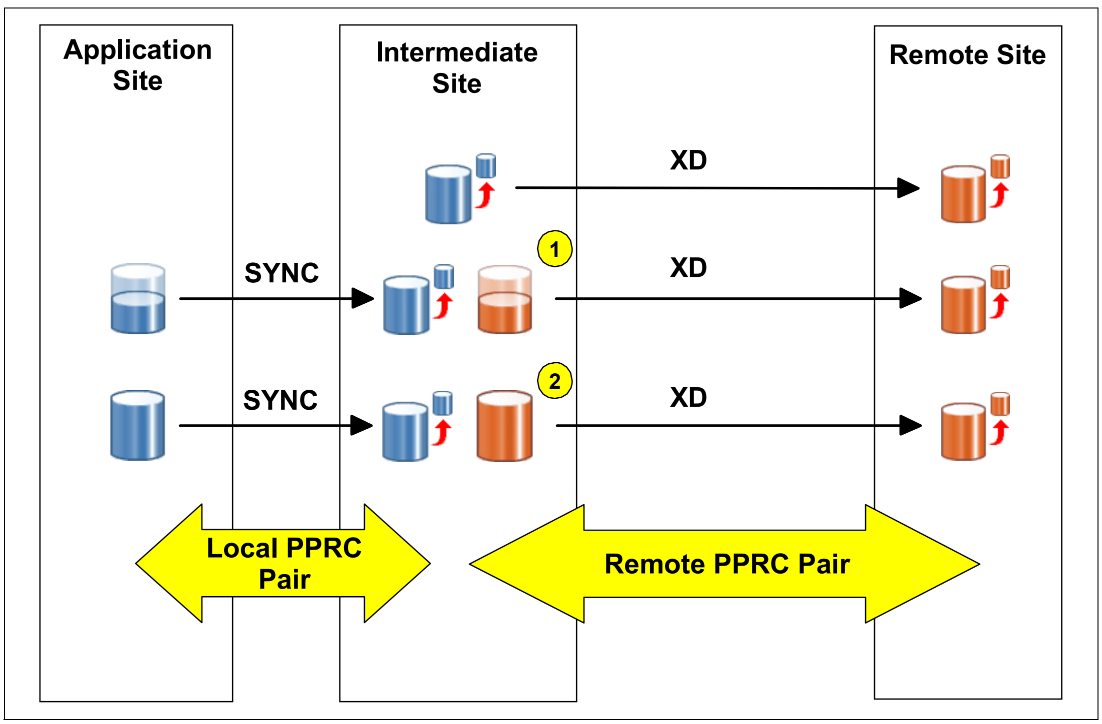



*Figure 3-21 Intermediate site PPRC volume states*

The intermediate site volumes, being at the same time the PPRC secondaries in the local PPRC-SYNC pair relationship and primaries in the remote PPRC-XD pair relationship, can present now two volume states combinations. The combined volume states, illustrated in Figure 3-21 with the icons that the ESS Copy Services Web User Interface uses to represent them, are as follows:

1. Duplex-pending as a local pair, PPRC-SYNC secondary, in combination with duplex pending-XD as a remote pair, PPRC-XD primary.

   The intermediate site volume is not yet 100% in synchronization with the local site primary volume, and PPRC is copying tracks from the local site PPRC primary volume to the intermediate site volume. At the same time, the data is copied non-synchronously to the remote site secondary volume. While in this volume state both the intermediate site volume and the remote site volume, hold an unusable copy of the data.

2. Full duplex as a local pair PPRC-SYNC secondary, in combination with duplex pending-XD as a remote pair PPRC-XD primary

   This is the state of the intermediate volumes after PPRC has completed the initial copy operation (or a re-synchronization) of the local PPRC pair. At this moment the local volume pair is 100% in synchronization. The remote site volume, being in a PPRC-XD relationship, is holding a *fuzzy* copy of the primary local volume.

### Allowed configurations

The typical implementation of a cascaded solution will have the local pair in a PPRC-SYNC relationship, while the remote pair is in a PPRC-XD relationship. This provides a long distance solution without affecting the application performance. Still, other combinations are allowable.

These are the possible combinations:

- Local pair SYNC → remote pair XD:

  This is the recommended cascading PPRC configuration (see Figure 3-20 on page 109). There is a Synchronous PPRC relationship from the local site to the intermediate site, and then a non-synchronous PPRC-XD relationship from the intermediate site to the remote site.

  In case of an outage at the local site, the intermediate site holds a valid copy of the data. A short time after the outage, the intermediate site will complete sending the updates to the remote site, and then both the intermediate and the remote site will have a valid copies of the data up to the time of outage.

- Local pair XD → remote pair XD:

  In this implementation both the local PPRC pairs and the remote PPRC pairs are in a non-synchronous PPRC-XD relationship. The intermediate site and remote site volumes hold fuzzy copies of the local site volumes.

- Local pair SYNC → remote pair SYNC:

  In this implementation both the local PPRC pairs and the remote PPRC pairs are in Synchronous PPRC relationships. With this implementation the remote site volumes hold a real time copy of the local site volumes, at all times. However, this configuration is limited in the maximum distance that can be used. Also, the application I/Os can be affected in an unacceptable manner, because each write I/O has to wait for two PPRC data transfers before its completion.

- Local pair SYNC → remote pair XD periodically switching to local pair SYNC → remote pair SYNC combination:

  In this implementation there is a SYNC → XD configuration running, which periodically is changed into a SYNC → SYNC configuration. This forces the remote PPRC pairs to go-to-SYNC, this temporary state provides a consistent point-in-time copy of the data at the recovery site allowing consistent FlashCopies to be made. In a long distance implementation, the local PPRC pairs should be frozen for the remote pairs to catch up. Alternatively, the application can be stopped or its I/O quiesced.

## Setting up the cascading environment

It is important to define the procedure that will be used to set up the Asynchronous Cascading PPRC configuration. This section discusses the considerations that can help you plan the setup procedure for the cascading PPRC configuration. In this discussion, the PPRC primary volumes are referred to as A volumes, the PPRC secondary volumes are referred to as B volumes, and the PPRC-XD secondary volumes are referred to as C volumes.

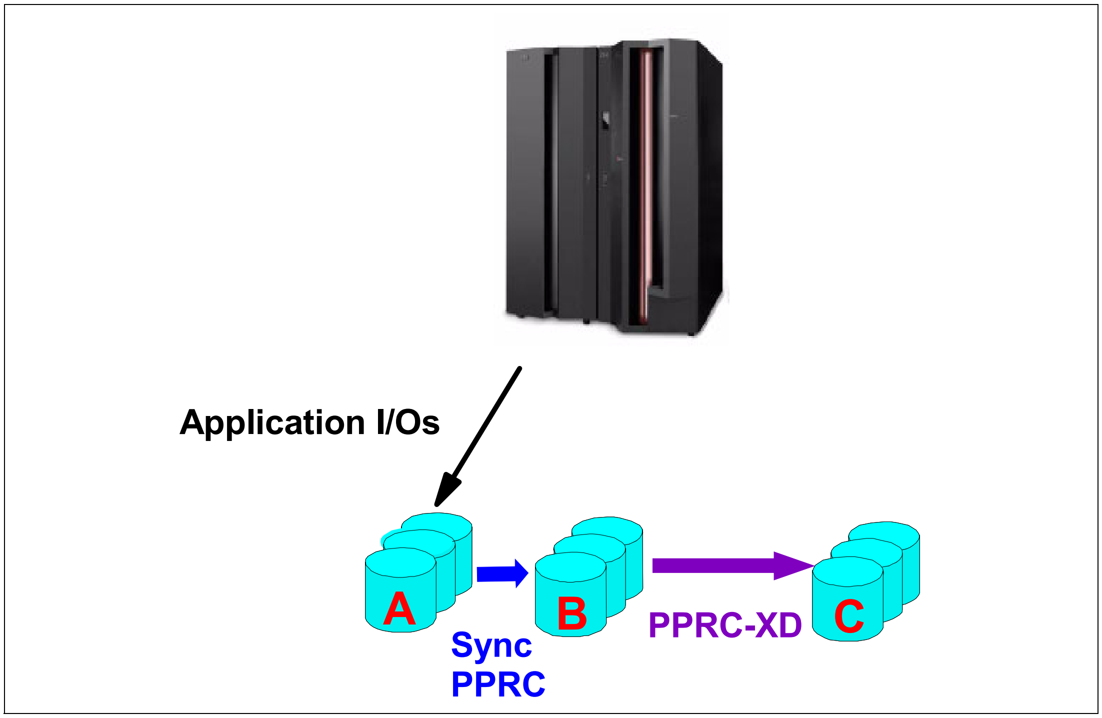



*Figure 3-22 Asynchronous Cascading PPRC*

### *Establishment order*

There are two valid ways of establishing the Asynchronous Cascading PPRC configuration:

The recommended sequence is as follows:

1. Establish the PPRC paths A → B.
2. Establish the XD paths B → C.
3. Establish the XD pairs B → C.
   – Establish the remote PPRC pairs in XD mode.
   – Use the NOCOPY option via the CESTPAIR command or the equivalent option of the ESS Copy Services Web User Interface.
   – Specify the CASCADE option.
4. Establish the PPRC pairs A → B.
   – Establish the PPRC pairs in PPRC mode.
   – Use the COPY option via the CESTPAIR command or the equivalent option of the ESS Copy Services Web User Interface.

Note that this sequence establishes the B → C pairs with the NOCOPY option and does the actual copy as the A → B copy changes occur. This is the quickest methodology for A → B to be in full duplex and B → C to be in *near* full duplex.

Alternatively, you can use the following sequence:

1. Establish the PPRC paths A → B.
2. Establish the XD paths B → C.
3. Establish the PPRC pairs A → B.
   – Establish the PPRC pairs in PPRC mode.
   – Use the COPY option via the CESTPAIR command or the equivalent option of the ESS Copy Services Web User Interface.
4. Wait until A → B volumes are in full DUPLEX.
5. Establish the XD pairs B → C.
   – Establish the remote PPRC pairs in XD mode.
   – Use the NOCOPY option via the CESTPAIR command or the equivalent option of the ESS Copy Services Web User Interface.
   – Specify the CASCADE option.

Note that this sequence establishes the B → C pairs with the COPY option after A → B is in full DUPLEX. If the establishment of B → C pairs is started after A → B but before A → B becomes full DUPLEX, then some tracks will be double copied.

## 3.8.2 Making volumes eligible for cascading

A volume pair can be made eligible for cascading at the time the pair is initially established or at a later time. What this means is that when a PPRC pair is initially established, the user can indicate whether the pair is eligible for cascading or not.

Alternatively, a PPRC pair which was initially established without the cascading option can later be made eligible for cascading by reestablishing the relationship — this time with the cascading option selected — and without needing to delete the pair.

### The CASCADE option

**Note:** The CASCADE option is always set on the remote PPRC pair, when establishing the PPRC relationship between the intermediate site and the remote site volumes.

The following examples illustrate three different scenarios using an Asynchronous Cascading PPRC environment and where and how the CASCADE option is used. Site A is the local site, site B is the intermediate site and site C is the remote site:

- **Example 1:** The site B to site C remote pair is initially established with the CASCADE option, so that the volumes at the intermediate site B are eligible for cascading. After the site A to site B volumes relationships are established, this three-site configuration becomes ready for the Asynchronous Cascading PPRC.
- **Example 2:** The site B to site C remote pair was initially established without the CASCADE option, so the volumes at the intermediate site B were not eligible for cascading. To set up the cascading configuration you have to establish the remote PPRC pairs again, but this time with the CASCADE option, but without needing to previously delete or suspend the pairs. After this, you can proceed to establish the local PPRC pairs from site A to site B, so that this three-site configuration becomes ready for the Asynchronous Cascading PPRC.

- **Example 3:** The site A to site B relationships are already established. In this situation, to configure the Asynchronous Cascading PPRC environment, you proceed to establish the site B to site C relationships with the CASCADE option, and without having to either suspend or delete the already established site A to site B local pairs. This three-site configuration now becomes ready for the Asynchronous Cascading PPRC.

### 3.8.3 Consistent point-in-time copy

This section discusses a procedure that can be used for obtaining a consistent point-in-time copy of the data at the remote site (site C) in an Asynchronous Cascading PPRC configuration.

Data consistency across all secondary volumes spread across any number of LSSs is essential for data integrity, and to enable a normal database restart in the event of a disaster. Data consistency on the primary volumes is managed by the applications, so that a consistent set of application data on disk is maintained if a failure occurs in the host processor, software, or storage subsystem. Applications, where one write is dependent on the completion of another, are said to have *dependent writes* (see discussion in 3.5, “Data consistency in a PPRC environment” on page 86).

One important consideration about PPRC-XD is that it requires the periodic creation of consistent point-in-time copies at the remote site, if the data is to be used for application recovery. This has a correlation with the amount of data to rebuild — the Recovery Point Objective (RPO). The RPO is affected by how often you can go-to-sync with PPRC-XD.

In an Asynchronous Cascading PPRC configuration, one method for obtaining a consistent point-in-time copy of the data at the remote site is by freezing the local PPRC pairs. The main advantage of this approach is that it is not necessary to stop or quiesce the applications in order to get the consistent point-in-time copy. This procedure is illustrated in Figure 3-23.

**Note:** When using Fibre Channel links between site A and site B, if the Synchronous PPRC is active with application I/O still running at site A and a go-to-SYNC is triggered for the XD pair, then the XD pair will automatically suspend.

We recommend that when go-to-SYNC is issued for the XD pair, the Synchronous PPRC pairs should be suspended. This is normally done via a freeze operation as described in our example.

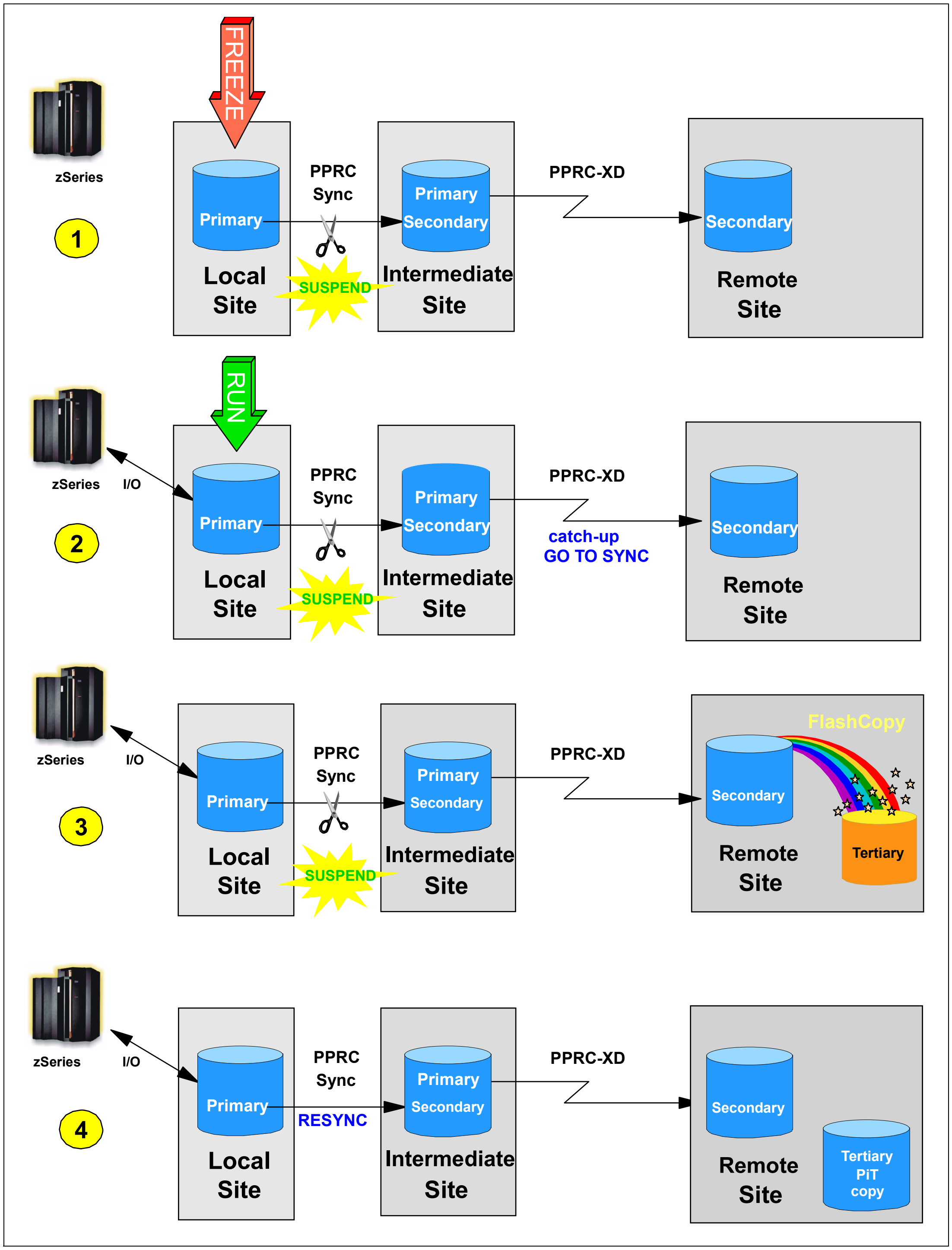



*Figure 3-23 Getting a consistent point-in-time copy at remote site without stopping application*

The following sequence is a description of the procedure illustrated in Figure 3-23:

1. As soon as the primary LSS is frozen, the paths for that LSS are removed and the PPRC volume pairs in the LSS are suspended.

   If the paths for the LSS were defined with Consistency Group enabled (recommended if application volumes are on two or more LSSs), then an extended long busy (ELB) condition is triggered. During the ELB window, update activity does not occur in the primary LSS volumes. With this mechanism, dependent writes do not proceed on the primary volumes — thus ensuring that the volumes at the intermediate site are consistent to the point-in-time when the freeze was done.

2. Applications are not stopped at any time, because it is the ESS that briefly enqueues the I/O activity for the creation of the point-in-time. In order to reset the ELB condition and resume the updates to the primary volumes, an ICKDSF RUN command or an ESS Copy Services task with the **Consistency Created** option can be run. At this point, the secondary volumes of the local PPRC pair remain suspended, not receiving updates, and holding the point-in-time copy at the time the freeze was requested.

   After a short time PPRC-XD will do an automatic catch-up of the remote site volumes. Alternatively, a catch up can also be done by commanding PPRC-XD to do so. When triggering a catch-up by commanding PPRC to go-to-SYNC, the `IEA494` system messages will allow you to detect the moment when the volume pairs reach the duplex state — that is, when the catch-up is completed.

3. After a while, when the volumes at the remote site are 100% in synchronization with the intermediate site volumes, a FlashCopy to tertiary volumes can be done. If FlashCopy Version 2 is enabled in the remote ESS, then Incremental FlashCopy can be done (FlashCopy Version 2 is discussed in 2.2, “FlashCopy Version 2” on page 31).

4. Once the FlashCopy relationships are established at the remote site, the procedure for re-synchronizing the suspended local PPRC pairs can be initiated. This means redefining the paths that were deleted by the freeze operation, and re-synchronizing the local site to intermediate site pairs. This will resume the updates on the intermediate site volumes, which then will be propagated by PPRC-XD to the remote site volumes.

This procedure can be repeated at regular intervals in order to hold recent consistent point-in-time copies of the data at the remote site, without having to stop the applications at any moment.

## 3.9 Asynchronous PPRC

More and more customers are separating data centers by larger distances to provide protection against regional disasters such as terrorist attacks, earthquakes and hurricanes, or they want to use their existing data centers at long distances for disaster recovery purposes. Synchronous copy solutions such as PPRC over Fibre Channel using DWDMs have a limitation of 300 km and as the distance increases the associated response time delay may be unacceptable for some customer’s environments. Extended Remote Copy (XRC) is available only for z/OS customers.

Asynchronous PPRC is designed to provide a long-distance remote copy solution across two sites for open systems and z/OS data using asynchronous technology. It is based on existing Copy Services functions PPRC-XD and FlashCopy and additional new functionality for the periodic creation of Consistency Groups across one or more ESSs and for the management and control of the Asynchronous PPRC session.

It operates over high-speed Fibre Channel communication links and supports virtually an unlimited distance between the local and remote sites. The distance is typically limited only by the capabilities of the network and channel extension technologies. This provides a wider choice of remote site locations based on business needs. It also enables site separation to add protection from localized disasters.

The data at the remote site is maintained to be a consistent and application restartable copy of the data at the local site without significant impact to applications at the local site.

Compared to PPRC-XD or Asynchronous Cascading PPRC, Asynchronous PPRC provides automation to periodically create Consistency Groups at the primary and the secondary site.

Data currency at the remote site lags behind the local site typically 3 to 5 seconds (when the Consistency Group Interval Timer is set to 0 seconds under a typical workload), minimizing the amount of data exposure in the event of an unplanned outage. Data currency is also known as the recovery point objective (RPO). The actual lag in data currency experienced will depend upon a number of factors, including specific workload characteristics and available bandwidth between the local and remote sites.

Data consistency at the remote site is internally managed across up to four (current limit) primary ESS machines located at the local site.

Given a known primary workload and the available bandwidth to the remote site, a timer variable in the Asynchronous PPRC session can dynamically be adjusted to meet the Recovery Point Objective of the customer.

## 3.9.1 Terminology

With Asynchronous PPRC, the data consistency at the secondary site is guaranteed in comparison to PPRC-XD where a disruption is required to create consistency and take remote copies for each point-in-time.

The following terms are used in further explanations in the following sections.

### *Dependent writes*

If the start of one write operation is dependent upon the completion of a previous write, the writes are dependent. Application examples for dependent writes are databases with associated log files, catalogs, and VTOCs as well as VSAM indexes and VSAM data components. For instance, the Data Base Log will be updated after a new entry has been successfully written to a Table Space, or the VTOC will be updated if you have created a new data set.

The chronological order of dependent writes to primary volumes is the basis for providing consistent data for the copy operations.

### *Consistency*

The consistency of data is ensured if the order of dependent writes to disks or disk groups is maintained. With copy services solutions the data consistency at the secondary site is important for the usability of the data at the remote site. Consistent data for instance provides the ability to perform a data base restart rather than a data base recovery that could be hours or even days.

Data consistency across all secondary volumes spread across any number of LSSs is essential for logical data integrity.

### *Data currency*

This term describes the difference of time since the last data was written at the primary site versus the time the same data was written to the secondary site and determines the amount of data you have to recover at the remote site after a disaster. Only synchronous copy solutions like Synchronous PPRC have a currency of zero. (All asynchronous copy solutions have a data currency greater than zero.) With the new Asynchronous PPRC solution a data currency of a few seconds can be achieved, while data consistency is maintained by the Asynchronous PPRC process.

For Copy Services this means, that the data is not copied at the same time as the local I/O. The data is written with a time difference to the remote site. Examples of asynchronous Copy Services are:

- PPRC-XD with non-consistent data at the remote site
- Asynchronous Cascading PPRC
- XRC with consistent data at the remote site
- Asynchronous PPRC with consistent data at the remote site

#### *Session*

An Asynchronous PPRC Session is a collection of volumes that are managed together when creating consistent copies of data. This set of volumes can reside in one or more LSSs and one or more ESSs at the customer's primary site. Open systems volumes and z/OS volumes can both be members of the same session.

When you start or resume a session, the creation of Consistency Groups is performed and the Master ESS controls the session by communicating with the Subordinate ESSs.

#### *Master*

The Master is a term for a function inside an ESS that, by communicating with Subordinates in other ESSs, controls the creation of Consistency Groups and manages the Asynchronous PPRC Session. The Master is defined in the start command for a session by addressing any LSS in the Master ESS.

The Master needs PPRC Fibre Channel communication paths to any one of the LSSs in each Subordinate ESS.

#### *Subordinate*

The Subordinate is a term for a function inside an ESS that communicates with the Master and is controlled by the Master. At least one of the LSSs of each Subordinate ESS needs PPRC Fibre Channel communication paths from the Master to the Subordinate. It is used to enable the Master to create Consistency Groups of volumes in more than one ESS.

If all the volumes of an Asynchronous PPRC Session reside in one primary ESS no Subordinate is required, because the Master can communicate to all LSSs inside the primary ESS.

#### *Consistency Group*

This is a group of volumes in one or more ESSs whose data must be kept consistent at the remote site.

> **Note:** With Asynchronous PPRC this is done periodically and automatically without disruption to the host I/O in a very short time.

### 3.9.2 Asynchronous PPRC - How it works

The purpose of Asynchronous PPRC is to provide recoverable copies of production data at a long distance remote site by continually forming consistent sets of data with no significant performance impact. This allows a quick restart at the remote site in case of a disaster at the primary site.

For better understanding, let's discuss a possible solution using only PPRC-XD and FlashCopy as described in Figure 3-24, before we describe how Asynchronous PPRC works. Without Asynchronous PPRC, the following procedure can be used to periodically get consistent data at a long distance remote site.

Periodically you can switch from PPRC-XD to synchronous mode, then either stop the application I/O or freeze the primary volumes, which suspends the PPRC pairs, then release the application I/O to the primary site.

Next, take a FlashCopy at the remote site to obtain the consistent data and then reestablish the PPRC-XD relationship with the RESYNCH parameter. This can be done manually or with an automated process.

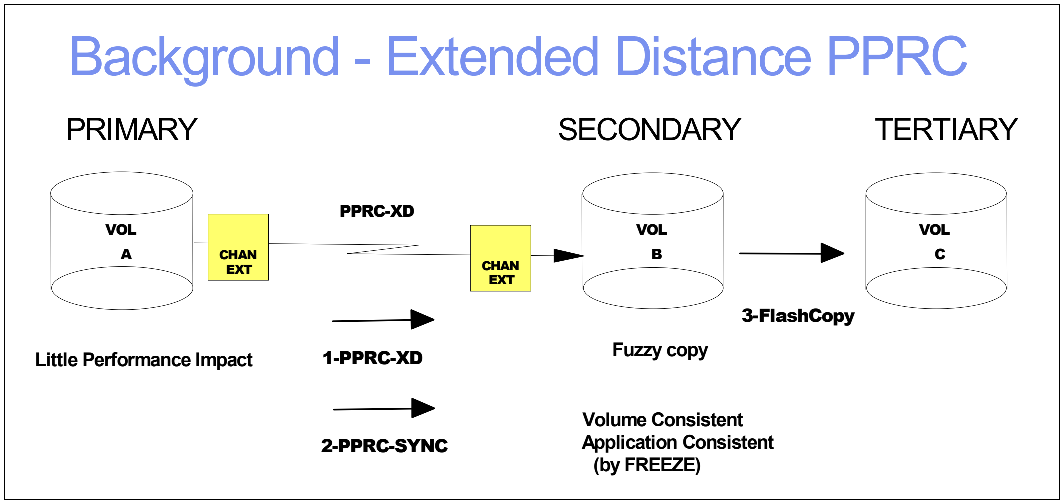



*Figure 3-24 PPRC Extended Distance background*

This is a solution, but it has a lot of disadvantages:

- Unacceptable currency of the remote site data for time critical applications
- Application downtime or performance impact during the time to go-to-sync and the Freeze-Run process, when I/Os are stopped
- Several separate manual steps to manage

## Basics of Asynchronous PPRC

In contrast to the solution above, Asynchronous PPRC overcomes these disadvantages and automates this type of solution so that you can get these benefits:

- A very good data currency of only a few seconds
- Automatic creation of Consistency Groups in a very short time
- No application impact at the local site
- One central point of control

The new process of Asynchronous PPRC is implemented at Licensed Internal Code level 2.4.x or higher of the ESSs and works as described in Figure 3-25 for all volumes in one or more LSSs that previously have been added to a defined Session. The creation of the Consistency Group at the local site is a very quick process followed by a drain of the consistent data to the remote site, while application I/Os are continuously running.

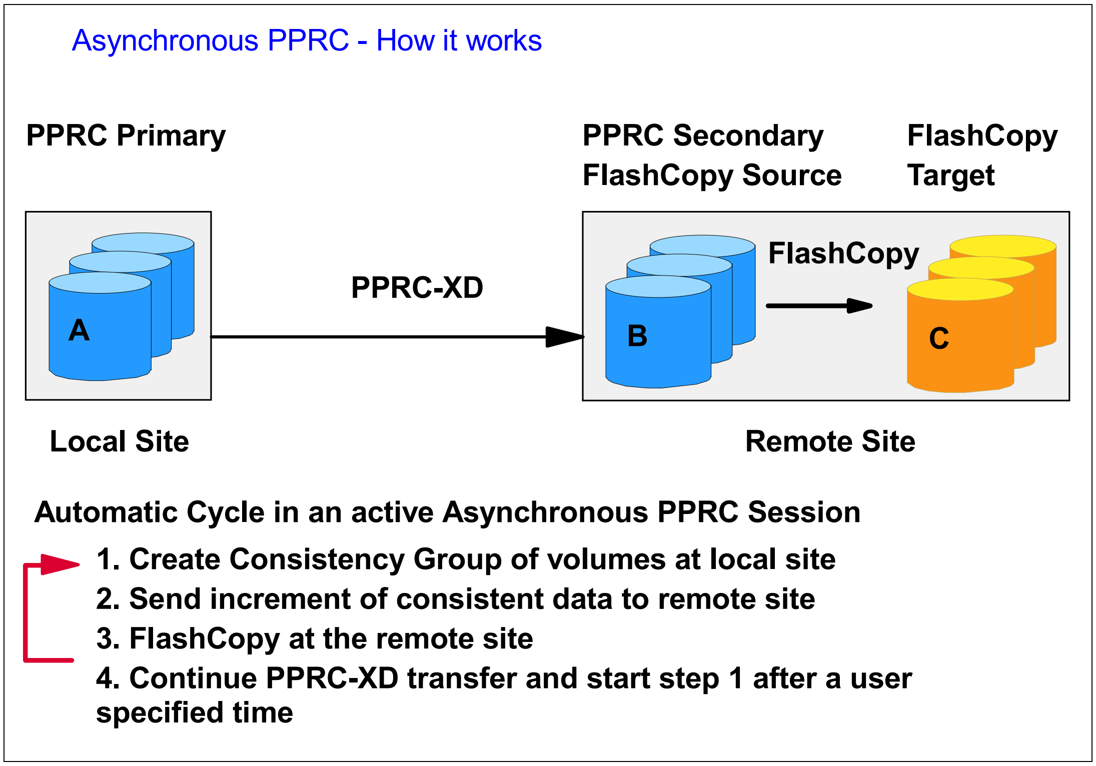



*Figure 3-25 Asynchronous PPRC*

A persistent FlashCopy with no background copy and active change recording continues immediately after the drain has finished. According to your Recovery Time Objectives (RTO), the available bandwidth and your application workload, you can assign a Consistency Group Interval Timer value before the process starts again at step one.

The currency of the data at the remote site can achieve typically 3 to 5 seconds (with a zero second Consistency Group Interval Timer value), but this depends on the workload and bandwidth available to the remote site.

Notice that the consistent data at the remote site resides on the FlashCopy target volume and not on the PPRC secondary volume.

To enable this very fast cycle across multiple ESSs, a fast inband communication network is required. The following configuration example, Figure 3-26, explains the necessary communication across LSSs and ESSs.

One of the local site ESSs is the Master ESS that communicates to Subordinate ESSs over Fiber Channel links to enable the creation and control of the Consistency Groups. The Fibre Channel communication paths have to be defined from one or more LSSs inside the Master ESS to at least one LSS in each Subordinate ESS. The paths can be established with the same commands as PPRC data paths. They can be defined on separate physical links or on links combined with PPRC data paths usually in a SAN. For redundancy purposes at least two paths from the Master to each Subordinate should be defined.

The volumes of the Master ESS do not have to be members of an Asynchronous PPRC session, but they may be.

The volumes in the session can be open systems volumes as well as z/OS volumes. Consistency Groups are created for all volumes at the same time.

.

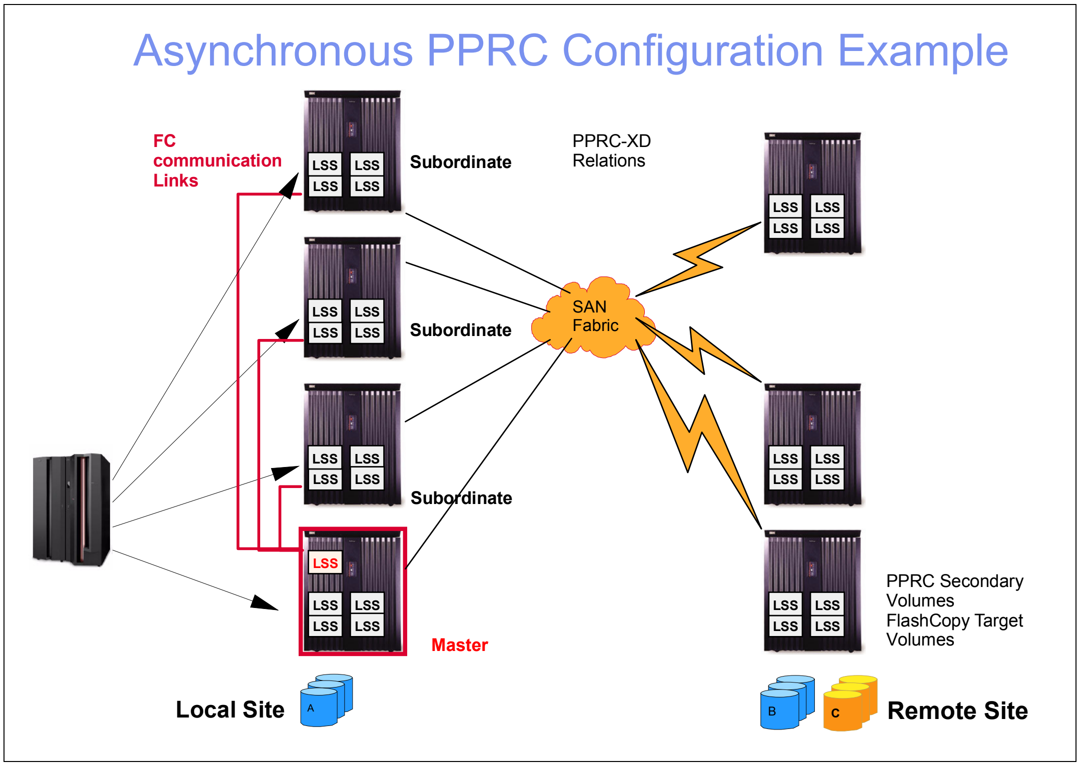



*Figure 3-26 Example of an Asynchronous PPRC configuration*

## Asynchronous PPRC Session management

A session is a collection of volumes in an ESS or in multiple ESSs that are managed together when creating consistent copies of data. When you start or resume a session, the Asynchronous PPRC configuration is specified or modified. This causes Asynchronous PPRC to form a Consistency Group that includes each volume in the session. The topology of a session is defined by its Master and Subordinate relationship.

At the time of General Availability of LIC V2.4.0 support, only the Copy Services Web User Interface, the Command Line Interface and ICKDSF can be used to manage Asynchronous PPRC; see Figure 3-27. It is planned to enable the Multiple Device Manager, TSO, ANTQRST Macro and GDPS for managing Asynchronous PPRC. In this redbook we focus on the management with ICKDSF, but we also explain the capabilities of the Web Copy Services, that can be easily be used for monitoring purposes.

Open systems and z/OS volumes are managed in the same session and the creation of the Consistency Groups is one process for both platforms. With ICKDSF, only the z/OS portion of the volumes can be addressed, but the entire session can be controlled. With the Web User Interface you can manage both types of volumes.

It is also possible to establish an Asynchronous PPRC relationship in both directions with primary and secondary volumes at the local and remote sites.

### *Restrictions*

The following restrictions apply:

- Currently only one Master can reside in an ESS and therefore only one Session can be managed by one ESS. (Architecture allows for growth.)
- An LSS can be assigned only to one Session. (Architecture allows for growth.)

- Maximum of three subordinates with LIC 2.4.0 using Web Copy Services.
- An individual Session consists some or all of the volumes in a Master and up to three primary ESS subordinates. (Architecture allows for growth.)
- Maximum of eight ESSs (in any combination of primaries and secondaries) can participate in an Asynchronous PPRC environment using ICKDSF or MDM. (Architecture allows for growth.)

The Asynchronous PPRC mechanism will only act upon volumes which have been added to a Session, other PPRC volumes may still operate in any Copy Services mode and not be included in the formation of Consistency Groups.

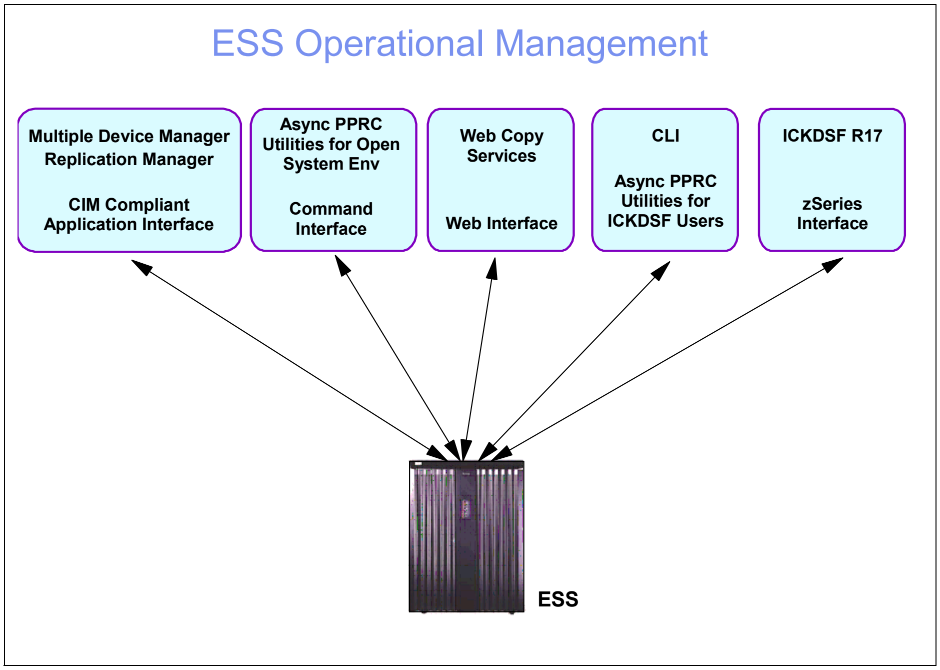



*Figure 3-27 Management of Asynchronous PPRC*

## 3.9.3 Steps to establish an Asynchronous PPRC Session

First of all, identify all PPRC primary and secondary volumes and the corresponding LSSs as well as the FlashCopy target volumes at the remote site for the Asynchronous PPRC Session. Determine the Master ESS and LSS and the session ID. See Figure 3-28 for the initialization steps:

1. Establish data paths from the primary ESSs to PPRC secondary ESSs.
2. Establish the required PPRC volume pairs with mode XD.
3. Establish Inband FlashCopies (Inband is not required, but recommended) at the remote site with the options: *nocopy*, *change recording*, *inhibit target writes*, and *persistent,* which are addressed to the PPRC primary volumes.
4. Define the logical PPRC paths from the planned LSS in the Master ESS to any one LSS inside of each Subordinate ESS.
5. Define a session with an associated ID to any LSS that has volumes to be managed by Asynchronous PPRC.
6. Add all the primary PPRC volumes needed to the session.
7. Start the session with the session ID addressing the planned Master LSS.

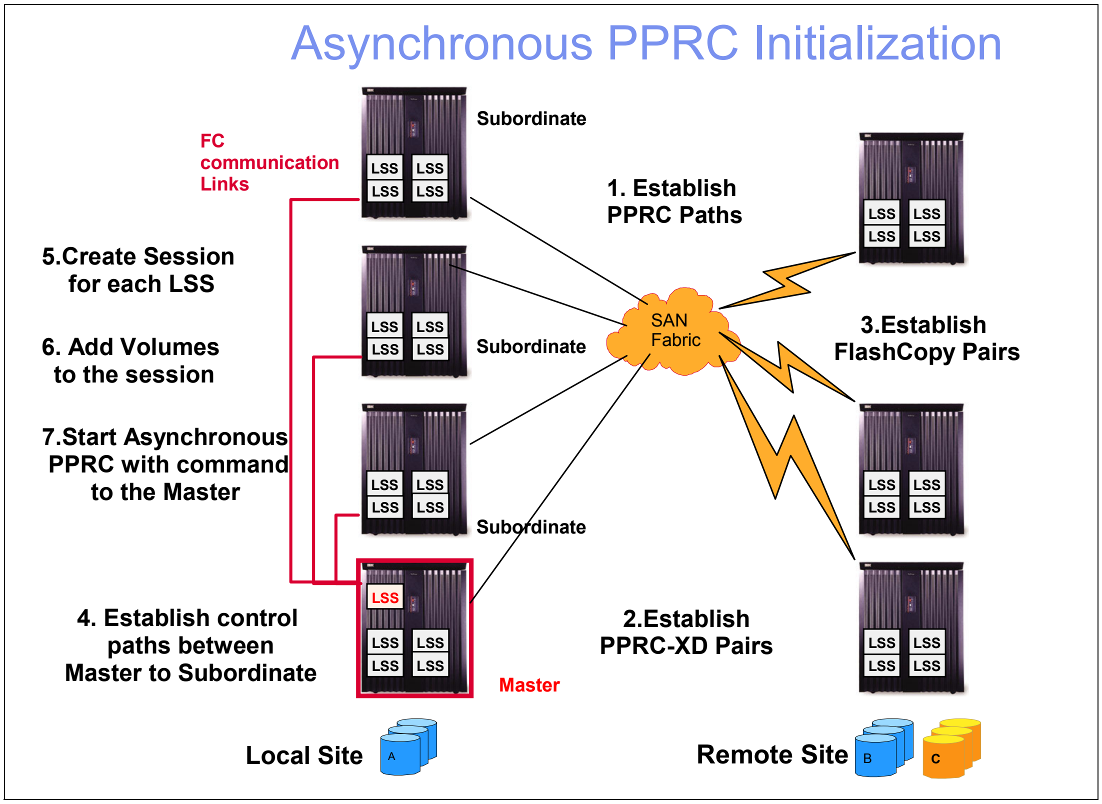



*Figure 3-28 Initialization of Asynchronous PPRC*

Once all of the volumes have joined the session, you can monitor the creation of Consistency Groups by watching the Web User Interface or submitting ICKDSF Queries to the Master or volumes of the session.

### *Session management commands*

The description of the ICKDSF commands can be found in 4.3, “Using ICKDSF to control Copy Services” on page 156 and in detail in the *Device Support Facility Users Guide and Reference,* GC35-0033.

### *Open/close a session*

A session must be opened using a specific session ID and a session must be opened using the same session ID for each Logical Subsystem that participates in the Asynchronous PPRC session. This step must be done prior to adding volumes to a session. Before closing a session you need to remove the volumes. Closing a session in an LSS deletes all of these definitions in that LSS. Close a session when you no longer want the volumes in a particular LSS to participate in Asynchronous PPRC.

### *Add/remove volumes*

To add volumes to the session you have to specify all PPRC primary volumes. The volumes are set to the *join pending state* until the first PPRC synchronization pass has occurred. They can be dynamically removed and added to a running session.

### *Start/resume a session*

The session will be started by addressing any LSS in the ESS that you wish to become the Master. All subsequent Asynchronous PPRC commands must be addressed to this same LSS. This LSS must have PPRC communication to all Subordinate ESSs. A `Start` command starts the Asynchronous PPRC mechanism for the first time or resumes the Asynchronous PPRC mechanism after a `Pause` command has been issued.

> **Note:** Once the initial Asynchronous PPRC configuration is created and for as long as it exists, you can start or resume a session for the Master only on the same LSS where the initial session was created.

You can modify three timer values:

- **Consistency Group Interval Timer:** (in seconds, default 0, maximum 64K sec), determines the time from last successful FlashCopy to the start of the creation of the next Consistency Group.
- **Maximum Coordination Timer:** (in milliseconds, default 50ms, maximum 64K ms), determines the maximum time to coordinate the ESSs at the local site in order to form a Consistency Group.
- **Maximum Consistency Group Drain Timer:** (in seconds, default 30 sec, maximum 64K sec), determines the maximum time to transfer the consistent data to the remote site

### *Pause/terminate a session*

This command can be used to terminate or pause the Asynchronous PPRC Session. Both commands stop the Asynchronous PPRC Session. `Pause` will ensure the current Consistency Group forms before stopping the Asynchronous PPRC Session. `Terminate` will interrupt the formation of a Consistency Group if one is running and clean up by reverting to the previous Consistency Group. If no group is being formed it will terminate immediately. Use Terminate when you need to change the session topology such as defining a new Master and adding or removing a Subordinate. The command removes Master/Subordinate relationship information. Force Terminate will terminate immediately and will interrupt the creation of the Consistency Group without recovery if one is being processed.

> **Note:** Force should only be done when instructed to do so by IBM support. It requires that the Consistency Group be manually verified before the data can be used.

### *Query a session or volumes*

There are several different queries available for the Web User Interface and ICKDSF. The ICKDSF queries are described in more detail in 4.3, “Using ICKDSF to control Copy Services” on page 156.

There are three PPRC queries that provide:

- Data about the Asynchronous PPRC configuration, the session status and statistics about creating Consistency Groups.
- Data about the number and status of volumes in the specified session that are out-of-sync.
- Status of the session open for a specific LSS.

Three FlashCopy queries provide the following information:

- Status of the volumes and the FlashCopy relationship.
- Status of the Consistency Groups, FlashCopy information and out-of-sync tracks.
- Revertible status of the FlashCopy.

There are different command names for ICKDSF and for the Web User Interface that are summarized in Table 3-5.

*Table 3-5 Summary of command names*

| **Web Interface** | | **ICKDSF** | |
|---|---|---|---|
| Manage Session | Open session | DEFINESESSION | Open |
| | Close session | | Close |
| Manage session members | Add | POPULATESESSION | Join |
| | Remove | | Remove |
| Start/Resume session | Start | STARTSESSION | Start |
| | Resume | | Modify |
| Pause/Terminate session | Pause | TERMINATESESSION | Pause |
| | Terminate | | Terminate |

### *Prerequisites for Asynchronous PPRC*

These are the prerequisites:

- ESS model 800 or 750 for the primary and secondary subsystems.
- PPRC V2 for primary and secondary ESSs.
- FCP PPRC links.
- ESS LIC 2.4.0 minimum.
- Existing ESS machines do not require the new feature code.
- FlashCopy V2 required at the remote site.
- Any site with PPRC V2 that uses FlashCopy requires FlashCopy V2.

## 3.10 PPRC links

PPRC pairs are set up between volumes in LSSs, usually in different ESSs, and these are normally in separate locations. To establish a PPRC pair, there needs to be a PPRC path between the LSSs that the volumes reside in. These paths are uni-directional and can be shared by any PPRC pairs in the same LSS to the secondary LSS in the same direction. For bandwidth and redundancy, more than one path can be created between the same LSSs, PPRC will balance the workload across the available paths between the primary and secondary.

PPRC pairs can only be established between storage control units of the same type and features, which means that the primary and secondary must both be ESSs with the applicable optional PPRC functions enabled (see Appendix D, “ESS Copy Services feature codes” on page 383).

While it is allowable to have paths both to and also from an LSS, individuals paths can only flow in one direction. So if there are both primary and secondary volumes in an LSS, then paths must be established in both directions. A path (or group of paths) needs to be established *from* the LSS to each LSS with related secondaries. Also, a path (or group of paths) must be established *to* the LSS from each LSS with related primaries.

These logical paths are transported over physical links between the ESSs. The physical link includes the host adapter in the primary ESS, the cabling, switches or directors, any wideband or long distance transport devices (DWDM, channel extenders, WAN) and the host adapters in the secondary ESS. Physical links can carry multiple logical PPRC paths.

Two methods are supported:

- **ESCON:** This requires ESCON host adapters in both ESSs. The link can be directly cabled or routed through ESCON directors in the normal way. Because ESCON is a direction oriented protocol, ESCON links can only operate in one direction. Separate physical links must be installed if PPRC paths need to be connected between ESSs using ESCON in the other direction.

  PPRC over ESCON supports distances of up to 103 km for synchronous operation. This method is enabled with the PPRC V1 or PPRC V2 features.

- **PPRC over FCP:** This requires a Fiber Channel host adapter in both ESSs. These can be 1Gb or 2Gb. The path can then be transported over a SAN network using switches and directors in accordance with normal SAN protocols. As Fibre Channel protocol is bi-directional then the physical links can carry logical PPRC paths in both directions, and through a SAN fabric, to multiple ESSs.

  PPRC over FCP supports distances of up to 300 km for synchronous operation. This method is enabled with the PPRC V2 feature.

> **Important:** It is highly recommended that the *Interoperability Matrix* is consulted when planning any physical links. This can be found by visiting the ESS homepage:
>
> `http://www-1.ibm.com/servers/storage/support/disk/2105.html`

## 3.10.1 PPRC connectivity using ESCON ports

This section discusses connectivity considerations useful for planning the PPRC solution when using ESCON ports.

> **Note:** Connectivity considerations when using Fiber Channel Ports is discussed in the following section 3.10.3, “PPRC connectivity using Fibre Channel ports” on page 134.

ESCON is an IBM architecture and fiber optic technology that is used extensively and very successfully in @server zSeries systems to connect from the host to I/O controllers and subsystems. It is also used in ESS for PPRC connections. This connection can go over distances beyond 100 km with suitable repeaters. With these increased distances, disaster recovery methods can be established based on the remote copy functions.

ESCON links between ESS subsystems are required — these are the PPRC links. The local ESS is usually called primary if it contains at least one PPRC source volume, while the remote ESS is called secondary if it contains at least one PPRC target volume. An ESS can act as primary and secondary at the same time if it has PPRC source and target volumes.

Each ESCON adapter card in the ESS supports two ESCON ports or links. An ESCON link consists of a fiber pair connection —there is a transmit fiber and a receive fiber. The ESCON protocol is unidirectional. This means that a physical ESCON link can be used to transmit data from the primary ESS to the secondary ESS. If you want to set up a PPRC configuration with source and target volumes on each ESS, you need ESCON PPRC links in both directions (see Figure 3-29). The number of links depends on the bandwidth needed as a result of the write activity to the primary volumes and on the number of ESSs connected together.

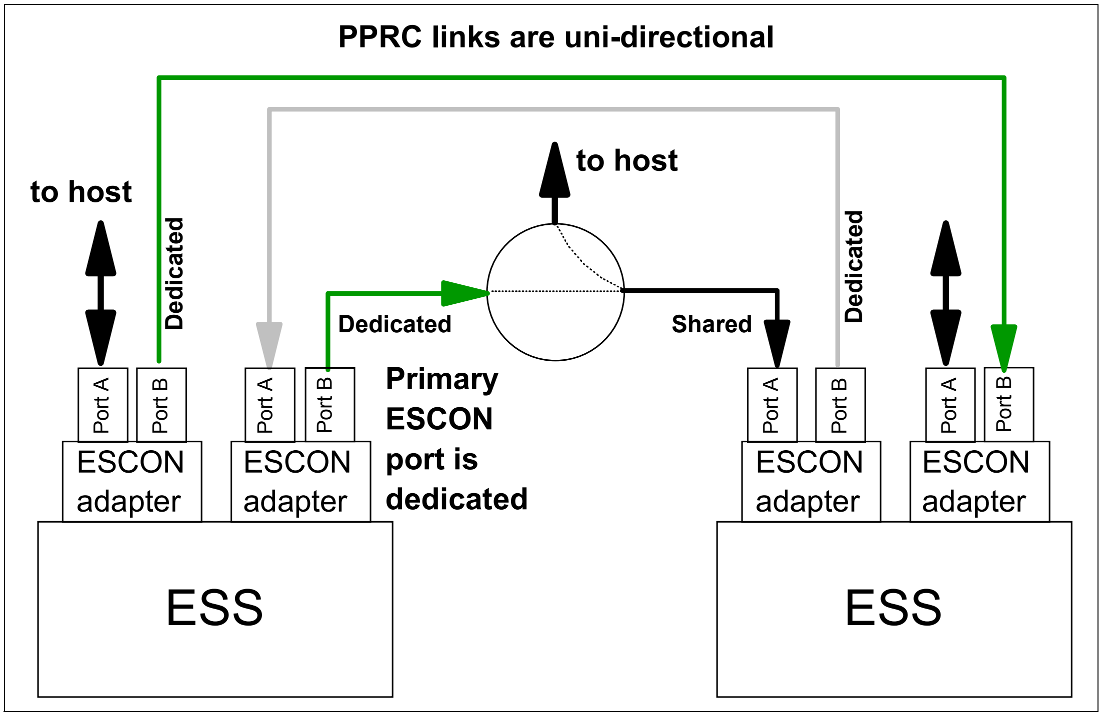



*Figure 3-29 ESCON PPRC links*

### *Operation modes*

Primary PPRC ESCON ports are dedicated for PPRC use. An ESCON port is operating in *channel mode* when it is used on the primary ESS for PPRC I/O to the secondary ESS.

A port operates in *control unit mode* when it is talking to a host. In this mode, a secondary ESCON port can also receive data from the primary control unit when the ESS port is connected to an ESCON director. So, the ESCON port on the secondary ESS does not need to be dedicated for PPRC use. The switching between *control unit mode* and *channel mode* is automatic.

If there is any logical path defined over an ESCON port to a host, you cannot switch this port to channel mode for PPRC to use as primary port. You must first configure offline all logical paths from the host to that port. Now you can define a PPRC logical path over this ESCON port from the primary to the secondary ESS. When you establish the logical path, the port will automatically switch to channel mode (see Figure 3-29).

## PPRC logical paths

Before PPRC pairs can be established, logical paths must be defined between the primary and secondary ESS logical subsystems (LSSs). The logical paths between LSSs are established over physical ESCON links (see Figure 3-30). The following guidelines apply for the configuration of PPRC logical paths:

- The ESS supports up to 16 CKD logical subsystems.
- An ESCON port supports up to 64 logical paths.
- A pair of LSSs can be connected with up to eight logical paths in one direction.

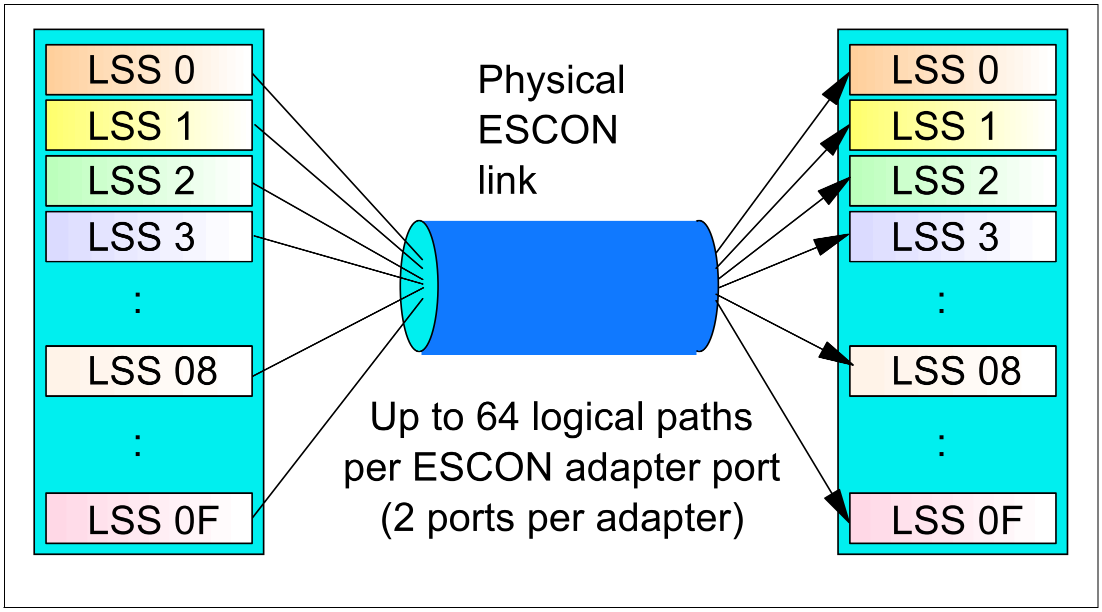



*Figure 3-30 Logical paths*

The PPRC logical paths are established between the LSSs (see Figure 3-31). A primary LSS can be connected to up to four secondary LSSs from the same or different ESSs. A secondary LSS can be connected to as many primary LSSs as ESCON links are available.

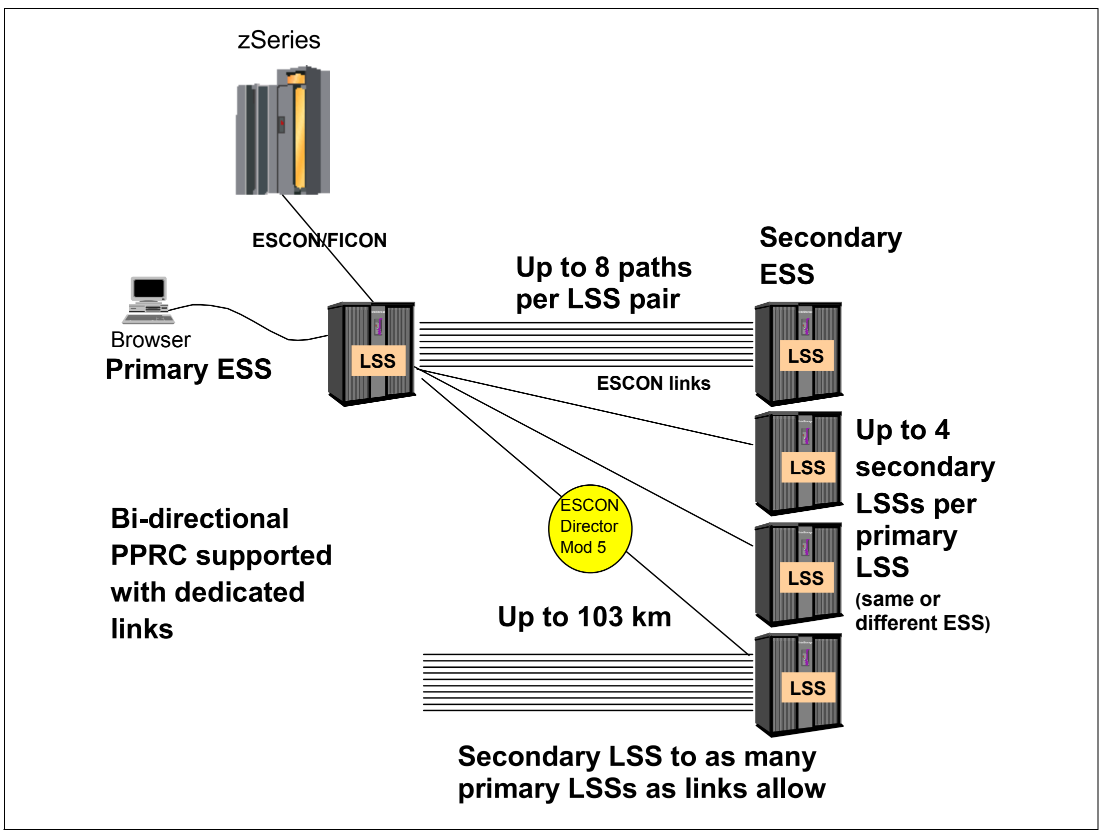



*Figure 3-31 PPRC configuration options*

## ESCON links

Although ESCON provides capabilities to connect over long distances, the following factors affect the distance that can be achieved:

- Physical size of fiber
- Fiber transmission mode
- Connection mode

### *Physical size of fiber*

An optical fiber functions as a wave guide for light. It is usually made of silica glass. The fiber itself has a central core and a surrounding cladding of slightly different glass material. The physical size of an optical fiber is determined by the diameter of the core and cladding, expressed in microns (µ). A fiber optic cable having a core diameter of 62.5 µ and a cladding of 125 µ is designated as 62.5/125 µ optical fiber. Other fibers commonly used for ESCON are 50/125 µ and 9/125 µ.

ESCON jumper cables are typically used to connect:

- A channel to a control unit
- A channel to an ESCON Director
- An ESCON Director to an ESCON Director
- An ESCON Director to a control unit
- An ESS to an ESS (PPRC)

An ESCON jumper cable consists of two unidirectional fibers: one that carries data in one direction (transmit), and the other that carries data in the opposite direction (receive). The unidirectional characteristic is due to the way the fibers are connected to transmitters and receivers, not to any property of the fiber itself.

### *Fiber transmission mode*

Two modes can be used to send light signals through an optical fiber: *single mode* (mono mode) or *multimode*. The optical fibers used are called accordingly single-mode or multimode fibers. These fibers have different physical dimensions and light transmission characteristics.

The intensity of the light decreases when the light passes through the fiber, so the longer the fiber, the lower the light intensity. The light intensity attenuation results in length limitations of the physical link because the receiver needs a minimum level of light intensity to correctly detect the signals.

The multimode fibers supported by IBM are either 62.5/125 µ or 50/125 µ. The light source used for multimode fiber is usually a light emitting diode (LED). The LED jumpers are orange, and their duplex connectors are black. The maximum distance for a multimode fiber link is 3 km if 62.5/125 µ fibers are used, and 2 km if 50/125 µ fibers are used. The maximum distance for a multimode fiber link is 2 km if both 62.5/125 µ and 50/125 µ fibers are used.

Single-mode fiber usually has a core diameter of 8 to 10 µ and a cladding diameter of 125 µ. The light source used for single-mode fiber is a laser. IBM supports single-mode 9/125 µ fibers for use in an ESCON environment. Single-mode fiber must be used on links between machines with ESCON-Extended Distance Feature (XDF) adapters installed. These adapters are available only on ESCON Directors and on remote channel extenders.

Fiber jumper cables of 9/125 µ for IBM ESCON Directors are supplied in standard lengths up to 122 m (400 ft.), if XDF ports are used. XDF jumpers are yellow, and their duplex connectors are grey. The maximum distance for a single-mode fiber link is 20 km.

### *ESCON connection modes*

An I/O control unit can be connected to a host channel in the following ways:

- **A point-to-point connection:**

  A point-to-point connection is a fiber connection between a host channel and an I/O control unit. This is called a physical link. Through this physical link a host can communicate with only the devices attached to that I/O control unit.

- **An ESCON Director as a static connection:**

  A static connection through an ESCON Director involves having one fiber connected from a host channel to an ESCON Director, and a second fiber from an ESCON Director to an I/O control unit.

  A static connection through an ESCON Director is more flexible than a point-to-point connection. A different connection can be achieved by statically switching to another port, without reconnecting the fiber.

- **An ESCON Director as a dynamic connection:**

  A dynamic connection through an ESCON Director involves having the same fiber arrangement as for a static connection. However, instead of setting up an ESCON Director for a static connection, the connections to ESCON Director destination ports are coded in the link statement of the control unit macro in the IOCDS or equivalent HCD panel. In this mode, a host can communicate with multiple devices behind different I/O control units by using the same physical link between the host channel and the ESCON Director. The ESCON Director dynamically connects to the destination port, as required.

## Guidelines and considerations

These are the options and considerations to consider when planning the PPRC connectivity over an ESCON link:

- The possible PPRC ESCON connectivity alternatives are:
  - Direct (point-to-point) ESCON links
  - Via ESCON Directors
  - Channel Extenders over wide area network (WAN) lines
  - Dense Wave® Division Multiplexors (DWDM) over dark fibers
- Synchronous PPRC (SYNC) maximum supported distance over ESCON is 103 km. For longer distances an RPQ must be submitted.
- The connectivity infrastructure is transparent to PPRC.
- Evaluation, qualification, approval, and support of PPRC configurations using channel extender products is the sole responsibility of the channel extender vendor.
- The channel extender vendors and DWDM vendors should be consulted regarding prerequisites when using their products.

> **Note:** A current list of PPRC supported connectivity products can be found in the interoperability matrix:
>
> `http://www.storage.ibm.com/hardsoft/products/ess/supserver.htm`

When connecting beyond the native ESCON link range of 3 km, the following components come into play:

- **ESCON Director:**

  An ESCON Director offers the possibility to extend the ESCON distances to up to 20 km on a physical link, by using an ESCON-XDF port. These ports are available as a feature on the ESCON Directors.

  In addition to its switching function within an ESCON network, an ESCON Director acts as an amplifier or repeater — the light signals are received on one port and transmitted by another — thus allowing distances of up to 2 km, 3 km, or 20 km to be reached.

- **Channel extender:**

  Channel extender vendors connect ESS PPRC systems through a variety of wide area network (WAN) connections. PPRC can be used over all the network technologies that are currently supported by the channel extender vendors, including Fibre Channel, Ethernet/IP, ATM-OC3, and T1/T3.

  When using channel extender products with PPRC, the channel extender vendor will determine the maximum distance supported between the primary and secondary ESS. The channel extender vendor should also be contacted for their line quality requirements, and WAN attachment capabilities. Evaluation, qualification, approval and support of PPRC configurations using channel extender products should be expected from the channel extender vendor.

- **Dense Wave Division Multiplexor (DWDM):**

  Dense Wave Division Multiplexing (DWDM) is the basic technology of fibre optical networking. It is a technique for carrying many separate and independent optical channels on a single dark fibre.

  A simple way to envision DWDM is to consider that at the primary end, multiple fibre optic input channels — such as ESCON, Fibre Channel, FICON, or Gbit Ethernet — are combined by the DWDM into a single fibre optic cable. Each channel is encoded as light of a different wavelength. You might think of each individual channel as an individual color: the DWDM system is transmitting a *rainbow*. At the receiving end, the DWDM fans out the different optical channels. Receivers at the remote end then re-drive the signal in its original medium.

  The DWDM vendor should be contacted regarding hardware and software prerequisites when using their products in an ESS PPRC configuration.

## 3.10.2 Distance connections

This section illustrates the most common distance configurations that can be used for PPRC connectivity.

> **Note:** The discussion in this section is limited to configurations using ESCON ports; for connectivity considerations when using Fiber Channel Ports refer to “PPRC connectivity using Fibre Channel ports” on page 134.

As discussed in the previous section, the intensity of the light decreases when the light passes through the fiber link, so the longer the fiber, the lower the light intensity. This light intensity attenuation results in distance limitations of the physical link, because the receiver needs a minimum level of light intensity to correctly detect the signals. Also, signals need time to travel, and in the Synchronous PPRC implementation, I/O operations must wait for the secondary acknowledgement before they can complete, so there will be distance limitations resulting from the acceptable I/O time overheads that the applications can tolerate.

The maximum supported distance for Synchronous PPRC is 103 km (slightly longer distances may be possible, but an RPQ must be submitted). This distance can be achieved using DWDM (Dense Wave Division Multiplexers) connecting through dark fibers, or using channel extenders over WAN lines. Also, switches or directors can be used for shorter distances.

## ESCON configuration up to 3 km

Figure 3-32 shows a direct point-to-point connection between a primary and a secondary ESS. Because standard ESCON adapters are used in the ESS, this connection can only be a multimode connection. Depending on the fibers you use, the maximum distance can be 2 km (if 50/125 μ fibers are used) or 3 km (if 62/125 μ fibers are used).

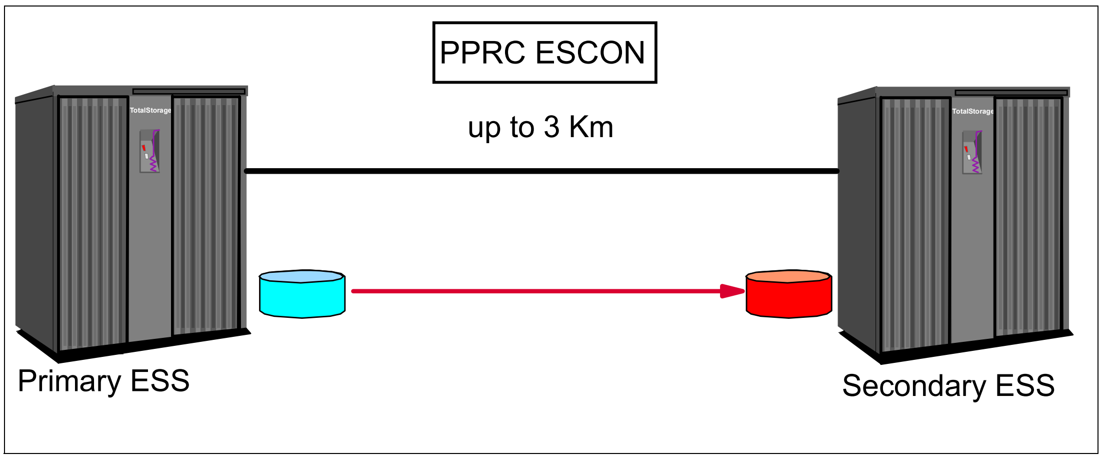



*Figure 3-32 Point-to-Point PPRC configuration - up to 3 km*

## ESCON configuration up to 6 km

Figure 3-33 illustrates a connection using an ESCON Director. In terms of distance, there is no difference between a static and a dynamic connection through an ESCON Director. The maximum distance between each ESS and the ESCON Director can be 2 km (50/125 μ fiber) or 3 km (62.5/125 μ fiber), resulting in a maximum distance between both ESSs of 4 km or 6 km, respectively.

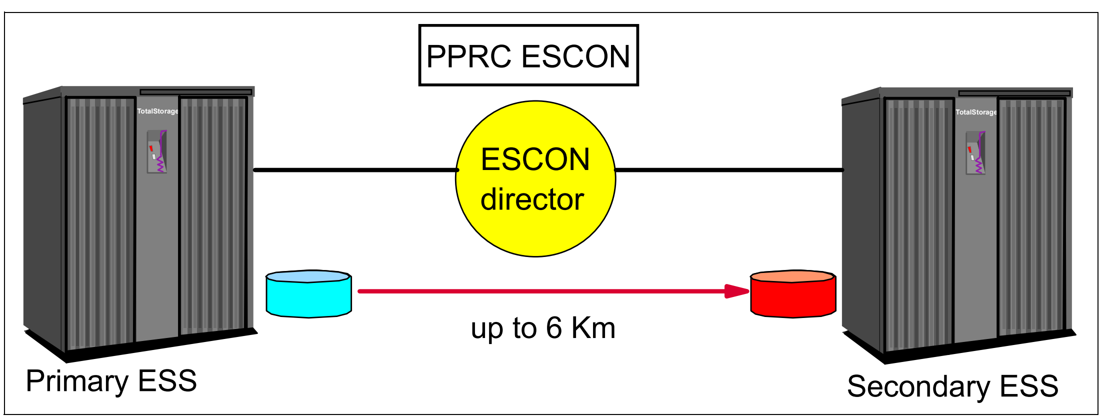



*Figure 3-33 Configuration with one ESCON Director - up to 6 km*

## ESCON configuration up to 26 km

Figure 3-34 illustrates a configuration with two ESCON Directors. On such configurations, only one of the ESCON Directors can be configured for a dynamic connection. Distances of up to 26 km between the two ESSs can be realized with this configuration:

- The distance between the ESSs and the ESCON Directors is 2 km with 50/125 μ fiber, or 3 km with 62.5/125 μ fiber.
- The distance between the two ESCON Directors can be 2 km for 50/125 μ fiber, 3 km for 62.5./125 μ fiber, or 20 km for 9/125 μ fiber.

The 9/125 μ fiber requires an ESCON-XDF adapter installed on the ESCON Directors.

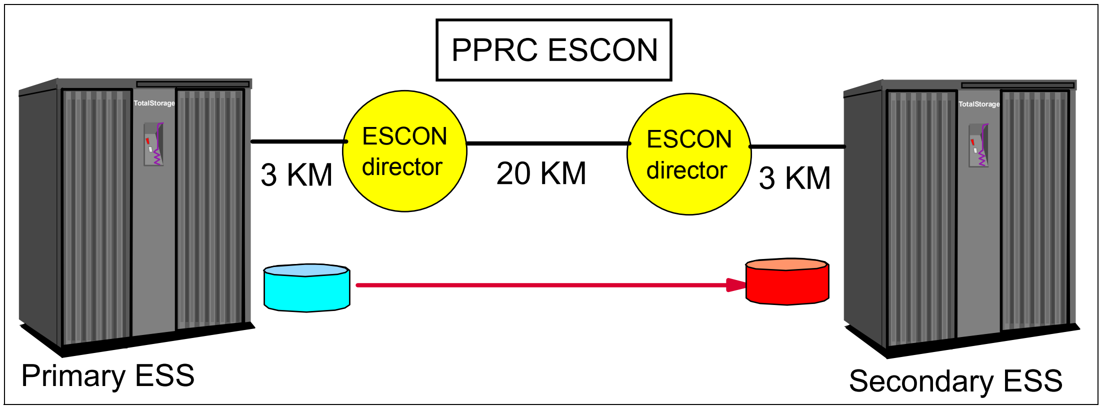



*Figure 3-34 Configuration with two ESCON Directors - up to 26 km*

## ESCON configuration up to 103 km

Figure 3-35 illustrates a configuration with the maximum supported distance of 103 km. This distance can be achieved using Dense Wave Division Multiplexers (DWDM) connecting through dark fibers, or using channel extenders over WAN lines. The example in Figure 3-35 presents a solution with the DWDMs connecting on one side to standard ESCON adapters, and the other side to a pair of dark fibers.

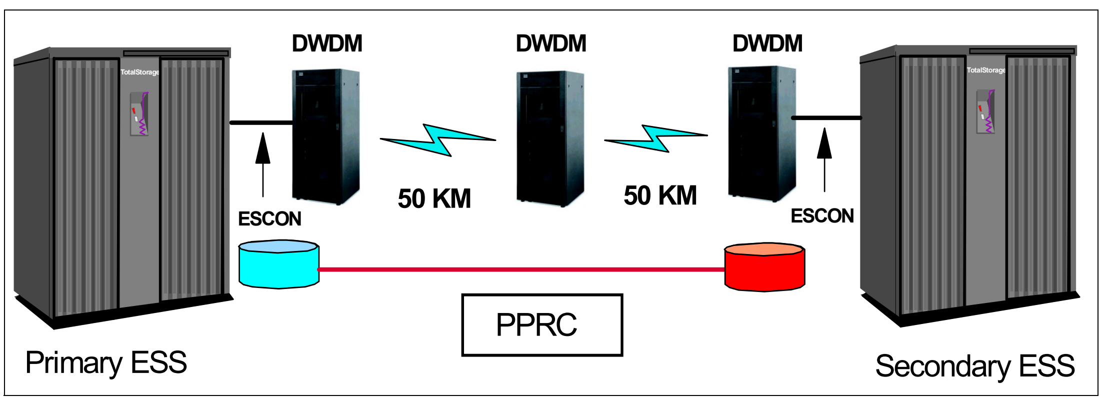



*Figure 3-35 Configuration to support 103 km*

> **Note:** Synchronous PPRC is distance sensitive. Some testing should be done before implementing long distance configurations in production. Also, with modeling tools (such as Disk Magic) the distance affect on the average I/O response time can be evaluated. Your IBM Field Technical Sales Specialist can be consulted for this.

### 3.10.3 PPRC connectivity using Fibre Channel ports

Fibre Channel can be used as the communications link between PPRC primary ESSs and secondary ESSs using Fibre Channel protocol (FCP). This allows exploitation of existing Fibre Channel infrastructure and capacity.

PPRC over Fibre Channel requires PPRC Version 2 (feature #85xx) and is supported on the ESS Model 800 starting at LIC level 2.3.0. Also, the ESS must have Fibre Channel/FICON host adapters (features #3024 and #3025) installed.

Compared to ESCON, FCP links for PPRC communication reduce the link infrastructure by at least 4 to 1 with equivalent or better performance.

When implemented over Fibre Channel links, PPRC can be managed using:

- ESS Copy Services Web User Interface (WUI) and, provided a supported server, the ESS Copy Services Command Line Interface (CLI) — for all environments
- ANTRQST API, TSO commands, and ICKDSF — for z/OS environments
- ICKDSF — for z/VM and VSE/ESA environments

#### Configuration guidelines

This section discusses the configuration guidelines that must be considered when planning the PPRC environment. The present section addresses the guidelines that apply when using FCP links.

##### *PPRC Fibre Channel links*

An ESS Model 800 or 750 Fibre Channel port can simultaneously be:

- Sender for a PPRC primary
- Receiver for a PPRC secondary
- Target for FCP host(s) I/O from open systems and Linux on zSeries — this does not apply to z/OS FICON host I/O.

A PPRC Fibre Channel link is bi-directional. This means that one link can have a PPRC path established in one direction and simultaneously have a PPRC path established in the opposite direction. This contrasts with ESCON links which are uni-directional. ESCON ports are dedicated and so cannot be used for host I/O.

As for ESCON links, FCP links support both Synchronous PPRC and all non-synchronous PPRC features.

Although one FCP link would have sufficient bandwidth for most PPRC environments, we recommend configuring two Fibre Channel links between each primary and secondary ESS, as illustrated in Figure 3-36.

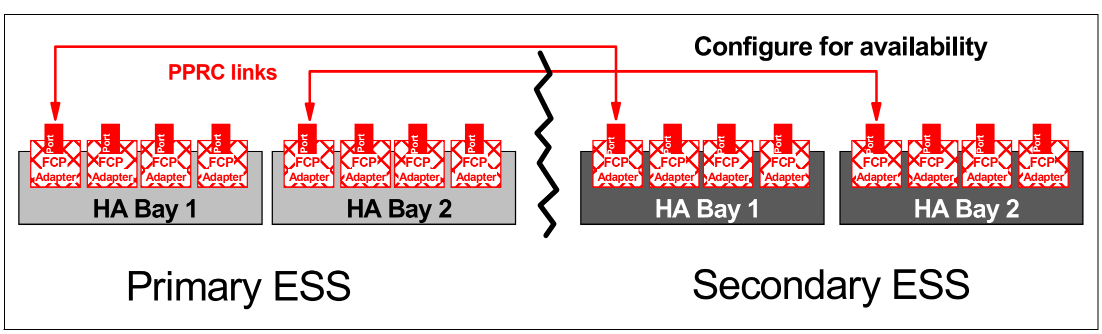



*Figure 3-36 Configuring for availability*

PPRC Fibre Channel links can be direct connected, or connected via up to two switches.

Figure 3-37 illustrates an example of FCP links direct connected between two ESS Fibre Channel ports. There is no sharing of these Fibre Channel ports; they are used exclusively for PPRC (both ports can be sending and receiving PPRC data at the same time due to the bi-directional capability of Fibre Channel).

Dedicating Fibre Channel ports for PPRC use guarantees no interference from host I/O activity. This is recommended with Synchronous PPRC, which is time critical and should not be impacted by host I/O activity. The PPRC ports used provide connectivity for all LSSs within the ESS and can carry multiple logical PPRC paths.

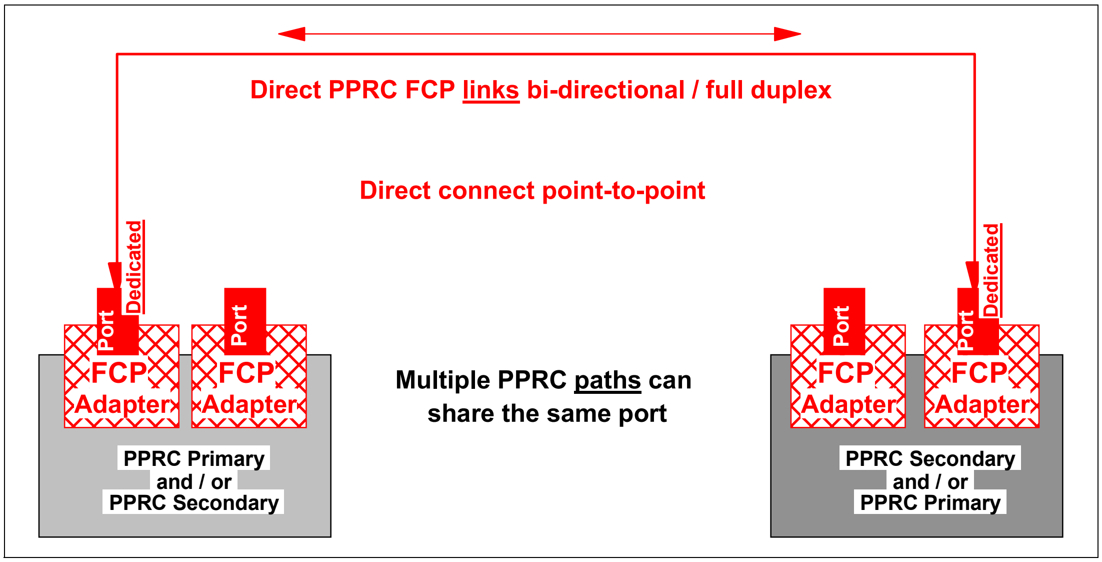



*Figure 3-37 Direct connected (dedicated) links*

Figure 3-38 shows a typical configuration for zSeries using two switches. As in the dedicated configuration, we recommend that the PPRC Fibre Channel ports be dedicated and not used for host I/O.

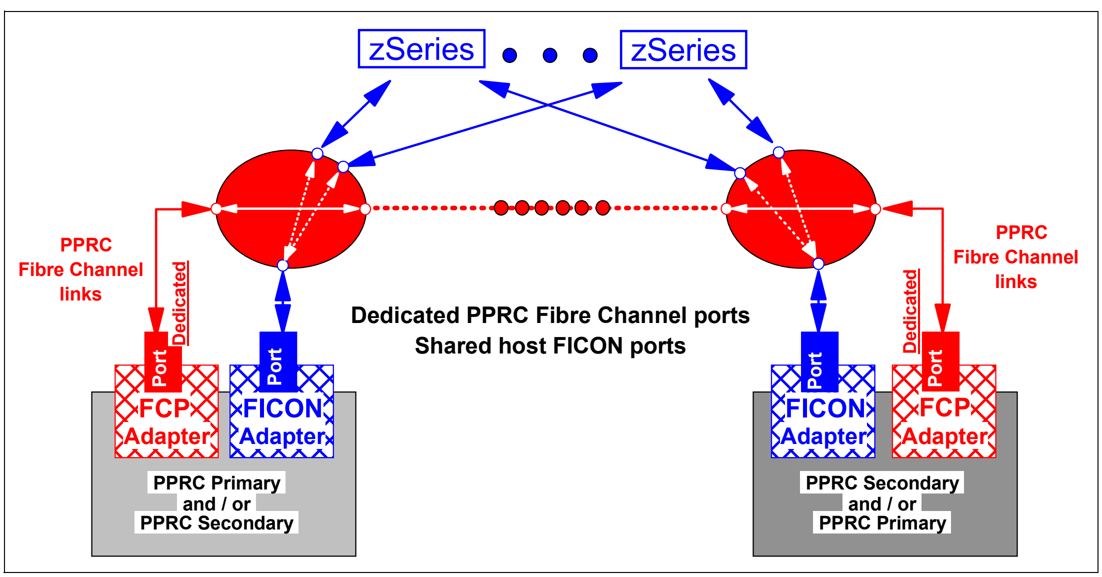



*Figure 3-38 Typical zSeries configuration*

### *Logical paths*

A logical path is a logical connection between the host channel and the I/O device control unit to perform device-level operations. This differs from a channel path, which is the collection of elements (links, ports, switches) that provide a physical link for the information flow. Figure 3-39 shows how multiple logical paths can use the same physical link.

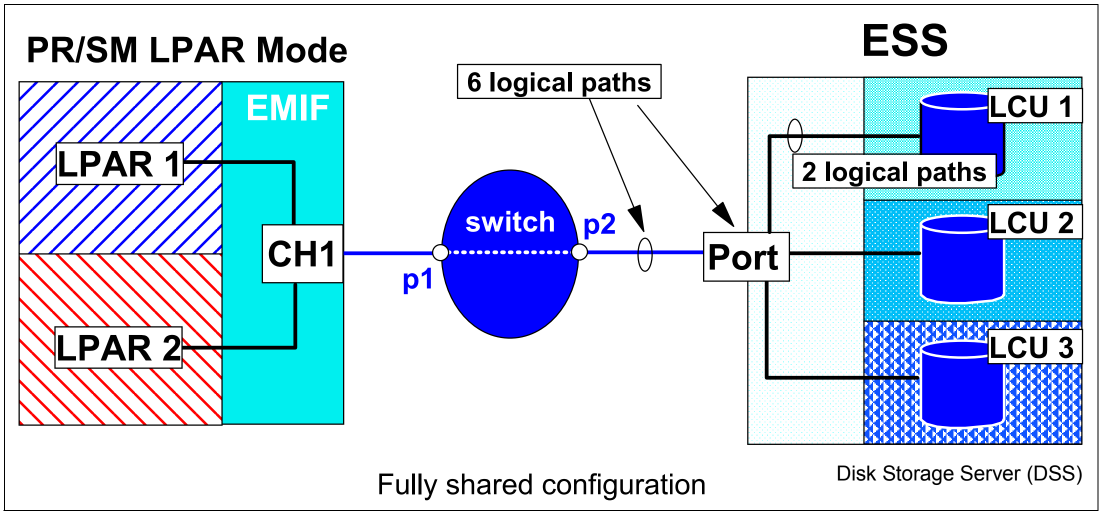



*Figure 3-39 Logical paths between zSeries host images and LSS/LCU*

Logical paths also apply to PPRC connections. A PPRC logical path is a logical connection between the sending LSS and the receiving LSS. An FCP link can accommodate multiple PPRC logical paths.

Figure 3-40 shows an example where we have a 1:1 mapping, and where the 3 logical paths are accommodated in one PPRC link:

- LSS1 in ESS1 to LSS1 in ESS2
- LSS2 in ESS1 to LSS2 in ESS2
- LSS3 in ESS1 to LSS3 in ESS2

Alternatively, if the volumes in each of the LSSs of ESS1 map to volumes in all three secondary LSSs in ESS2, there will be 9 logical paths over the PPRC link (not fully illustrated in Figure 3-40). Note that we recommend a 1:1 LSS mapping.

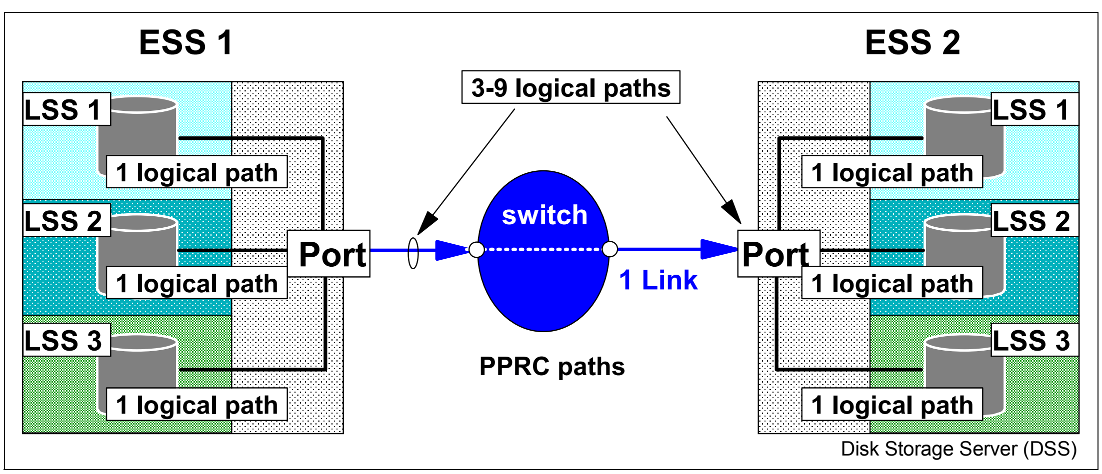



*Figure 3-40 Logical paths for PPRC*

PPRC over Fibre Channel links has not changed certain PPRC architectural characteristics, for example:

- A primary LSS can maintain paths to a maximum of four secondary LSSs. Each secondary LSS can reside in a separate ESS.
- Up to eight logical paths per LSS-LSS relationship can be defined. Each PPRC path requires a separate physical PPRC link. This is illustrated in Figure 3-41.
- The Copy Services Server (CSS) can manage up to 4096 volumes/LUNs. This is the total number including primary and secondary volumes/LUNs as well as source and target FlashCopy volumes/LUNs. This limit does not apply when using PPRC TSO commands.

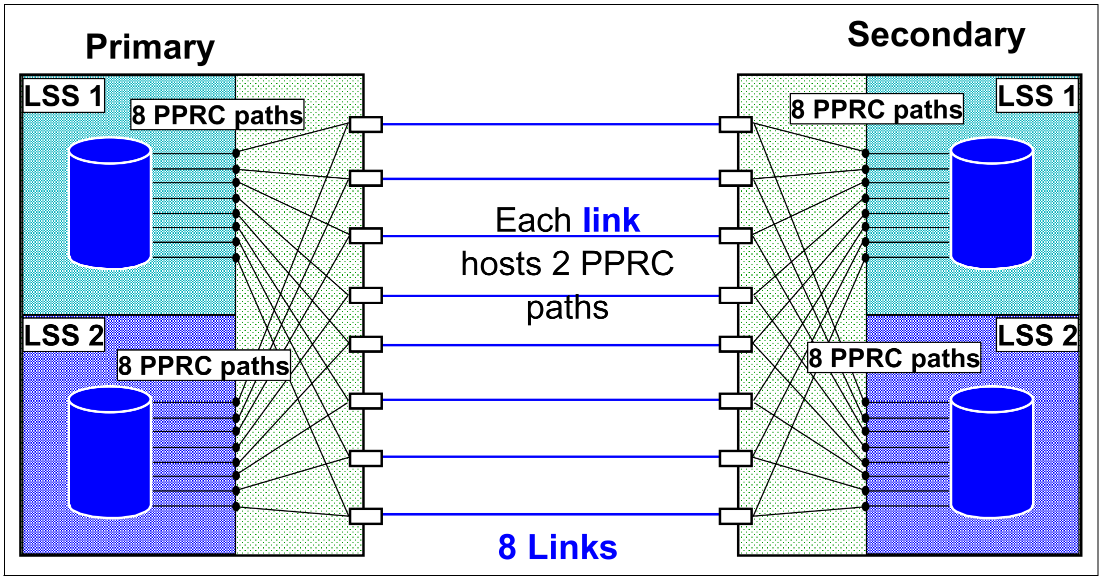



*Figure 3-41 Up to 8 paths per LSS-LSS relationship*

Some PPRC implementation constraints have been relieved when using Fibre Channel links. The new limitations are illustrated in Figure 3-42.

- An FCP port can host up to 1024 logical paths. These are the logical and directional paths that are made from LSS to LSS. The ESCON limit was 64.
- An FCP path (the physical path from one port (SAID) to another port (SAID)) can host up to 128 logical paths (PPRC paths). The ESCON limit was 64.
- An FCP port can accommodate up to 126 different physical paths (ESS port (SAID) to ESS port (SAID) through the SAN).

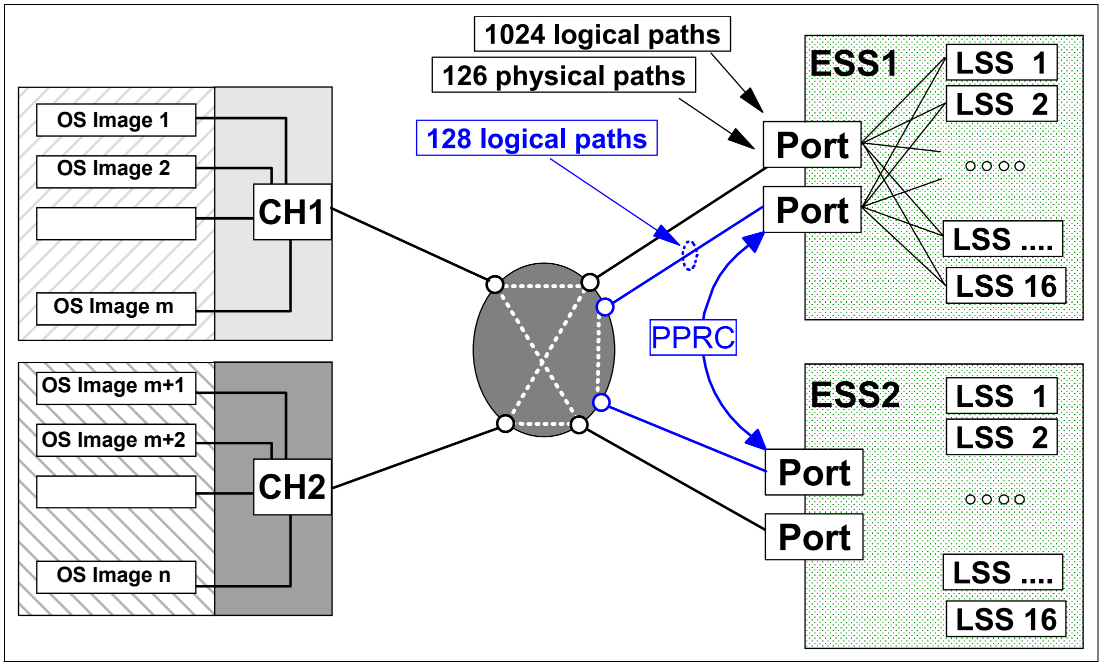



*Figure 3-42 Logical path/ physical path limits*

Table 3-6 compares FCP versus ESCON architectural characteristics.

*Table 3-6 FCP versus ESCON characteristics*

| Characteristic | FCP | ESCON |
|---|---|---|
| PPRC primary port accept host I/O | Yes | No |
| PPRC secondary port accept host I/O | Yes | Yes |
| Link is bi-directional (can have paths established in both directions simultaneously) | Yes | No |
| Support Synchronous PPRC | Yes | Yes |
| Supports PPRC Extended Distance | Yes | Yes |
| Number of secondary LSSs per primary LSS | 4 | 4 |
| Number of PPRC logical paths per LSS | 8 | 8 |
| Number of logical paths from a primary ESS | 1024 | 1024 |
| Number of logical paths per port | 1024 | 64 |
| Number of logical paths per physical path | 128 | 64 |
| Number of physical paths per port | 126 | 64 |

## Distance considerations

The distance between Fibre Channel ports without any channel extender infrastructure is dependent on the Fibre Channel port type. The ESS uses 2 Gbit GBICs on its Fibre Channel ports. For short wave GBICs using 50 micron multimode fiber, the maximum distance is 200 meters. With 62.5 micron multimode fiber, the maximum distance is 250 meters. For long wave GBICs over single-mode fiber, the maximum distance is 10 km.

Figure 3-43 illustrates the Fibre Channel maximum distances when switches and channel extending devices are not used.

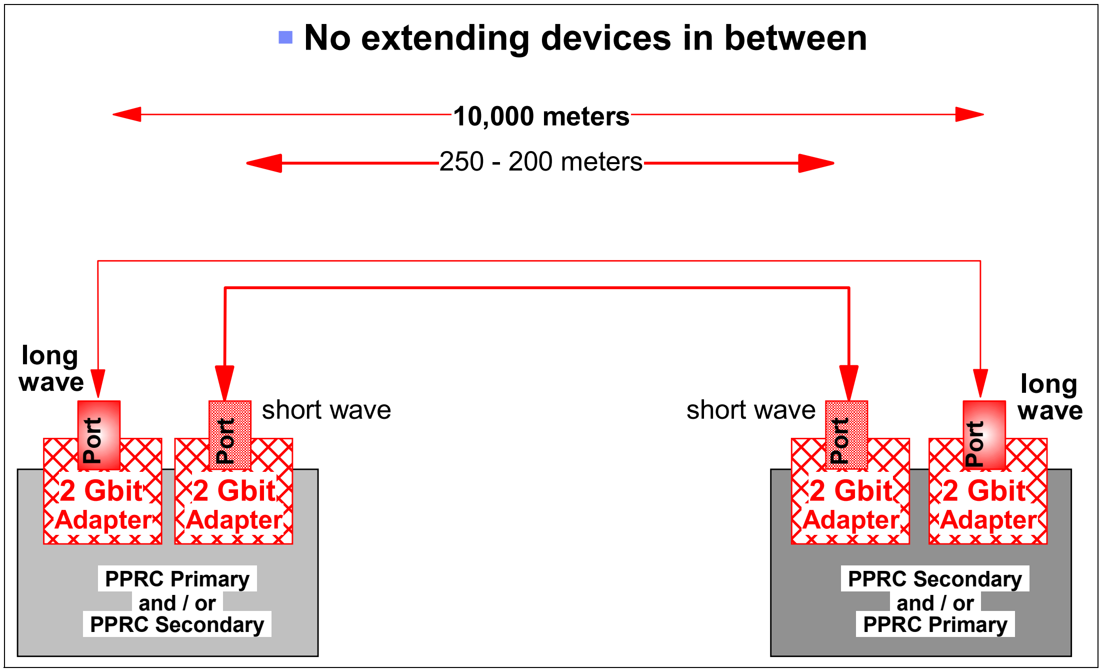



*Figure 3-43 PPRC Fibre Channel distances without switches*

These distances can be extended using Fibre Channel switches (refer to “SAN fabric and networking” on page 143 for more information). The maximum supported distance for Synchronous PPRC over FCP is 300 km. For PPRC-SYNC distances beyond 300 km an RPQ must be submitted.

## Fibre Channel Port System Adapter ID (SAID)

The Fiber Channel ports used for PPRC links are specified by their *system adapter* ID (SAID). The system adapter ID is specified within the LINK parameter of the TSO CESTPATH command and the ICKDSF ESTPATH command as part of the addressing of the physical link where the logical path is defined.

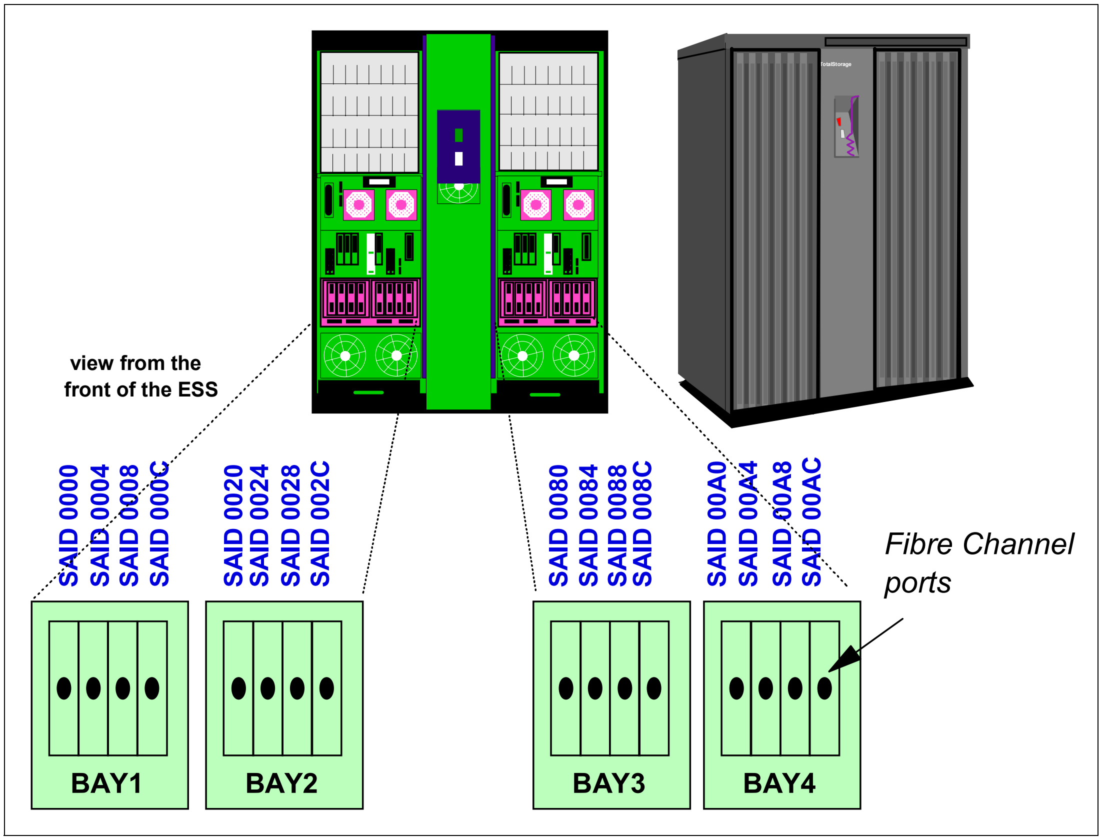



*Figure 3-44 System Adapter ID (SAID) with Fibre Channel ports*

Figure 3-44 illustrates how the FCP system adapter IDs (SAIDs) are designated when the host adapter bays are seen from the front of the ESS. The first two values are zeros, and the last two values are the hexadecimal SAID byte values that correspond to the Fiber Channel port interface of the ESS. See Appendix E, “System Adapter ID (SAID)” on page 391 for more information on the SAID.

## WWNN and WWPN

The WWNN (World Wide Node Name) is a single name given to an entity that is in a Fibre Channel network. A WWPN (World Wide Port Name) is the name of one of the ports into that entity. For example, if you look at a Windows® server with multiple HBAs, the code in the first adapter to get enabled will assign the WWNN of the server, and the second, third, and fourth adapters will all *inherit* the WWNN but will represent themselves as the WWPN that is burned into the host bus adapter (HBA).

In the case of the ESS, there is a single WWNN of the ESS that is reported in the ESS Specialist Welcome panel (see Figure 3-45). The WWNN is also reported by the TSO CQUERY PATHS command (see 4.2, “TSO commands for PPRC” on page 147) and the ICKDSF PPRCOPY QUERY command (see 4.3, “Using ICKDSF to control Copy Services” on page 156).

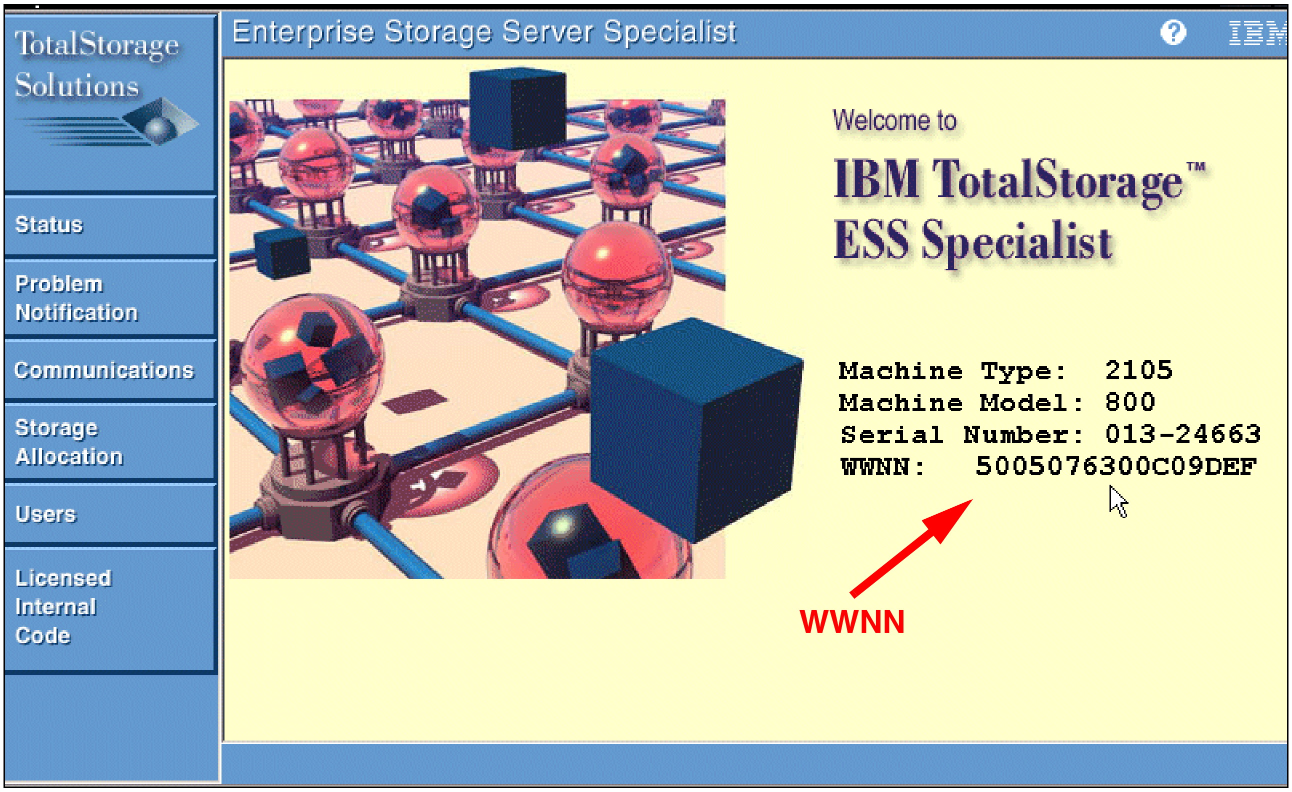



*Figure 3-45 ESS WWNN information in the ESS Specialist Welcome panel*

If you go into the ESS Specialist and click each of the Fibre Channel host adapters (see Figure 3-46), you will see that they all have a unique WWPN, but that the last 4 or 6 digits are all the same. In this case, instead of the burned-in WWPN, the ESS dynamically assigns the WWPN for each adapter. This way, if a host adapter needs to be replaced, it will retain its previous identity in the Fibre network.

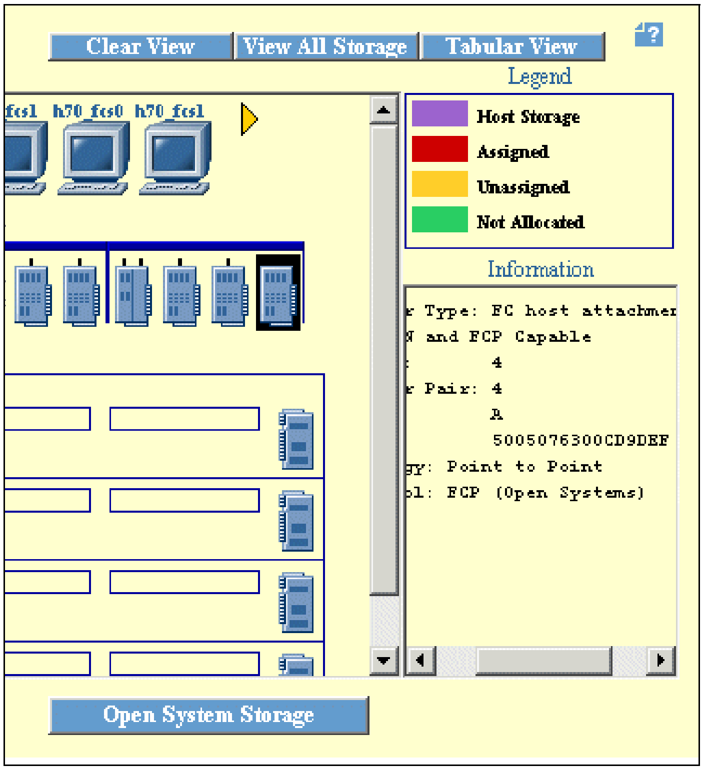


*Figure 3-46 WWPN of ESS Fiber Channel Port 4-4-A*

**Note:** When looking at the Information Panel for a selected Fibre Channel port from the ESS Copy Services Web User Interface Paths panel (see Figure 7-18 on page 321) you will see the WWNN and WWPN of the Fibre Channel switch to which the ESS FCP port is connected.

## Recognizing the ESS ports within the SAN fabric

In the ESS there is a function called *locally administered WWPNs*. The WWPN (World Wide Port Name) of a Fibre Channel port is calculated by a byte-level operation from the ESS node WWNN (World Wide Node Name, the one shown by the entry panel of ESS Specialist) and the adapter's port location in the ESS bay. Thus the adapter's WWPN remains unchanged when the adapter is replaced.

An advantage of this characteristic is that you can easily recognize your ESS ports in the SAN; all Fibre Channel switch GUIs have some window displaying the device's WWPNs attached to the fabric ports (with Brocade switches, it's the Name Server panel). You see there something like “IBM 2105 ... 50:05:07:63:00:c1:84:c8 ...”. When you know the way the WWPNs are identified in the ESS, you then can verify that the ESS ports are attached to the fabric ports as you want. This knowledge will also help you in other fabric management activities, for example, verifying that the correct WWPNs appear in zoning lists.

The way to determine how the FCP ports are assigned their WWPN in the ESS is as follows: Take the ESS node WWNN (you can get if from the ESS Specialist Welcome panel, as shown before in Figure 3-45 on page 141). Byte 6 in the WWNN is always c0 (in hexadecimal code). You get an FCP adapter's WWPN by replacing the x'c0' with the location code (c1...d0) for that adapter, as indicated in the diagram and example given in Figure 3-47.

```
ESS node or port id:          xx:xx:xx:xx:xx:yy:xx:xx    WWPN
                                             ||
                                             c0 or port id

Example:

  ESS node WWNN:              50:05:07:63:00:c0:84:c8

  ESS port WWPN:              50:05:07:63:00:yy:84:c8
                                             ||
                                     replace Port id from diagram below


ESS Port ids:
+---------------+---------------+---------------+---------------+
| Adapter Bay 1 | Adapter Bay 2 | Adapter Bay 3 | Adapter Bay 4 |
|  C4 C3 C2 C1  |  CC CB CA C9  |  C8 C7 C6 C5  |  D0 CF CE CD  |
+---------------+---------------+---------------+---------------+

For the last FCP port in Bay 4, the WWPN will be:

  ESS port WWPN:              50:05:07:63:00:cd:84:c8
```

*Figure 3-47 Determining the WWPN of and FCP port in the ESS*

## SAN fabric and networking

In addition to Fibre Channel link direct connection, PPRC over Fibre Channel is also supported with the use of SAN fabric products and DWDMs. All Fibre Channel switches and directors supported by the ESS today are also supported for PPRC over Fibre Channel.

For extending the PPRC Fibre Channel distances, the following DWDMs are supported:

- ADVA FSP 2000
- CIENA CN 2000 Storage Extension Platform
- Cisco ONS 15530 / 15540
- CNT Edge Storage Router
- Nortel Networks OPTera Metro 5200 / 5300

The DWDM vendor should be contacted regarding evaluation, qualification, approval, and hardware and software support prerequisites when using their products in an ESS PPRC configuration.

A current list of supported environments, configurations, networks, and products is available at:

http://www-1.ibm.com/servers/storage/support/disk/2105.html

4

# Managing Copy Services

In this chapter we discuss the different tools available to establish, control, manage, and monitor PPRC in the z/OS environment (see Figure 4-1). The following command and batch interfaces are covered, and management examples are described:

- TSO
- ICKDSF
- ANTRQST macro
- Web User Interface
- CLI
- API
- Multiple Device Manager Replication Manager
- Asynchronous PPRC Utilities for Open System Environment

We explain VM and VSE/ESA special considerations and provide information for planning of PPRC. This information should be read in conjunction with the production considerations and management techniques described in 4.8, “Planning for PPRC” on page 178.

### *Asynchronous PPRC*

The ESS Web Copy Services, the ESS Command Line Interface, ESS API and ICKDSF can currently be used to manage an Asynchronous PPRC session. It is planned to enable TSO, Multiple Device Manager, Asynchronous PPRC Utilities for Open System Environment, and the ANTQRST macro of z/OS and GDPS to control the sessions. Asynchronous PPRC Utilities for ICKDSF Users is also planned to be available with Asynchronous PPRC.

For Asynchronous PPRC management on z/OS, VM and VSE operating systems, ICKDSF is currently the only zSeries based interface.

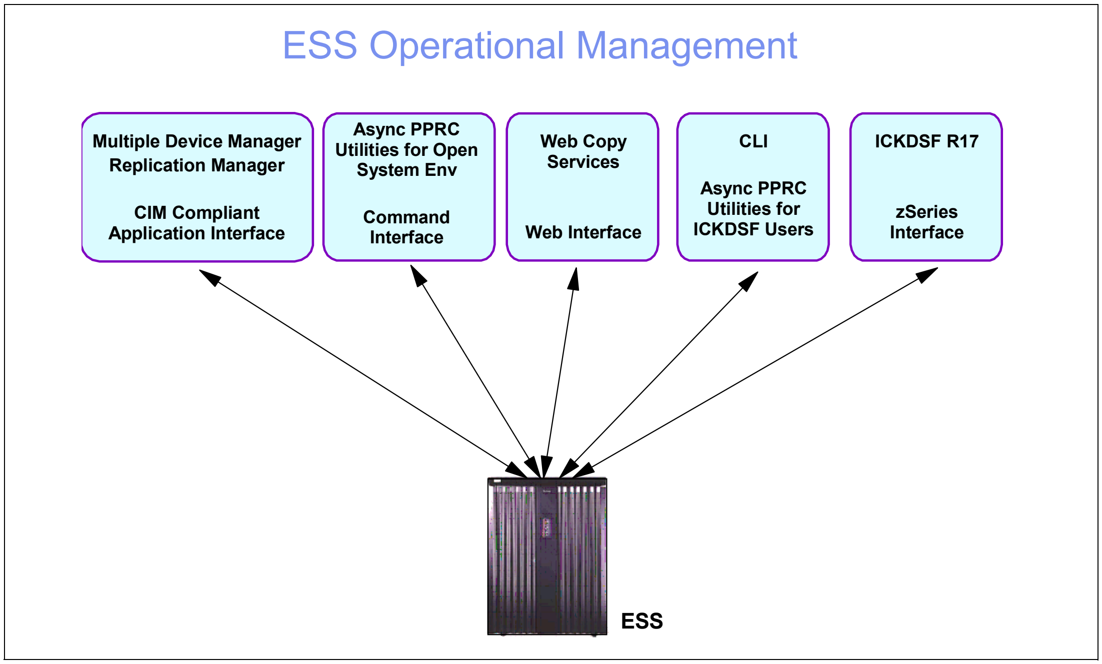



*Figure 4-1 ESS operational management*

## 4.1 Introduction to the PPRC management tools

PPRC can be managed by the following means when operating in a zSeries environment.

### *TSO commands*

TSO commands give a direct command line interface from an authorized TSO user login. This allows for a quick and convenient interface as well as enabling command batching and automation. These are discussed in 4.2, “TSO commands for PPRC” on page 147.

### *ICKDSF utility — PPRCOPY commands*

The ICKDSF utility is a powerful tool and very useful for repeat processes or when batch processing is preferred. The commands are discussed in 4.3, “Using ICKDSF to control Copy Services” on page 156.

### *ANTRQST macro*

The ANTRQST macro provides an application program call to the z/OS system data mover’s (SDM) application programming interface (API). This macro allows you to call PPRC, as well as the XRC and FlashCopy functions.

The ANTRQST macro is discussed in 4.4, “ANTRQST macro” on page 175.

### *Web User Interface*

The ESS Copy Services provides a Web User Interface (WUI) which requires a standard browser on a desktop workstation to access. While not directly accessible from z/OS it provides a useful tool to graphically view and manage Copy Services on the ESS.

The Web User Interface is discussed in Chapter 7., “ESS Copy Services Web User Interface” on page 299 and also in 4.9.4, “Example for starting and monitoring an Asynchronous PPRC Session” on page 195.

### *Command Line Interface (CLI)*

The command line interface can be used from a selected set of supported open systems servers, including Windows, AIX, Solaris, HP-Unix and others. Normally used only with open systems implementations with a suitable host, the CLI can be used to manage the ESS. If the CLI is required, details can be found in the companion book *IBM TotalStorage Enterprise Storage Server Implementing ESS Copy Services in Open System Environments*, SG24-5757.

### *Application Programming Interface (API)*

Starting with LIC level 2.3.0, the ESS API is enhanced to support ESS Copy Services configuration and use for PPRC and FlashCopy. It supports these activities through the use of the interface as defined by the Storage Networking Industry Association (SNIA) Storage Management Initiative Specification (SMI-S). The ESS API is discussed in more detail in Appendix A, “ESS Application Programming Interface (API)” on page 351.

### *Geographically Dispersed Parallel Sysplex (GDPS)*

GDPS is a high availability disaster recovery and continuous availability solution. This solution manages and controls critical data mirroring and efficiently balances workload between the sites. The GDPS solution uses automation and Parallel Sysplex technology to help manage multi-site databases, processors, and PPRC configurations, thus maintaining data consistency and application availability. GDPS is discussed in Appendix B, “Geographically Dispersed Parallel Sysplex (GDPS)” on page 359.

### *Remote Copy Management Facility (RCMF)*

RCMF significantly simplifies the storage administrator’s remote copy management functions by managing the entire PPRC configuration. RCMF is discussed in Appendix B, “Geographically Dispersed Parallel Sysplex (GDPS)” on page 359.

## 4.2 TSO commands for PPRC

In z/OS and OS/390 systems, PPRC can be managed using TSO commands. In this section we discuss the PPRC commands that TSO provides. At the time of writing, TSO command support is not available for Asynchronous PPRC operation and PPRC Version 2 functions.

Before you can use PPRC TSO commands, you have to define them to RACF. See 4.8.2, “Defining TSO commands to RACF” on page 178.

For a detailed description of these commands, the following IBM publication can be referenced: *z/OS DFSMS Advanced Copy Services,* SC35-0428.

### Command overview

An overview of these commands is presented in Table 4-1.

*Table 4-1 TSO Commands*

| Command | Description |
|---|---|
| CESTPAIR | Establishing PPRC volume pairs |
| CESTPATH | Establish PPRC path |
| CDELPAIR | Deleting volume pairs |
| CDELPATH | Deleting PPRC paths |

| Command | Description |
|---|---|
| `CGROUP` | Controlling volume groups with FREEZE and RUN |
| `CQUERY` | Querying of the status of volumes and paths |
| `CRECOVER` | Recovering data on the recovery system |
| `CSUSPEND` | Suspending PPRC volume pairs |

### CESTPAIR

This command is used to establish the PPRC primary to secondary relationship. The primary and secondary volumes must have the same number of tracks on each cylinder, and the same number of bytes on each track. The secondary device must have the same number, or a greater number of cylinders, than the primary device (see “Refreshing the VTOC” on page 164).

This command has parameters that allow it to indicate whether the operation is an initial establish of volumes that were in the simplex state, or whether it is a re-synchronization of a suspended pair of volumes (Example 4-1).

*Example 4-1 PPRC establish pair*

```
CESTPAIR DEVN(X’1142’)
   PRIM(‘X6060’ 62019 X’42’ X’00’)
   SEC(X’8061’ 68006 X’42’ X’00’)
   MODE(COPY)
   MSGREQ(YES)
```

In Example 4-1, the primary volume resides on LSS **`x’00’`** of the application site ESS. Its CCA is **`x’42’`**. The secondary volume resides in LSS **`x’00’`** of the recovery site ESS. Its CCA is **`x’42’`**. This matching between the last two digits of the device number address and the CCAs of the primary and the secondary volumes, although it is not mandatory, it is useful for working with the configuration. CCA is discussed in “Channel connection address” on page 151.

The MSGREQ(YES) parameter specifies that PPRC wait until the initial full volume copy operation is complete before issuing the completion message **`ANTP0001I`**. If you are using automation, use this message as a trigger for other operations.

### CESTPAIR command with PPRC-XD

This command has the optional parameter OPTION, which has two mutually exclusive values: SYNC or XD. You specify SYNC if you want to establish the pair in the typical synchronous duplex state. You specify XD if you want to establish the pair in the non-synchronous duplex pending XD state.

For the pair to become an XD pair (in addition to specifying the XD value for the OPTION parameter) you will also specify whether this pair comes from the simplex or from the suspended state. That is to say, whether this is an initial copy of a newly established pair, or if it is a re-synchronization of a suspended pair. To indicate this, you use the MODE parameter specifying either COPY or RESYNC, respectively.

> **Note:** The CRIT(YES) and the MODE(NOCOPY) parameter values are mutually exclusive with the OPTION(XD) parameter value, when used in a PPRC V1 environment.
>
> With PPRC V2 the MODE(NOCOPY) parameter value is allowed with OPTION(XD).

Example 4-2 shows the CESTPAIR command with the OPTION parameter value XD and MODE parameter value of COPY. This example illustrates a transition from a simplex volume state to a *duplex-pending XD* volume state.

*Example 4-2 Establishing a PPRC-XD pair - CESTPAIR command*

```
CESTPAIR DEVN(X'F40')
PRIM(X'A763' FCA76 X'4F' X'03')
SEC(X'A762' FCA76 X'10' X'02')
MODE(COPY)
OPTION(XD)
```

Table 4-2 summarizes the values of the CESTPAIR command parameters that you will be using when working with XD pairs. Refer also to Table 4-2 for the parameter values when doing a go-to-SYNC transition.

*Table 4-2 CESTPAIR parameter values for the PPRC-XD transitions*

| **Operation** | **Volume pair state** | | **CESTPAIR parameter values** | |
|---|---|---|---|---|
| | **from** | **to** | **OPTION** | **MODE** |
| Establish initial copy pair | *simplex* | *duplex pending* XD | XD | COPY |
| Reestablish suspended pair | *suspended* | *duplex pending* XD | XD | RESYNC |

### *CESTPATH over ESCON paths*

This command is used to establish logical paths (over the ESCON links) between a primary LSS and a secondary LSS. Each CESTPATH command can establish up to eight paths from a primary LSS to a secondary LSS. The following considerations apply:

- Each primary LSS can connect to up to four secondary LSSs — using four CESTPATH commands.
- Each secondary LSS can connect to any number of primary LSSs, as long as there are enough ESCON connections.
- Paths must be established before any volume pairs can be established (CESTPAIR command).

The ESS of the primary LSS keeps information about the paths that connect it to the secondary LSS. The ESS of the primary LSS automatically attempts to restart a failed path, and then resumes copy operations. If all paths to the secondary volume’s LSS fail and cannot be recovered, the PPRC pairs are suspended. These paths should be re-established with the CESTPATH command after the cause of the failure has been corrected (Example 4-3).

*Example 4-3 PPRC Establish Path*

```
CCESTPATH DEVN(X’1142')
   PRIM(X'6060' 62019 X'00')
   SEC(X'8060' 68006 X'00')
   LINK(X'0000B000' X'0020B000' X'0080B400' X'00A0B400')
```

The four paths established in Example 4-3 connect LSS x'00' on the primary ESS (s/n 62019) to LSS x'00' on the secondary ESS (s/n 68006) through ESCON directors. For the LINK parameter, these are the values:

- The first two digits are always zero, followed by the ESS ESCON System Adapter ID (SAID). In Example 4-3 the following SAIDs were used: 00, 20, 80, and A0.
- The fourth and fifth digits are zero if the ESSs are connected through point-to-point ESCON links. If they are connected through ESCON directors (as shown in Figure 4-3) then the output ports on the first ESCON director have to be specified.
- The last two digits indicate the target LSS number.

### ESCON System Adapter ID (SAID)

The ESCON adapters are specified by their *system adapter* ID (SAID). The system adapter ID is specified within the LINK parameter of the CESTPATH command as part of the addressing of the physical link where the logical path is defined (Figure 4-2).

**Note:** When using Fiber Channel ports, see the discussion in 3.10.3, "PPRC connectivity using Fibre Channel ports" on page 134.

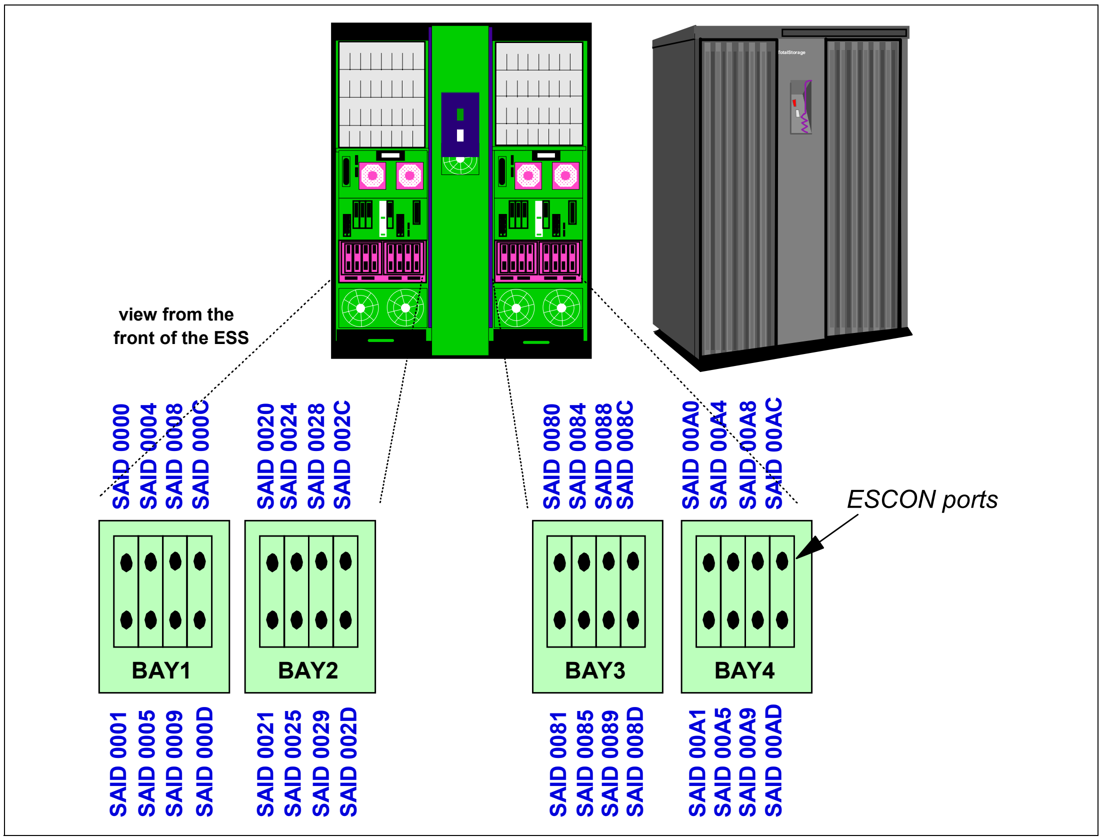



*Figure 4-2 System Adapter ID (SAID) identification with ESCON ports*

Figure 4-2 illustrates how the system adapter IDs (SAIDs) are designated when the host adapter bays are seen from the front of the ESS. The first two values are zeros (00aa), and the last two values are hexadecimal SAID byte values that correspond to ESCON channel interfaces of the primary ESS. See Appendix E, "System Adapter ID (SAID)" on page 391 for more on this topic.

## Channel connection address

The *channel connection address* (CCA) on an ESS is always relative to the LSS, and is always between 00 and FF. If the device numbers on the ESS have been generated as a contiguous group of 256 addresses, where each group belongs to an LSS (logical control unit), then the CCAs that are going to be used for the PPRC commands are the last two digits of the device number.

For example, for device numbers AA00 - AAFF, the corresponding CCAs are 00 - FF. If you did not define the device addresses contiguously, you can get the CCA for a device by using the TSO CQUERY command, or the DEVSERV system command.

## Subsystem identifier

During the installation and configuration process, each logical control unit (that is each LSS) is assigned a unique four-digit subsystem identifier (SSID). This setting is required to identify the control unit for the host for error reporting reasons and also for functions like PPRC.

Each LSS must have a different SSID. It is recommended to assign consecutive SSID numbers to each LSS within an ESS box. This will simplify the configuration procedure and make it easier for the subsequent identification of the LSSs. LSSs attached to the same operating system image must have different SSIDs: *SSIDs must be unique*. z/OS checks during the IPL to insure that there are no duplicate SSIDs.

When establishing PPRC paths, the SSIDs are used in the PPRC TSO commands to identify the primary and secondary LSSs that will be paired (Figure 4-3).

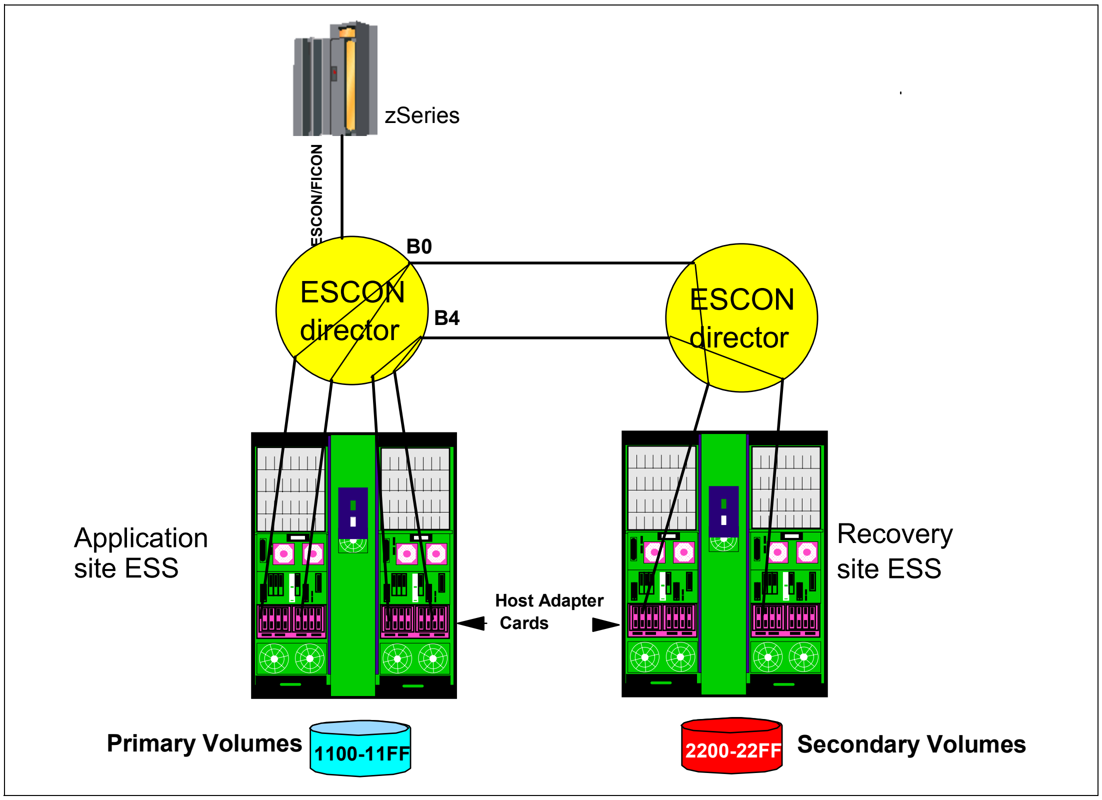



*Figure 4-3 PPRC paths established*

Figure 4-3 illustrates how the PPRC links come out from different ESCON host adapters in the primary ESS. It is recommended to spread the PPRC links across ESCON host adapters and host adapter bays.

**Tip:** The CESTPATH command is a “replace” function. Specified paths, which establish a link between a primary LSS and a secondary LSS, replace any previously established PPRC paths. Each time you issue a CESTPATH command, the paths specified on it replace the paths that were established by the last CESTPATH command that was issued.

### CESTPATH syntax for paths using Fibre Channel links

When establishing a path using Fibre Channel links, the WWNN is used instead of the ESS serial number when defining the primary and secondary LSS. Also, the link address has a new format: aaaabbbb, where:

aaaa = SAID for the primary subsystem Fibre Channel adapter
bbbb = SAID for the secondary subsystem Fibre Channel adapter

See Example 4-4.

*Example 4-4 CESTPATH Command for FCP*

```
CESTPATH DEVN(device_number) -
        PRIM(ssid WWNN lss) -
        SEC(ssid WWNN lss) -
        LINK(linkaddr)
```

### CDELPAIR

The CDELPAIR command is used to specify the primary and secondary volumes to be removed from a PPRC pairing. This command should be directed to the primary device; see Example 4-5.

*Example 4-5 CDELPAIR command*

```
CDELPAIR DEVN(X’1142’)
   PRIM(‘X6060’ 62019 X’42’ X’00’)
   SEC(X’8061’ 68006 X’42’ X’00’)
```

Before issuing a CDELPAIR command, it should be verified that there are some active paths between the respective primary and secondary LSSs. If all of the paths that were established between both LSSs have been disabled, the CDELPAIR command will fail. After the failing CDELPAIR command times out, the state of the primary volume is simplex. The secondary volume remains in its previous state (duplex, pending, or suspended). You then need to issue a CRECOVER command to return the secondary volume to the simplex state.

### CDELPATH

The CDELPATH command is used to delete established PPRC paths between a primary LSS and the corresponding secondary LSS; see Example 4-6. Only the paths to the specified secondary LSS are affected — all other paths to other LSSs are not affected.

Before issuing a CDELPATH command, a CDELPAIR command to all PPRC volume pairs should be issued. The CDELPATH command may cause an `ANTP0121I` message if this sequence is not followed.

*Example 4-6 CDELPATH command*

```
CDELPATH DEVN(X’1142')
   PRIM(X'6060' 62019 X'00')
   SEC(X'8060' 68006 X'00')
```

### CDELPATH syntax for paths using Fibre Channel links

When deleting paths using Fibre Channel links, the WWNN is used instead of the ESS serial number for the primary and secondary LSS; see Example 4-7.

*Example 4-7* CDELPATH command for FCP

```
CDELPATH DEVN(device_number) -
         PRIM(ssid WWNN lss) -
         SEC(ssid WWNN lss)
```

### CGROUP

The CGROUP command is used to control operations for multiple PPRC volume pairs on an given LSS:

**CGROUP FREEZE**
Suspends PPRC operations for all PPRC volumes on the specified LSS pair. With the FREEZE parameter, PPRC volume pairs are suspended and PPRC paths are deleted for the specified LSS pair.

If the PPRC paths for the LSS pair were defined with Consistency Group enabled (CGROUP parameter of the CESTPATH command), then an Extended Long Busy (ELB) condition is initiated when CGROUP FREEZE is issued for the LSS. During the ELB window, update activity cannot proceed in the LSS where the freeze was done.

**CGROUP RUN**
Resumes all operations for the PPRC volumes on the specified LSS. The PPRCOPY RUN command is used in order to reset the ELB condition and release application I/O to the primary volumes.

**Tip:** The CGROUP FREEZE command deletes the paths between the specified LSS pair. To restart normal PPRC operation back again, the paths will have to be re-established and — if the I/O activity on the primary volumes was not interrupted during the freeze — the pairs will have to be re-synchronized.

Example 4-8 illustrates the CGROUP command.

*Example 4-8 CGROUP FREEZE and RUN commands*

```
CGROUP DEVN(X’1142') PRIM(X'6060' 62019 X'00') SEC(X'8060' 68006 X'00') FREEZE
CGROUP DEVN(X’1142') PRIM(X'6060' 62019 X'00') SEC(X'8060' 68006 X'00') RUN
```

**Note:** A single LSS can be paired with up to four other LSSs. Therefore, you may have to issue up to four CGROUP commands to suspend all pairs on a single primary LSS.

### CQUERY

The CQUERY command is used to query the status of one volume of a PPRC volume pair or all paths that are associated with the LSS for the device number that is specified. The CQUERY command can be issued to either a primary or secondary PPRC volume. A host system that is attached only to a primary volume cannot obtain the status of the secondary volume for that pair. In the same way, a host attached only to the secondary volume cannot obtain the status of the primary volume.

In Example 4-9, a CQUERY command for volume 1142 is issued.

*Example 4-9 CQUERY command*

```
CQUERY DEVN(X'1142')
```

The output of the CQUERY command (shown in Figure 4-4) provides volume related information, such as SSID, CCA, LSS number, and ESS serial number.

The TSO CQUERY command output for an XD pair returns the *duplex-pending XD* state and the number of out-of-sync tracks.

```
************** PPRC REMOTE COPY CQUERY - VOLUME ********************
*                                         (PRIMARY)   (SECONDARY) *
*                                        SSID CCA LSS SSID CCA LSS*
*DEVICE   LEVEL      STATE     PATH STATUS  SERIAL#     SERIAL#    *
*------ ---------  ----------  ----------- ---------    ---------  *
* 1142  PRIMARY..  DUPLEX....   ACTIVE..   6060 42 00   8061 42 00 *
*       CRIT(NO).......       CGRPLB(NO). 000000062019 000000068006*
* PATHS SAID/DEST STATUS: DESCRIPTION                              *
* ----- --------- ------  -------------------                      *
*   1   0020 0000   01    PATH ESTABLISHED...                      *
*       ---- ----   00    NO PATH............                      *
*       ---- ----   00    NO PATH............                      *
*       ---- ----   00    NO PATH............                      *
********************************************************************
```

*Figure 4-4 CQUERY VOLUME command output for PPRC with ESCON*

The TSO CQUERY command output for an XD pair returns the *duplex-pending XD* state and the number of out-of-sync tracks.

Figure 4-5 illustrates the output of a CQUERY command addressed to volume x'F40', which is the primary volume of an XD pair. You can see in the output information, the STATE of the volume is PENDING.XD. The pair will remain in this condition, never reaching the duplex state unless commanded to do so.

Also, from the output information, you can check the number of primary updated tracks, which at that moment were not reflected on the secondary volume (TRACKS OUT OF SYNC).

In addition, you can see the PPRC volume state of the device and how many PPRC paths are established.

```
CQUERY DEVN(X'F40')

CQUERY FORMATTED LVL 2
VOLUME REPORT
************************ PPRC REMOTE COPY CQUERY - VOLUME *******************************
*                                                    (PRIMARY)  (SECONDARY)     *
*                                                    SSID CCA LSS SSID CCA LSS  *
*DEVICE      LEVEL       STATE          PATH STATUS  SERIAL#     SERIAL#       *
*----------- -------------- ----------------- -------------------- ------------ ------------ *
* 0F40       PRIMARY..   PENDING.XD     ACTIVE..    A763 4F 03   A762 10 02    *
*            CRIT(NO).......            CGRPLB(YES) 0000000FCA76 0000000FCA76  *
* PATHS      SAID/DEST STATUS:   DESCRIPTION                                   *
* ----------  ------ --------  ------------  ----------------------------------------- *
*  1  0020 EA02   01         PATH ESTABLISHED...                     *
*             ------ ------   00         NO PATH............                   *
*             ------ ------   00         NO PATH............                   *
*             ------ ------   00         NO PATH............                   *
* IF  STATE  =  PENDING/SUSPEND:  TRACKS OUT OF SYNC =       19471             *
*                                 TRACKS ON VOLUME    =       33390             *
*                                 PERCENT OF COPY COMPLETE =  42%              *
*****************************************************************************************
CQUERY COMMAND COMPLETED FOR DEVICE 0F40. COMPLETION CODE: 00
```

*Figure 4-5 Output information for a CQUERY VOLUME on a primary XD volume*

The output from CQUERY displays Fibre Channel information appended to the end of the ESCON format information. If the device is in SIMPLEX status or the paths are ESCON, the default ESCON format is used.Figure 4-6 shows the output from CQUERY PATHS for a device with paths using Fibre Channel links.

```
*************** PPRC REMOTE COPY CQUERY - PATHS *******************
* PRIMARY UNIT: SERIAL#= 000000024409 SSID= B000 SS= 2105 LSS= 00  *
*                  FIRST        SECOND        THIRD        FOURTH      *
*                SECONDARY    SECONDARY     SECONDARY    SECONDARY    *
*SERIAL NO: 000000024215 000000032054 ............ ............ *
* SSID LSS:    C000  00      4200  02      .......      .......      *
*     PATHS:        1             0             0             0       *
*            PFCA SFCA S*  SAID DEST S*  SAID DEST S*  SAID DEST S* *
*            --------- --  --------- --  --------- --  --------- -- *
*         1: 0004 001F 13  00EC DA00 01  ---- ---- 00  ---- ---- 00 *
*         2: 0005 0020 13  00B4 7F00 01  ---- ---- 00  ---- ---- 00 *
*         3: 0006 0021 13  00CD 2900 01  ---- ---- 00  ---- ---- 00 *
*         4: 0007 0022 13  ---- ---- 00  ---- ---- 00  ---- ---- 00 *
*  SUBSYSTEM         WWNN                                           *
* -----------  ----------------                                     *
* PRIMARY       5005076300CABCDE                                    *
* SECONDARY 1   5005076300CEDCBA                                    *
* SECONDARY 2   5005076300C7FD28                                    *
*                                                                   *
* S* = PATH STATUS:                                                 *
* 00=NO PATH          01=ESTABLISHED          02=INIT FAILED        *
* 03=TIME OUT         04=NO RESOURCES AT PRI 05=NO RESOURCES AT SEC*
* 06=SERIAL# MISMATCH 07=SEC SSID MISMATCH    08=ESCON LINK OFFLINE *
* 09=ESTABLISH RETRY  0A=PATH ACTIVE TO HOST 0B=PATH TO SAME CLUSTR*
* 10=CONFIGURATION ERROR                                            *
* 13=ESTABL FIBRE PTH 14=FIBRE PATH DOWN      15=FIBRE RETRY EXCEED *
* 16=SEC ADPTR INCAP  17=SEC ADPTR UNAVAIL    18=FIBRE LOGIN EXCEED *
*******************************************************************
ANTP0001I CQUERY COMMAND COMPLETED FOR DEVICE 0F40. COMPLETION CODE: 00
```

*Figure 4-6 CQUERY PATHS output with PPRC over Fibre Channel*

### CRECOVER

The CRECOVER command is used to allow the recovery system to gain control of the volumes. This command is issued from the recovery system (see Example 4-10). It signals the recovery ESS to force the secondary volume into simplex state, thus allowing the recovery system to gain control of the volume.

During this procedure the volume serial number can be verified and also it can be relabeled. The volume can be varied online after this command is complete.

*Example 4-10 CRECOVER command*

```
CRECOVER DEVN(X’2242’)
   PRIM(‘X6060’ 62019 X’42’ X’00’)
   SEC(X’8061’ 68006 X’42’ X’00’)
   ID(VOL002 VOL001)
   MSGREQ(YES)
```

In Example 4-10, the CRECOVER command brings device x’2242’ (on the recovery ESS) to simplex state. It also changes the volume label from VOL002 to VOL001.

### CSUSPEND

This command is used to suspend PPRC operations between a volume pair. PPRC stops mirroring data to the secondary volume and starts keeping record of the primary volume tracks that are updated. The information of which tracks were updated while the pair was suspended will be used later when the pair is re-established, to copy just the updated tracks. In Example 4-11 a CSUSPEND command is issued for PPRC primary volume 1142.

*Example 4-11 PPRC SUSPEND pair*

```
CSUSPEND DEVN(X’1142’) PRIM(‘X6060’ 62019 X’42’ X’00’) SEC(X’8061’ 68006 X’42’ X’00’)
```

### Batch execution of PPRC TSO commands

Batch procedures can be automated to manage PPRC activities. JCL can be used to issue PPRC commands in batch jobs, and automation procedures can be set to watch for their completion. Automation could analyze the results from previous job executions or PPRC messages, and then schedule other jobs to be executed. Example 4-12 illustrates a batch job for executing a PPRC TSO command.

*Example 4-12 Batch job used for executing PPRC TSO commands*

```
//IKJEFT01 JOB MSGLEVEL=(1,1),MSGCLASS=A,NOTIFY=user id
//STEP1 EXEC PGM=IKJEFT01
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
   CDELPATH DEVN(X’1142')
   PRIM(X'6060' 62019 X'00')
   SEC(X'8060' 68006 X'00')
```

## 4.3 Using ICKDSF to control Copy Services

For the z/OS, VM and VSE operating systems ICKDSF is currently the only zSeries based interface. Seven new ICKDSF PPRC commands have been added to the already existing command set, and five new ICKDSF FlashCopy commands are introduced to control an Asynchronous PPRC Session. These commands will be described in the following section.

The complete syntax is documented in *Device Support Facility User's Guide and Reference,* GC35-0033.

The ESS Web Copy Services, the ESS Command Line Interface, ESS API and ICKDSF can currently be used to manage an Asynchronous PPRC session. It is planned to enable TSO, Multiple Device Manager, Asynchronous PPRC Utilities for Open System Environment, the ANTQRST macro of z/OS and GDPS to control the sessions. Asynchronous PPRC Utilities for ICKDSF Users is also planned to be delivered on a CD with the Enterprise Storage Server.

### 4.3.1 Setup of PPRC and FlashCopy with ICKDSF

The following PPRC commands have already been available for years. All of them start with **PPRCOPY** or **PPRC** in front of each command:

| | |
|---|---|
| **ESTPATH** | Establishes PPRC paths between a primary and secondary LSS |
| **DELPATH** | Deletes PPRC paths between primary and secondary LSSs |
| **ESTPAIR** | Establishes a PPRC primary to secondary volume relationship |
| **RECOVER** | Allows a system to regain control of a volume on the secondary ESS |
| **SUSPEND** | Puts a PPRC volume pair in suspended state |
| **FREEZE** | Suspends all PPRC operations at the LSS level |
| **RUN** | Resumes I/O operations after a freeze with extended long busy |
| **QUERY** | Queries the status of a PPRC volume pair and the status of paths |

The following additional commands are introduced with Asynchronous PPRC. All of the following parameters have **PPRCOPY** (or **PPRC**) in front of them:

| | |
|---|---|
| **DEFINESESSION** | Sessions are OPENed (created) and CLOSed (terminated) with an associated Session ID to LSSs |
| **POPULATESESSION** | Populates or depopulates the previously OPENed session with volumes |
| **QUERY ASYNCCOPY** | Queries data about the Asynchronous PPRCOPY configuration |
| **QUERY OUTOFSYNCSTATE** | Queries data about the number of volumes in the specified session that are out-of-sync |
| **QUERY SESSIONDEVICES** | Queries status of the session open on the LSS |
| **STARTASYNCCOPY** | Starts or modifies the configuration of an Asynchronous PPRC session |
| **TERMASYNCCOPY** | Pauses or terminates Asynchronous PPRC for the specified session |

These FlashCopy commands are completely new with ICKDSF. All of the following parameters have the command **FLASHCOPY** (or **FLASH** or **FC**):

| | |
|---|---|
| **ESTABLISH** | Establishes full volume relationships |
| **QUERY** | Returns volume level data |
| **QUERY INCREMENTSTATUS** | Returns information on Consistency Groups, FlashCopy information and out-of-sync tracks |
| **QUERY RELATIONS** | Determines the status of FlashCopy relations and the state of inconsistent FlashCopy volumes when Asynchronous PPRC fails |
| **WITHDRAW** | Terminates FlashCopy relationships, enables Revert or Commit |

## General ICKDSF job usage

All of the commands have to be addressed to a device, that is either online or offline to the system the job is submitted from. For offline volumes, the ICKDSF job has to address the volumes directly by specifying UNITADDRESS; see Example 4-13.

*Example 4-13 Addressing offline volumes*

```
//DELPTHRV JOB (ACCT),'DEL REVRSE PPRC PATH',
//        CLASS=A,MSGCLASS=H
//STEP01  EXEC PGM=ICKDSF
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
  PPRCOPY DELPATH                   -
          UNITADDR(3000)            -
          ..................
```

Online volumes cannot be addressed directly and DD statements with the volume serial number have to be used. This assures the access to the volume serial number of the volume, regardless of what real address this volume resides on; see Example 4-14.

*Example 4-14 Addressing online volumes by using DD statements*

```
//DEFSESS  JOB (ACCT),'DEFINE ASYNC SESS',
//        CLASS=A,MSGCLASS=H
//STEP01  EXEC PGM=ICKDSF
//VLB000   DD VOL=SER=VLB000,UNIT=SYSALLDA,DISP=SHR
//VLB200   DD VOL=SER=VLB200,UNIT=SYSALLDA,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
   PPRC  DEFINESESSION              -
         DDNAME(VLB000)             -
          ...........
   PPRC  DEFINESESSION              -
         DDNAME(VLB200)             -
          ...........
```

### *Display the Fibre Channel Connection Information Table*

The ANALYZE command allows the user to specify the WWNN of the secondary PPRC volume connected via Fibre Channel links. When PPRC Fibre Channel support is present, the Analyze pathing reports will then include the *Fibre Channel Connection Information Table*. This information indicates the potential connectivity of the Fibre Channel ports in the ESS where the I/O is issued to each system adapter port in the ESS specified by the secondary WWNN.

**SECWWNN(X'swwnn')**

The SECWWNN parameter:

- Specifies the World Wide Node Name (WWNN) of the volume that you want Fibre Channel Connection information for. This command will return information about the adapters available to the primary (represented by the UNIT parameter) that are able to connect to the secondary device represented by SECWWNN. The ‘swwnn’ is the WWNN, an 8-byte hexadecimal value X'wwwwwwwwwwwwwwww'.

For example, consider the following command:

```
ANALYZE UNIT(200) NODRIVE NOSCAN SECWWNN(x'1020304050607080')
```

This would generate the following table showing that device 200 with WWNN='0102030405060708' can potentially connect to a device with WWNN='1020304050607080' from:

PRI SAID 00 TO THE SECONDARY TAGS 00 THRU 0F
PRI SAID 01 TO THE SECONDARY TAGS 04, 05, 06
PRI SAID 02 TO THE SECONDARY TAGS 0D, 0E, 0F
PRI SAID 03 TO THE SECONDARY TAGS 00 THRU 02, 0E, 0F

*Example 4-15 ANALYZE NODRIVE NOSCAN output*

FIBRE CHANNEL CONNECTION INFORMATION TABLE

| PRI WWNN | PRI SAID | SEC WWNN | SEC TAG |
|---|---|---|---|
| 0102030405060708 | 00 | 1020304050607080 | 00-0F |
| 0102030405060708 | 01 | 1020304050607080 | 04<br>05<br>06 |
| 0102030405060708 | 02 | 1020304050607080 | 0D<br>0E<br>0F |
| 0102030405060708 | 03 | 1020304050607080 | 00-02<br>0E<br>0F |

## PPRCOPY DELPAIR

The DELPAIR command is used to specify a PPRC pair that will be removed from a PPRC relationship returning the devices to a simplex state. An example of the JCL is shown in Example 4-16.

*Example 4-16 ICKDSF DELPAIR command example*

```
PPRCOPY DELPAIR                         -
         UNITADDRESS(120C)                -
         PRIMARY(X'1132',18767,X'0C')   -
         SECONDARY(X'1142',12320,X'0C') -
         LSS(X'02',X'02')
```

## PPRCOPY DELPATH

The DELPATH command is used to delete the defined PPRC paths between a primary LSS and a secondary LSS. Only active paths to the specified recovery site LSS are affected; all other paths to other LSSs are unaffected. Refer to the PPRCOPY ESTPATH command for a description of the link or WWNN parameter.

Example 4-17 shows how to specify the PPRCOPY DELPATH command to delete a PPRC path which uses Fibre Channel links.

*Example 4-17 Using ICKDSF to delete paths defined over Fibre Channel links*

```
PPRCOPY DELPATH UNIT(4C80)
        PRI(X'2901',FCA29) SEC(X'2905',FCA29) -
        LSS(X'01',X'05') -
        FCPPATHS(x'002000E6' -
        WWNN(x'0102030405060708',x'1020304050607080')
```

## PPRCOPY ESTPATH for ESCON Channel

This command is used to establish paths between the primary and the secondary LSSs. Each ESTPATH command can establish up to eight paths from one application site LSS to a single recovery site LSS. The following considerations apply:

- For PPRCOPY ESTPATH, the last byte of the link address must be the logical subsystem number of the secondary ESS.

  The LINK parameter of the PPRCOPY ESTPATH command has the following syntax: LINK(x’aaaabbcc’), where:

  **aaaa** System Adapter ID (SAID). See “ESCON System Adapter ID (SAID)” on page 150, for more information on the SAIDs.

  **bb** ESCON Director destination address (x’00’ if directly attached ESS or static ESCON Director connection)

  **cc** Logical subsystem number of the secondary LSS

- You can determine the LSS number by running the ICKDSF CONTROL CONFIGURE(DISPLAY) command.
- The FORCE parameter is used to force removal of existing logical paths. If the links used for PPRC do not have existing logical paths established, then FORCE is not necessary.
- When FORCE is used and existing logical paths are removed, you will get the following z/OS messages:
  - `IOS001E CCUU,INOPERATIVE PATH CHP`
  - `IOS450E CCUU,CHP, NOT OPERATIONAL PATH TAKEN OFFLINE`

## PPRCOPY ESTPATH for Fibre Channel

New parameters and sub-parameters are added to PPRCOPY ESTPATH for Fibre Channel support:

- `FCPPATHS(path1,...,path8)`
- `WWNN(X'pwwnn',X'swwnn')`

### `FCPPATHS(path1,...,path8)`

The FCPPATHS parameter is used to indicate that PPRC paths are to be established using Fibre Channel links. Up to 8 paths can be specified where each path is an 8-digit hexadecimal address in the form X’aaaabbbb’, where:

- ‘aaaa’ is the primary system adapter ID (SAID)
- ‘bbbb’ is the secondary system adapter ID (SAID)

Paths using Fibre Channel links and ESCON links cannot be established between the same PPRC pair or specified in the same PPRCOPY ESTPATH command. The FCPPATHS parameter is not valid with the LINK parameter.

The FCPPATHS parameter is only supported on an ESS with ESS LIC 2.3.0 or higher. Otherwise the following message is issued:

```
ICK31054I Device not supported for specified function
```

**WWNN(X'pwwnn',X'swwnn')**

This specifies the World Wide Node Name of the primary and secondary ESS. Each WWNN is an 8-byte hexadecimal value X'wwwwwwwwwwwwwwww' where X'pwwnn' represents the primary WWNN and X'swwnn' represents the secondary WWNN.

The WWNN can be determined via:

- ESS Copy Services Web User Interface
- ICKDSF CONTROL CONFIGURE(DISPLAY) command
- ICKDSF ANALYZE NODRIVE NOSCAN output

Example 4-18 shows how to specify the PPRCOPY ESTPATH command to establish a PPRC path using Fibre Channel links.

*Example 4-18 Using ICKDSF to define one path over a Fibre Channel link*

```
PPRCOPY ESTPATH UNIT(4C80)
        PRI(X'2901',FCA29) SEC(X'2905',FCA29) -
        LSS(X'01',X'05') -
        FCPPATHS(x'002000E6') -
        WWNN(x'0102030405060708',x'1020304050607080')
```

Example 4-19 shows how to specify the PPRCOPY ESTPATH command to establish multiple PPRC paths using Fibre Channel links.

*Example 4-19 Using ICKDSF to define multiple paths over Fibre Channel links*

```
PPRCOPY ESTPATH UNIT(4C80)
        PRI(X'2901',FCA29) SEC(X'2905',FCA29) -
        LSS(X'01',X'05') -
        FCPPATHS(x'002000E6', x'002000E7') -
        WWNN(x'0102030405060708',x'1020304050607080')
```

## PPRCOPY ESTPAIR

The ESTPAIR command is used to establish a PPRC relationship between a primary and a secondary volume.

Example 4-20 shows the PPRCOPY ESTPAIR command with the optional parameter OPTION. In the example OPTION(XD) was specified, so a PPRC-XD pair is established.

*Example 4-20 PPRCOPY ESTPAIR*

```
PPRCOPY ESTPAIR UNIT(4C80) LSS(X’01’,X’05’) -
PRI(X’2901’,FCA29,X’00’)
SEC(X’2905’,FCA29,X’00’)
MODE(COPY) OPTION(XD)
```

FAILOVER and FAILBACK are optional parameters of the PPRCOPY ESTPAIR command. The PPRC Failover and Failback functions are described in 3.5.1, “PPRC Failover and Failback modes” on page 91, and a sample failover and failback procedure is discussed in Chapter 8, “Solutions” on page 409.

The ICKDSF PPRCOPY ESTPAIR command allows you to establish a PPRC volume pair in XD mode, and also to transition the volume pair from XD to synchronous mode. The command includes an OPTION parameter, which can have the following values:

- **SYNC**: This value establishes a PPRC volume pair in synchronous mode.
- **XD**: This value establishes a PPRC volume pair in non-synchronous mode.

### *PPRCOPY ESTPAIR with Asynchronous Cascading PPRC*

An additional parameter exists for Asynchronous Cascading PPRC invocation:

`CASCADE` This parameter indicates that the volume is eligible for PPRC cascading. It is used with the volume that will become the intermediate volume of an Asynchronous Cascading PPRC relationship.

If PPRCOPY ESTPAIR with CASCADE is issued to a volume which is in PPRC simplex state, this only indicates that the volume is eligible for cascading. The volume does not become part of a cascading PPRC configuration until it is established as an intermediate volume.

If PPRCOPY ESTPAIR with CASCADE is issued to an intermediate volume, and the volume is already the primary volume of a remote PPRC pair, then the volume becomes eligible for cascading without breaking this existing remote PPRC pair.

If PPRCOPY ESTPAIR with CASCADE is issued to a volume that is to become the intermediate volume and the volume is already the secondary of an existing local PPRC pair, the volume becomes the primary of the remote PPRC pair and is then part of a cascaded relationship.

When PPRCOPY ESTPAIR with CASCADE is issued to a storage device not supporting PPRC Version 2, the command will be terminated, with the message `ICK31054I DEVICE NOT SUPPORTED FOR SPECIFIED FUNCTION.`

If PPRCOPY ESTPAIR without CASCADE is issued to a volume that is part of a PPRC pair, the cascading attribute remains unchanged. If a volume is set as eligible for cascading, you must break the PPRC pair in order to reset the cascading state.

Example 4-21 shows an ESTPAIR command issued to a volume at the intermediate site, to establish a PPRC remote pair (intermediate site to remote site). The CASCADE parameter makes the intermediate volume eligible for cascading.

*Example 4-21 PPRCOPY ESTPAIR command with CASCADE parameter*

```
PPRCOPY   ESTPAIR UNIT(1410)        -
          PRI(X'1134' 18767 X'04')  -
          SEC(X'1154' 22331 X'04')  -
          LSS(X'04' X'04') -
          OPTION(XD) -
          MODE(NOCOPY) -
          CASCADE
```

## PPRCOPY FREEZE

The PPRCOPY FREEZE command suspends operations for the PPRC volumes on a given LSS pair. It is issued to control operations for multiple PPRC volume pairs on the specified LSS pair. With the FREEZE command, PPRC volume pairs are suspended and PPRC paths are deleted.

## PPRCOPY QUERY

The QUERY command is used to query the PPRC status of a volume and the associated PPRC paths. A query can be issued to either a primary or secondary PPRC volume.

A host system that is attached only to a primary volume cannot obtain the status of the secondary volume for that pair. In the same way, a host that is attached only to the secondary volume cannot obtain the status of the primary volume.

When the QUERY is issued to the volume at the intermediate site of an Asynchronous Cascading PPRC configuration, by default it will return the secondary volume information (the secondary volume information as it relates to the local PPRC pair). The returned data will have a flag set indicating that the volume is a cascading PPRC volume. If this flag is set, then ICKDSF will re-issue the QUERY with a flag set to obtain the primary volume information (the primary volume information as it relates to the remote PPRC pair).

The output has changed to include the path information for Fibre Channel links. Due to the length of the information added for PPRC Fibre Channel support, the QUERY PATHS output displays each secondary LSS following the previous secondary, rather than across the page.

Example 4-22 shows the output to a PPRCOPY QUERY PATHS command for a device with PPRC paths using Fibre Channel links.

*Example 4-22 PPRCOPY QUERY PATHS output*

```
QUERY REMOTE COPY - PATHS

 PRIMARY UNIT SERIAL#=  FCA29            SSID= 2901
 PRIMARY LSS NUMBER= 01

 FIRST
 SECONDARY
 ---------
 SERIAL NO:     FCA29
 SSID:   2905
 PATHS:    4

        SAID DEST   WWNN            STATUS
        -----------------------------------
 PATH 1: 0020 E605 0102030405060708    01
 PATH 2: 0020 E705 0102030405060708    01
 PATH 3: 0021 E605 0102030405060708    01
 PATH 4: 0021 E705 0102030405060708    01
 PATH 5: ---- ---- ----------------    00
 PATH 6: ---- ---- ----------------    00
 PATH 7: ---- ---- ----------------    00
 PATH 8: ---- ---- ----------------    00


SECOND
SECONDARY
---------
SERIAL NO:     ABC234
SSID:   3456
PATHS:    4

        SAID DEST   WWNN            STATUS
        -----------------------------------
 PATH 1: 0020 E605 0102030405060708    01
 PATH 2: 0020 E705 0102030405060708    01
```

```
PATH 3: 0021 E605 0102030405060708    01
PATH 4: 0021 E705 0102030405060708    01
PATH 5: ---- ---- ----------------    00
PATH 6: ---- ---- ----------------    00
PATH 7: ---- ---- ----------------    00
PATH 8: ---- ---- ----------------    00
```

## PPRCOPY RECOVER

The RECOVER command is used to allow the recovery system to regain control of a DASD volume. This command is issued from the recovery system. It signals the recovery site ESS to remove the volume from the PPRC relationship (the volume becomes simplex), and thus gives the volume control back to the recovery system. During this process the volser can be verified, and also the volume can be relabeled.

## PPRCOPY SUSPEND

The SUSPEND command is used to suspend PPRC mirroring between a primary and the corresponding secondary volume. When the SUSPEND command is directed to the primary or secondary device of a PPRC volume pair, the pair is suspended, and data is no longer mirrored onto the secondary volume. At the same time PPRC starts tracking the primary volume tracks that are updated while the pair is suspended. This way, when the pair is re-established with the PPRCOPY ESTPAIR command, the option MODE(RESYNC) can be used to copy just the updated tracks to synchronize the volumes again. An example of the JCL is shown in Example 4-23.

*Example 4-23 ICKDSF Suspend example JCL*

```
PPRCOPY SUSPEND                         -
          DDNAME(VLB000)                  -
          PRIMARY(X'1340',23953,X'00')    -
          SECONDARY(X'1070',24663,X'00') -
          LSS(X'00',X'00')
```

## PPRCOPY RUN

The PPRCOPY RUN command resumes operations for the PPRC volumes on a given LSS pair.

If an extended long busy (ELB) condition was triggered by a previous FREEZE command, then the PPRCOPY RUN command can be used to reset the ELB condition and resume the I/O updates on the primary LSS volumes.

## Refreshing the VTOC

If the secondary volume has more cylinders than the primary, the VTOC must be rebuilt in order to reflect the size of the volume before starting to use it. Example 4-24 illustrates an example of an ICKDSF batch job used for this purpose.

*Example 4-24 Refreshing VTOC information - ICKDSF job*

```
//IKJEFT01 JOB MSGLEVEL=(1,1),MSGCLASS=A,NOTIFY=user id
//STEP1 EXEC PGM=ICKDSF
//SYSPRINT DD SYSOUT=*
//DD1 DD DISP=SHR,UNIT=3390,VOL=SER=VOL001
//SYSIN DD *
   REFORMAT DDNAME(DD1) VERIFY(VOL001) REFVTOC
/*
```

**Note:** To run the job to refresh the secondary volume's VTOC, the PPRC volume pair must first be deleted.

## 4.3.2 ICKDSF commands for Asynchronous PPRC

Asynchronous PPRC is based on existing PPRC-XD pairs and persistent FlashCopy relations at the remote site, with PPRC secondary volumes being FlashCopy source volumes. ICKDSF can establish this base configuration with the previously mentioned commands ESTPATH and ESTPAIR in XD mode. FlashCopy in z/OS up to now is usually managed by DFSMSdss or TSO commands, but now ICKDSF can be used as well. Special options in ICKDSF enable the usage for Asynchronous PPRC.

### Establish FlashCopy relationship at the remote site

This can easily be done by addressing the command to the PPRC-XD primary volumes using Inband FlashCopy. For a detailed description of the options, see the syntax of the command in the *Device Support Facility User's Guide and Reference*, GC35-0033, and refer to Example 4-25. The options in this example have to be used for Asynchronous PPRC.

For disaster recovery purposes, a *Fast Reverse Restore* of the data from the FlashCopy target volumes to the FlashCopy Source volumes is possible, using the source volume with the data of the latest FlashCopy relationship. See Chapter 8, "Solutions" on page 409 for more details. The SOURCEVOL Parameter is used to address the Inband FlashCopy source volume on a PPRC secondary ESS.

***Syntax:***

**FLASHCPY ESTABLISH**

```
DDNAME(dname)|SYSNAME(sysxxx)|UNITADDRESS(ccuu)
TARGETVOL(lss,cca,cuuu)

Optional Parameters
SOURCEVOL(lss,cca,ssid,ser#,,cuuu)
MODE(COPY | NOCOPY)
ONLINTGT(NO | YES)
TGTOKASPPRCPRIM(NO | YES)
TGTCANCOMEONLINE(NO|YES)
RESTORE
INHIBITTARGETWRITES(NO|YES)
FASTREVERSERESTORE
CHANGERECORDING(NO|YES))
```

*Example 4-25 Establish FlashCopy Pair*

```
   FLASH   ESTABLISH -
          DDNAME(VLB000)                     -
          SOURCEVOL(X'00',X'00',X'1070',24663) -
          TARGETVOL(X'01',X'00',3100)         -
         INHIBITTARGETWRITES(YES) -
         CHANGERECORDING(YES) -
         MODE(NOCOPY) -
```

You can use this command to create a second FlashCopy target volume to the same source volume for testing or application development purposes. Only one FlashCopy relation is used for Asynchronous PPRC.

## Reverse a FlashCopy relationship

In disaster recovery cases with Asynchronous PPRC you need to use the consistent volumes at the remote site. The same ESTABLISH command with the FASTREVERSERESTORE parameter, directed to the PPRC secondary volumes, can be used to reverse (FlashCopy back) the FlashCopy target volumes with consistent data back to the FlashCopy source volumes. These volumes can then be varied online and the applications can be started. See Example 4-26. Before you do this operation, a failover to the PPRC secondary volumes should have been done. See Chapter 8, “Solutions” on page 409.

*Example 4-26 Reverse FlashCopy pair*

```
FLASH   ESTABLISH                            -
          UNITADDRESS(3100)                    -
          TARGETVOL(X'00',X'00',3000)          -
          FASTREVREST                          -
          TGTOKPRIM(YES)

FLASH   ESTABLISH                            -
          UNITADDRESS(3101)                    -
          TARGETVOL(X'00',X'01',3001)          -
          FASTREVREST                          -
          TGTOKPRIM(YES)
```

## Withdraw FlashCopy relationship

This can easily be done by addressing the command to the PPRC-XD primary volumes using Inband FlashCopy. See the syntax of the command and refer to Example 4-27.

### *Syntax:*

**FLASHCPY WITHDRAW**

```
DDNAME(dname)|SYSNAME(sysxxx)|UNITADDRESS(ccuu)

Optional Parameters
TARGETVOL(lss,cca,cuuu)
SOURCEVOL(lss,cca,ssid,ser#)
MODE(COPY | NOCOPY | ALL)
REVERT | COMMIT
RESETTGTWRTINHIBIT
```

*Example 4-27 Establish FlashCopy pair*

```
   FLASH   WITHDRAW -
         DDNAME(VLB000)                     -
         TARGETVOL(X'01',X'00',3100)        -
         SOURCEVOL(X'00',X'00',X'1070',24663)
```

## Open and close an Asynchronous PPRC Session

The first step to prepare an Asynchronous PPRC Session is the definition of the session. This has to be done for a group of PPRC primary logical subsystems that should be included in the session regardless of the location in the Master or Subordinate ESSs. The determination of the Master LSS is done by the STARTASYNCCOPY command in the third step. Currently only one session per ESS and therefore one Master per ESS is allowed. A session number from 1 to 255 has to be assigned, and the included LSSs can be addressed by any one of its volumes. This is an LSS related command.

See the syntax of the command and refer to Example 4-28, which defines a session with ID 50 for two LSSs.

***Syntax:***

**PPRCOPY DEFINESESSION**

```
DDNAME(dname)|SYSNAME(sysxxx)|UNITADDRESS(ccuu)
OPEN|CLOSE
SESSIONNO(sessionnumber)
```

*Example 4-28 Defining and opening a session.*

```
  PPRC    DEFINESESSION            -
          DDNAME(VLB000)           -
          SESSIONNO(50)            -
          OPEN
  PPRC    DEFINESESSION            -
          DDNAME(VLB200)           -
          SESSIONNO(50)            -
          OPEN
```

The status of the session can be queried by the QUERY ASYNCCOPY command.

## Add and remove volumes to an Asynchronous PPRC Session

In the second step, all the primary volumes of the previously defined LSSs, which should be part of the Asynchronous PPRC session, have to be inserted into the session. One or more ranges of channel connection addresses (CCA) can be specified for each LSS that belongs to a session. Volumes can be joined or removed dynamically any time a session exists. Individual volumes or volume ranges can be specified in the command. Only PPRC primary volumes can join a session.

If the session later is started with the STARTASYNCCOPY command *before* the PPRC-XD initial copy has ended, the volumes remain in *joined pending* state as long as they have not reached the first full copy status during the PPRC-XD initial copy. When all volumes of the session have reached this state, the Asynchronous PPRC process starts creating Consistency Groups. If new volumes join the session *after* the STARTASYNCCOPY command, the creation of Consistency Groups continues and the new volumes join the session when they have completed the first pass. This is an LSS related command. See the syntax of the command and refer to Example 4-29, which shows adding three volumes per LSS to the session.

***Syntax:***

**PPRCOPY POPULATESESSION**

```
DDNAME(dname)|SYSNAME(sysxxx)|UNITADDRESS(ccuu)
IVOLLIST(volume list) | RVOLLIST(volume list)
JOIN|REMOVE SESSIONNO(session number)
VOLCOUNT(volcount)
Optional Parameters
RANGE(YES|NO)
```

*Example 4-29 Populating the Asynchronous PPRC session with volumes*

```
PPRC    POPULATESESSION            -
          DDNAME(VLB000)           -
          SESSIONNO(50)            -
          VOLCOUNT(1)              -
          RANGE(YES)               -
```

```
         RVOLLIST(X'00',X'02')      -
         JOIN
PPRC    POPULATESESSION           -
         DDNAME(VLB200)             -
         SESSIONNO(50)              -
         VOLCOUNT(1)                -
         RANGE(YES)                 -
         RVOLLIST(X'00',X'02')      -
         JOIN
```

The progress of volumes joining a session can be monitored by the QUERY SESSIONDEVICES command, that shows the status of every volume in the LSS that has been queried. The number in the VOLCOUNT parameter specifies the number of Ranges in RVOLLIST here. When using IVOLLIST the VOLCOUNT number would specify the number of individual volumes.

### Start and modify an Asynchronous PPRC Session

The third step starts the Asynchronous PPRC Session. This command can be submitted to any of the previously defined primary LSSs. The ESS that contains this LSS becomes the Master ESS to this session. The Master is a function inside an ESS box that belongs to a session and communicates to Subordinate ESSs via Fiber Channel paths, to control the creation of the Consistency Groups and manage the session.

**Note:** ICKDSF is architected to supports 16 Subordinates. Currently only one Master and up to seven Subordinate ESSs are supported for use. If you use ICKDSF *and* the Web User Interface, you should be aware of the following restriction: The ESS Specialist is only able to manage a total of eight ESSs, including primary and secondary ESSs in a single Copy Services Domain.

PPRC Paths between the Master ESS and all Subordinate ESSs have to be established prior to the start of the session. The paths can be established between any LSS in the Master and Subordinate LSS.

This command creates a topology including the Master and all Subordinates and its volumes as well as the three previously described timers. See 3.9.2, “Asynchronous PPRC - How it works” on page 118. The *Maximum Coordination Interval Timer*, *Maximum Consistency Group Drain Time*, and *Consistency Group Interval Timer* values can be changed using the MODIFY parameter of this command if the session is already started. The session number and the ESS relationships in the session are unchangeable as long as the session is started. Before changing the timers the session has to be paused by the TERMASYNCCOPY PAUSE. You can dynamically add LSSs and volumes to the session that belongs to the ESSs, that are already in the Asynchronous PPRC session. If you need to include a new ESS you have to first Terminate the session.

All subsequent STARTASYNCCOPY commands should be issued to this LSS as long as you did not terminate the session with the TERMASYNCOPY command. This is a Session related command. See the syntax of the command and refer to Example 4-30.

#### *Syntax:*

**PPRC STARTASYNCCOPY**

```
DDNAME(dname)|SYSNAME(sysxxx)|UNITADDRESS(ccuu)
START | MODIFY
SESSIONNO(session number)
```

```
Optional Parameters
CGINTERVALTIME(seconds)
MAXCOORDTIME(milliseconds)
MAXDRAINTIME(seconds)
TOPOLOGY(msc1,msc2...msc16)
```

*Example 4-30 Starting the Asynchronous PPRC session*

```
  PPRCOPY STARTASYNCCOPY             -
          DDNAME(VLB000)             -
          SESSIONNO(50)              -
          MAXCOORDTIME(50)           -
          MAXDRAINTIME(30)          -
          CGINTERVALTIME(10)         -
          START
```

The status of the session can be monitored by the QUERY ASYNC command.

## Query the status of Asynchronous PPRC Session

Three PPRC and two FlashCopy queries are available to monitor the session and the state of the components. They can periodically be issued and used for automation purposes, providing statistics on the behavior of the Asynchronous PPRC process.

### *Determine the status of Asynchronous PPRC*

When a session is started, it creates Consistency Groups when all of the included volumes have joined the session. This query provides information such as status of the session, the topology, who is the Master and Subordinate, how many Consistency Groups have been created and how many failed and what was the reason. If z/OS LSSs as well as Open Systems LSSs are members of the session you can see the session information that is valid for both of them in the report. See the syntax of the command and refer to Example 4-31.

This is a session level query.

### *Syntax:*

**PPRC QUERY ASYNCCOPY**

DDNAME(*dname*)|SYSNAME(*sysxxx*)|UNITADDRESS(*ccuu*)
Optional Parameters
REMOTEVOL(*lss,cca,ssid,ser#*)

*Example 4-31 Querying the status of Asynchronous PPRC*

```
PPRCOPY QUERY                  -
        DDNAME(VLB000)          -
        ASYNCCOPY

ASYNCHRONOUS PPRC CONFIGURATION INFORMATION
MASTER CU SERIAL NUMBER/LSS              :  23953/00

NUMBER OF MASTERS IN THIS ESS BOX         :  1
NUMBER OF OTHER MASTERS TO THIS SUBORDINATE:  0
SESSION INFORMATION
CURRENT TIME = 2004.04.27.20:23:57
+------------------------------------------------------------------------------------------------------------------+
!    !     !       !       !       !          ! TOTAL #  !      !           ! TOTAL #  !                           !
!    !     !       !       !       !          ! GOOD CGS !      ! # FAILED  ! UNSUCCFL !                           !
!    !     !       ! MAX   ! MAX   !          !  FORMED  !      ! TRIES TO  ! TRIES TO !  THIS MASTER TO OTHER     !
!    !     !  CG   ! COORD ! CG    !  TIME    !  SINCE   ! %AGE ! FORM CG   ! FORM CG  !   SUBORDINATES DATA       !
!    !     ! INTVL ! INTVL ! DRAIN !  LAST    ! LAST IML ! GOOD ! SINCE     !  SINCE   !---------------------------!
!SESS!     ! TIME  ! TIME  ! TIME  ! GOOD CG  !    OR    ! CG   ! LAST GOOD ! LAST IML !MSTR/LSS!SUB/LSS!  SUBORD  !
! NO !STATE! (SEC) ! (MSEC)! (SEC) ! FORMED   ! FAILOVER ! FRMD ! CG FORMED !OR FAILOVR!SSID    !SSID   !   CUSN   !
!----+-----+-------+-------+-------+----------+----------+------+-----------+----------+--------+-------+----------!
! 50 !RUN  !   10  !   50  !  240  ! 04/27/04 !    108   !  97  !    00     !     3    ! NO SUBORDINATES DEFINED   !
!    !     !       !       !       ! 20:20:02 !          !      !           !          !                           !
+------------------------------------------------------------------------------------------------------------------+
```

```
LEGEND
  SESS NO
    UA   : UNABLE TO RETRIEVE INFORMATION, DATA STRUCTURES UNAVAILABLE

  STATE
ICKDSF - MVS/ESA    DEVICE SUPPORT FACILITIES 17.0              TIME: 16:24:59       04/27/04    PAGE   3
    RUN  : SESSION IS RUNNING
    PAUSE: SESSION IS PAUSED
    P PND: SESSION IS PAUSE PENDING
    FTL01: REVERT FLASHCOPY FAILED
    FTL02: ASYNC PPRC STRUCTURES INACCESSIBLE OR IN FAILED STATE
    FTL03: ASYNC PPRC CONSISTENCY CHECK OF SEQUENCE NUMBERS FAILED
    FTL04: ASYNC PPRC INVALID SESSION ID DETECTED
    FTL05: CONSISTENCY GROUP CORRUPTED
    FTL06: TIMEOUT DURING CONSISTENCY CHECK
    FTL07: COMMUNICATION FAILURE DURING CONSISTENCY CHECK
    FTL08: UNABLE TO COMPLETE CONSISTENCY CHECK
    FTL09: BUSY CONDITION PREVENTS CONSISTENCY CHECK
    FTL0A: REVERT FLASHCOPY FAILED DUE TO TIMEOUT

  TIME LAST GOOD CG FORMED, NUMBER FAILED TRIES TO FORM CG, %AGE TRIES FORMED GOOD CG
    HH:MM:SS: INDICATED TIME
    00      : CONSISTENCY GROUP NEVER FORMED (TIME LAST GOOD CG FORMED, ONLY)
    FL      : INFORMATION UNOBTAINABLE DUE TO FAILOVER OR IML OF ESS

CONSISTENCY GROUP FAILURES REPORT
+-----------------------------------------------------+
!      !         !            !     !        ! MASTER !
! SESS !  WHICH  !            !     ! ERROR  ! STATE  !
!  NO  ! FAILURE !   CUSN     ! LSS ! REASON ! CODE   !
!------+---------+------------+-----+--------+--------!
!  50  !  FIRST  !   23953    !  02 !    FC9 !    04  !
!      !---------+------------+-----+--------+--------!
!      !  PREV   !   23953    !FO/IM!     4  !    06  !
!      !---------+------------+-----+--------+--------!
!      !  M RCNT !   23953    !  10 !    FD1 !    04  !
+-----------------------------------------------------+
LEGEND
  SESS NO
    UA   : UNABLE TO RETRIEVE INFORMATION, DATA STRUCTURES UNAVAILABLE
  WHICH FAILURE
    FIRST : FIRST CONSISTENCY GROUP FAILURE
    PREV  : PREVIOUS CONSISTENCY GROUP FAILURE
    M RCNT: MOST RECENT CONSISTENCY GROUP FAILURE

  LSS, ERROR REASON, OR MASTER STATE CODE
    FO/IM: INFORMATION UNAVAILABLE DUE TO FAILOVER OR IML

  ERROR REASON
    01  : ASYNCHRONOUS PPRC STRUCTURES CANNNOT BE ACCESSED
    02  : ASYNCHRONOUS PPRC COMMUNICATION PATH FAILURE
    03  : EXTENDED DISTANCE CONSISTENCY SESSION MEMBERS NOT IN CORRECT STATE
    04  : MAXIMUM CONSISTENCY GROUP DRAIN TIME EXCEEDED
    400 : INVALID PARAMETER
    FC9 :SESSION ALREADY OPEN ON LSS OR MAXIMUM SESSIONS ALLOWED ALREADY METYY
    FC9 : SESSION ALREADY OPEN ON LSS OR MAXIMUM SESSIONS ALLOWED ALREADY MET
    FD1 : START REQUESTED AND ANOTHER MASTER IS ALREADY RUNNING OR MODIFY REQUESTED AND PARAMETERS DO
NOT MATCH EXISTING SESSION OR COMMAND ISSUED TO OTHER THAN ALLOWABLE LSS
    7ZZZ: MICROCODE LOGIC ERROR - ZZZ DESCRIBES ERROR
    F005: TEMPORARILY UNAVAILABLE
    F01D: LONG BUSY
 MASTER STATE
    01: PAUSE/TERMINATE ASYNCHRONOUS PPRC IN PROGRESS
    02: START/RESUME ASYNCHRONOUS PPRC IN PROGRESS
    03: ASYNCHRONOUS PPRC IS BETWEEN CONSISTENCY GROUP FORMATIONS
    04: XDC START INCREMENT IN PROGRESS
    05: XDC RUN IN PROGRESS
    06: DRAIN IN PROGRESS
    07: FLASHCOPY ESTABLISH WITH REVERTIBLE IN PROGRESS
    08: FLASHCOPY WITHDRAW WITH COMMIT IN PROGRESS
    09: XDC INCREMENT COMPLETE IN PROGRESS
    0A: FLASHCOPY WITHDRAW WITH REVERT IN PROGRESS
    0B: ASYNCHRONOUS PPRC IS PAUSED
    0C: ASYNCHRONOUS PPRC IS PERFORMING POST-CONSISTENCY GROUP TASKS
    0D: ASYNCHRONOUS PPRC IS FATAL
    0E: ASYNCHRONOUS PPRC IS COMPLETING ERROR RECOVERY

OTHER MASTERS TO THIS SUBORDINATE ASSOCIATIONS
 THERE ARE NO OTHER MASTERS TO THIS SUBORDINATE
```

### *Query the status of the volumes in the session*

This query provides information on the queried LSS about the number of volumes that are part of a session or are still join pending. In the example below, the session 50 is in normal state and the first volume in LSS02 with CCA=00 is already a member of the session (IS) but the remaining volumes are still join pending (JP). The reason in this case is that the PPRC-XD first copy pass of these volumes is still not finished. Refer to Example 4-32.

This is an LSS level query.

### *Syntax:*

**PPRC QUERY SESSIONSDEVICES**

```
PPRC QUERY SESSDEV
DDNAME(dname)|SYSNAME(sysxxx)|UNITADDRESS(ccuu)
```

*Example 4-32 Query Session devices*

```
PPRCOPY QUERY            -
        DDNAME(VLB200)   -
        SESSDEV
ICK00700I DEVICE INFORMATION FOR B200 IS CURRENTLY AS FOLLOWS:
        PHYSICAL DEVICE = 3390
        STORAGE CONTROLLER = 2105
        STORAGE CONTROL DESCRIPTOR = E8
        DEVICE DESCRIPTOR = 0A
        ADDITIONAL DEVICE INFORMATION = 4A000035

ICK04030I DEVICE IS A PEER TO PEER REMOTE COPY VOLUME

EXTENDED DISTANCE CONSISTENCY SESSIONS AND DEVICES TABLE
+-----------------------------------------------------------------------------+
! SESS ! SESS !                                                               !
!  NO  ! STAT ! VOLUMES IN SESSION                                            !
!------+------+---------------------------------------------------------------!
!  50  ! N    !  02.00(IS,DP)   02.01(JP,DP)   02.02(JP,DP)                   !
+-----------------------------------------------------------------------------+
LEGEND
 SESSION STATUS
  CGIP = CONSISTENCY GROUP IN PROGRESS
  INAC = NO INFORMATION, DATA STRUCTURES INACCESSIBLE
  IP   = INCREMENT PENDING
  NAV  = NO ACTIVE VOLUMES IN SESSION
  N    = NORMAL
  NSES = NO SESSIONS DEFINED ON ESS

 VOLUME STATUS (1ST ENTRY IN PARENTHESES)
  IS   = IN EXTENDED DISTANCE CONSISTENCY GROUP SESSION
  JP   = EXTENDED DISTANCE CONSISTENCY VOLUME IS JOIN PENDING
  RP   = EXTENDED DISTANCE CONSISTENCY VOLUME IS REMOVE PENDING
  1P   = PPRC PAIR IS IN FIRST PASS OF INTIAL COPY

 VOLUME STATE (2ND ENTRY IN PARENTHESES)
  SX = PAIR IS SIMPLEX
  DP = PAIR IS DUPLEX PENDING
  FD = PAIR IS FULL DUPLEX
  SP = PAIR IS SUSPENDED
```

### Query the number of volumes that are out-of-sync

The PPRC Query OUTOFSYNCSTATE returns information about the number of volumes in the specified session that are out-of-sync due to the initial PPRC-XD or cyclic synchronization process. The more volumes at the primary site that have a high workload, the more volumes will be out-of-sync. With quiesced workloads on all primary volumes, the number of out-of-sync volumes become zero in time, so you can get an indication of how the application workload is distributed across your primary volumes. Refer to Example 4-33 for an LSS level query.

This is an LSS or ESS level query.

### Syntax:

**PPRC QUERY OUTOFSYNCSTATE**

```
DDNAME(dname)|SYSNAME(sysxxx)|UNITADDRESS(ccuu)
SESSIONNO(session number)

Optional Parameters
ESSLEVEL(NO|YES)
```

*Example 4-33 Query the number of out-of-sync volumes*

```
PPRCOPY QUERY OOSSTATE            -
        DDNAME(VLB000)             -
        SESSIONNO(50)
ICK00700I DEVICE INFORMATION FOR B000 IS CURRENTLY AS FOLLOWS:
        PHYSICAL DEVICE = 3390
        STORAGE CONTROLLER = 2105
        STORAGE CONTROL DESCRIPTOR = E8
        DEVICE DESCRIPTOR = 0A
        ADDITIONAL DEVICE INFORMATION = 4A000035
ICK04030I DEVICE IS A PEER TO PEER REMOTE COPY VOLUME

ASYNCHRONOUS PPRC OUT-OF-SYNC INFORMATION AT LSS LEVEL
 3    VOLUMES IN SESSION 50  ARE OUT OF SYNC
```

To have a look at single volumes in detail, the already existing query command can be used. Refer to Example 4-34.

*Example 4-34 PPRCOPY QUERY*

```
QUERY REMOTE COPY - VOLUME

                                                (PRIMARY)    (SECONDARY)
                                                SSID    CCA  SSID    CCA
DEVICE  LEVEL      STATE            PATH STATUS SER #   LSS  SER #   LSS
------  ---------  --------------   ----------- -----------  -----------
B000    PRIMARY    PENDING    XD    ACTIVE      1340    00   1070    00
                                                23953   00   24663   00
```

### Query the status of the FlashCopy volumes in case of a disaster or a failure

Use the two commands described next, to determine the status of the FlashCopy relations when Asynchronous PPRC fails in case of a disaster or a planned failover and the state of the FlashCopy volumes could be inconsistent.

The query output provides an indication as to whether the FLASHCOPY WITHDRAW command to the FlashCopy primary volumes at the remote site should be issued or not, and if *Yes*, which of the parameters REVERT or COMMIT must be used with that command:

- Commit means that all FlashCopy volumes are set to the latest Consistency Group.
- Revert means that all FlashCopy volumes are reset to the previous Consistency Group.

The decision depends on the state of the FlashCopy relations of each volume pair and if the Consistency Group number is equal for each pair or not. But usually none of these Withdraw commands have to be applied because the FlashCopy volumes are in a consistent state.

It is intended that Multiple Device Manager will provide support for Asynchronous PPRC in the future and help automate this process.

> **Note:** These Withdraw commands with either REVERT or COMMIT do *not* withdraw the FlashCopy relation. They only direct all the FlashCopy volume pairs to the same FlashCopy relationship.

### *Query for the Increment Status if the Master ESS and links are available*

This query can be only addressed to the Master or Subordinate ESSs. It should be used for debugging purposes only. In case of a disaster, the primary site usually is not available and this command can not be used. In that case the QUERY RELATIONS command should be used to identify the status of the FlashCopies at the remote site.

The example below shows that the FlashCopy Withdraw is not required in this case. Refer to Example 4-35.

### *Syntax:*

**FLASHCPY QUERY INCREMENTSTATUS**

```
DDNAME(dname)|SYSNAME(sysxxx)|UNITADDRESS(ccuu)
SESSIONNO(session number)
Optional Parameters: None
```

*Example 4-35   FlashCopy Query IncrementStatus*

```
FLASHCPY QUERY INCRSTAT          -
         DDNAME(VLB000)          -
         SESSIONNO(50)
ICK00700I DEVICE INFORMATION FOR B000 IS CURRENTLY AS FOLLOWS:
         PHYSICAL DEVICE = 3390
STORAGE CONTROLLER = 2105
STORAGE CONTROL DESCRIPTOR = E8
DEVICE DESCRIPTOR = 0A
         ADDITIONAL DEVICE INFORMATION = 4A000035
ICK04029I DEVICE IS IN SUSPENDED PPRC STATE

FLASHCOPY ASYNCHRONOUS PPRC INCREMENT STATUS INFORMATION
 SESSION NUMBER: 50
 SESSION IS NON-REVERTIBLE AND EQUAL (NO WITHDRAW REQUIRED)
```

### *Use QUERY RELATIONS if the Master ESS and links are not available*

If the Master is no longer available and/or the PPRC paths to the secondary volumes do not exist anymore, this Query Relations is required. See Example 4-36.

This query has to be submitted to each PPRC secondary volume to obtain the FlashCopy Sequence Number and the information, as to whether the volume is revertible or not. If you summarize all the volume information and select the corresponding indication as shown in Table 4-3, you can see the appropriate action in the table. This information should be collected by automation tools.

*Table 4-3 Indication of revertible or non-revertible*

| Indication | Specification on FlashCopy Withdraw Command |
|---|---|
| Revertible or Non-Revertible Volumes and Equal Sequence Numbers | Commit |
| Non-Revertible Volumes and Equal Sequence Numbers | No Withdrawal required |
| Revertible and Non-Revertible Volumes | Revert |
| All Volumes Revertible | Revert |
| Consistency Group Corrupted | N/A |

This is a volume level query.

### *Syntax:*

**FLASHCPY QUERY RELATIONS**

```
Required Parameters
DDNAME(dname)|SYSNAME(sysxxx)|UNITADDRESS(ccuu)
Optional Parameters: None
```

*Example 4-36 Query FlashCopy Relations*

```
FLASHCPY QUERY RELATIONS          -
         UNITADDRESS(3000)
ICK00700I DEVICE INFORMATION FOR 3000 IS CURRENTLY AS FOLLOWS:
         PHYSICAL DEVICE = 3390
         STORAGE CONTROLLER = 2105
         STORAGE CONTROL DESCRIPTOR = E8
         DEVICE DESCRIPTOR = 0A
         ADDITIONAL DEVICE INFORMATION = 4A000035
ICK04032I DEVICE SPECIFIED IS A PPRC SECONDARY
FLASHCOPY RELATIONS INFORMATION TABLE
SOURCE CU SERIAL NUMBER: 0000024663
+-------------------------------------------------------------------+
! FLASHCOPY  !     !   !    !     !     !     !     SOURCE/TARGET     !
! SEQUENCE   !     !   !    !     !     !     !-----------------------!
! NUMBER     ! R/T ! R ! FV ! BCE ! BCP ! CRA ! VR ! SSID ! LSS ! CCA !
!-------------------------------------------------------------------!
! 1083097742 ! SRC ! F !  T !  F  !  F  !  T  ! F  ! 1071 ! 01  ! 00  !
+-------------------------------------------------------------------+
LEGEND
 R/T = FLASHCOPY RELATIONSHIP TYPE (SRC = SOURCE, TGT = TARGET)
 R   = REVERTIBLE (TRUE ! FALSE)
 FV  = FULL VOLUME RELATION (TRUE ! FALSE)
 BCE = BACKGROUND COPY ENABLED (TRUE ! FALSE)
 BCP = BACKGROUND COPY IN PROGRESS (TRUE ! FALSE)
 CRA = CHANGE RECORDING ACTIVE (TRUE ! FALSE)
 VR  = VALIDATION REQUIRED (T!F), REPORTED INFORMATION NOT CURRENT AND MAY NOT BE ACCURATE
```

### *Terminate Asynchronous PPRC*

The session can be terminated or paused. A paused session completes the last Consistency Group and keeps the configuration of the Master and Subordinate relations. It can be restarted with the STARTASYNC command. If you terminate the session with TERMINATE the creation of Consistency Groups stops immediately, refer to Example 4-37.

This is a session level command.

***Syntax:***

```
PPRCOPY TERMASYNCCOPY

DDNAME(dname)|SYSNAME(sysxxx)|UNITADDRESS(ccuu)
PAUSE | TERMINATE
SESSIONNO(session number)
TOPOLOGY(msc1,msc2...msc16)

Optional Parameters
MASTER | SUBORDINATE
FORCE
```

*Example 4-37 Terminate FlashCopy*

```
PPRCOPY TERMASYNCCOPY          -
          DDNAME(VLB000)         -
          SESSIONNO(50)          -
          TERMINATE
```

# 4.4 ANTRQST macro

The ANTRQST macro provides an application program call to the OS/390 and z/OS system data mover (SDM) application programming interface (API). This macro allows you to call XRC, PPRC, FlashCopy, and SnapShot copy functions. For PPRC requests the ANTAS000 address space must be active at the time of macro invocation. You have to ensure that the necessary address spaces, such as ANTAS000, ANTAS001, or ANTMAIN are active when you execute the macro.

For a detailed description of this macro the following publication can be referenced: *z/OS DFSMSdfp Advanced Services,* SC35-0423.

Detailed information on ANTRQST API can be found in the publication: *z/OS DFSMSdfp Advanced Services,* SC26-7400.

For example, this Application Programming Interface (API) is used by GDPS to manage PPRC. The ANTRQST API services for ILK=PPRC have been updated to provide FCP support. The changes affect the PESTPATH, PDELPATH, and PQUERY requests.

The PESTPATH and PDELPATH requests have two new parameters, PWWNN to identify the primary subsystem WWNN and SWWNN to identify the secondary subsystem WWNN. The presence of PWWNN and SWWNN in a PESTPATH or PDELPATH request signifies that FCP paths are being managed. ESCON paths continue to be managed using the PSERIAL and SSERIAL parameters.

The PQUERY output has been updated to include the primary and secondary WWNNs.

# 4.5 ESS API

The ESS API supports routine LUN management activities, such as LUN creation, mapping, and masking. Starting with ESS LIC level 2.3.0, the ESS API is enhanced to support ESS Copy Services configuration and use for PPRC and FlashCopy. It supports these activities through the use of the interface as defined by the Storage Networking Industry Association (SNIA) Storage Management Initiative Specification (SMI-S).

The ESS API is implemented through the IBM TotalStorage Common Information Model Agent (CIM Agent) for the ESS, a middleware application designed to provide a CIM-compliant interface. The CIM-compliant interface allows Tivoli® and third-party software management tools to discover, monitor, and control the ESS. These enhancements demonstrate IBM's continued support for CIM and Web-Based Enterprise Management (WBEM) technologies as the common interface for the discovery and management of resources in a multi-vendor storage area network.

The ESS API and CIM Agent are provided with the ESS at no additional charge. The CIM Agent is available for the AIX, Linux, and Windows 2000 operating system environments.

The ESS API is discussed in more detail in Appendix A, “ESS Application Programming Interface (API)” on page 351.

## 4.6 Multiple Device Manager Replication Manager

Data replication is the core function required for data protection and Disaster Recovery. It provides advanced copy services functions for supported storage subsystems on the SAN.

Replication Manager administers and configures the copy services functions and monitors the replication actions. Multiple Device Manager Replication Manager V1.1 capabilities consist of the management of two types of copy services: the Continuous Copy (also known as Peer-to-Peer Remote Copy, PPRC, or Remote Copy), and the Point-in-Time Copy (also known as FlashCopy®). The Asynchronous PPRC announcement included a preview statement for Multiple Device Manager Replication Manager to support Asynchronous PPRC,

Replication Manager includes support for replica sessions, which ensures that data on multiple related heterogeneous volumes is kept consistent, provided that the underlying hardware supports the necessary copy services operations. Multiple pairs are handled as a consistent unit, Freeze-and-Go functions can be performed when errors in mirroring occur. Replication Manager is designed to control and monitor the copy services operations in large-scale client environments.

Replication Manager is implemented by applying predefined policies to Groups and Pools, which are groupings of LUNs that are managed by the Replication Manager. It provides the ability to copy a Group to a Pool, in which case it creates valid mappings for source and target volumes and optionally presents them to the user for verification that the mapping is acceptable. In this case, it manages Pool membership by removing target volumes from the pool when they are used, and by returning them to the pool only if the target is specified as being discarded when it is deleted. A follow-on version of Replication Manager may provide the ability to copy a Group to another Group, in which case the element ordering within the groups is used to determine the pairings between source and target volumes.

Multiple Device Manager Replication Manager V1.1 supports FlashCopy and Synchronous PPRC. Future releases will add other copy services functionality. The managed storage prerequisites include:

- ESS at LIC 2.3.1
- CIM Agent for ESS 1.2.0.4
- ESS CLI at 2.3.0.13

The ESS Copy Services supported with Multiple Device Manager Replication Manager V1.1 include:

- ESS PPRC:
  - Synchronous remote copy
  - Add / delete volume pairs
  - Full background copy
  - Freeze / Run
  - Suspend / resume
  - Query status of the session, paths, and pairs
- ESS FlashCopy:
  - Full background copy
  - Source / target within an ESS

For much more detailed information on Multiple Device Manager and Replication Manager for use with ESS Copy Services refer to the publications:

- *IBM TotalStorage SAN Multiple Device Manager Command-Line Interface Guide*, SC26-7585
- *IBM TotalStorage SAN Multiple Device Manager Configuration Guide*, SC26-7586
- *IBM TotalStorage SAN Multiple Device Manager CIM Agent Developer's Reference*, SC26-7587
- *IBM TotalStorage Multiple Device Manager Usage Guide*, SG24-7097

Refer to the web site for up-to-date information and requirements for the Multiple Device Manager Director server:

`http://www.ibm.com/servers/storage/support/virtual/mdm.html`

## 4.7 Asynchronous PPRC Utilities for Open System Environment

The Asynchronous PPRC Utilities for Open System Environment are designed to provide commands for the single point setup and management of Asynchronous PPRC, where automation is needed for a large number of volumes. It is designed to complement existing interfaces, such as ICKDSF, the CLI, Web User Interface, and the Multiple Device Manager Replication Manager. The tool simplifies the setup, management, and recovery of Asynchronous PPRC implementations. The commands provided can easily be scripted to integrate with a client's environment.

It is distributed on a separate CD with ESSs at LIC 2.4.0 or higher. On that CD, there is a README file for installation of the commands, as well as a *User's Guide*. There are two components: the server and the user client.

The tool will run on a dedicated Windows server that meets the following requirements:

- Intel Pentium® 4 or compatible host server
- Microsoft Windows 2000 Advanced Server with SP3
- Minimum memory of 1 GB
- Minimum disk space of 40 GB
- One of the following Web Server products:
  - WebSphere® Application Server 5.1 or compatible Web Server, or

– Apache Tomcat 5.0, which has been fully tested and is supported.

The client workstation requirements are:

- Microsoft Windows 2000 or Windows XP
- Java 2 Runtime Environment Standard Edition (build 1.3.1)

The server receives requests from the client and communicates over TCP/IP to the ESS for the execution of the commands. The initial set of commands will include:

- Create, modify, start, stop, and resume an Asynchronous PPRC session.
- Manage Failover and Failback operations including managing consistency.
- Perform planned outages.

To monitor the ESS volume status and the Asynchronous PPRC session status, you will use the appropriate ESS Web User Interface or the CLI commands.

The commands rely on configuration files that are created by the user. Future versions of the tool will complete and enhance these functions. It will exist as a stand-alone set of commands and in the future will also be integrated into the functionality of the Multiple Device Manager Replication Manager.

# 4.8 Planning for PPRC

This section includes many tools and suggestions for your planning for PPRC.

## 4.8.1 Asynchronous PPRC Bandwidth Estimator

The Asynchronous PPRC Bandwidth Estimator is used by IBM ITS to support clients in the planning and sizing needed for Asynchronous PPRC. The customer requirements have to be analyzed to obtain the input data for the tool. This input data would include the required data currency, the distance to the remote location, the number of ESSs and HBAs, the compressibility of the data, and the current workload characteristics of all the Open Systems and z/OS data that should be part of the Asynchronous PPRC session.

The tool calculates the expected bandwidth required for the solution and provides a recommendation for the number of required remote links for several available link types, such as OC-3, OC-12 T3, and many others.

Please contact your IBM representative for this bandwidth analysis.

## 4.8.2 Defining TSO commands to RACF

In the following sections we discuss TSO commands in relation to RACF protection.

### Security

PPRC commands are extremely powerful, so it is important that they are used correctly and directed to the correct devices. It is recommended that the access to these commands be restricted by either putting these commands in a RACF protected library, or defining resource profiles in the RACF FACILITY class, and restrict access to those profiles.

### *Placing PPRC commands in a RACF-protected library*

TSO commands can be placed in a RACF-protected library to restrict PPRC TSO commands to authorized storage administrators.

To RACF-protect PPRC commands, the following steps should be performed:

1. Issue the following RDEFINE command for each PPRC command, and for each command abbreviation that you want defined to RACF:

   `RDEFINE PROGRAM` *cmdname* `ADDMEM('SYS1.CMDLIB')/`*volser*`/NOPADCHK) UACC(NONE)`

   The following terms apply to the example above:

   **cmdname** Defines the PPRC TSO command name or an abbreviation of a command. Issue a separate RDEFINE command for each command, and any command abbreviations you plan to use. RACF can only perform checking on commands and abbreviations that are defined to it.

   **volser** Defines the name of the volume that contains the SYS1.CMDLIB data set

2. Issue the PERMIT command for all commands, and authorized PPRC TSO command users as follows:

   `PERMIT` *cmdname* `CLASS(PROGRAM) ID(`*name*`) ACCESS(READ)`

   The following terms apply to the above example:

   **cmdname** Defines the PPRC TSO command or an abbreviation of a command.

   **name** Defines the user ID that will receive RACF access authority to that command.

3. Issue the following SETROPTS command from a user ID that has the appropriate authority:

   `SETROPTS CLASSACT(PROGRAM) WHEN(PROGRAM) REFRESH`

### *Defining resource profiles in the RACF FACILITY class*

You can limit the use of PPRC commands by defining resource profiles in the RACF FACILITY class, and restricting access to those profiles. To use a protected command, you need read-access authority to the applicable profile.

Table 4-4 lists the PPRC commands and the FACILITY class profiles that can restrict them. Refer to the z/OS Security Server publications of your current RACF release for details on activating the RACF FACILITY class, and defining and authorizing users to the PPRC command profiles.

*Table 4-4 PPRC FACILITY class profile names*

| Command | Profile name |
|---|---|
| CDELPAIR | **STGADMIN.ANT.PPRC.COMMANDS** |
| CDELPATH | |
| CESTPAIR | |
| CESTPATH | |
| CGROUP | |
| CQUERY | |
| CRECOVER | |
| CSUSPEND | |

| Command | Profile name |
|---|---|
| CQUERY | **STGADMIN.ANT.PPRC.CQUERY** |

You can authorize the CQUERY command utilization with either of these profiles: `STGADMIN.ANT.PPRC.COMMANDS or STGADMIN.ANT.PPRC.CQUERY.`

PPRC first checks `STGADMIN.ANT.PPRC.COMMANDS` for authorization. If authorization is not permitted with the `STGADMIN.ANT.PPRC.COMMANDS` profile, then PPRC checks the `STGADMIN.ANT.PPRC.CQUERY` profile for authorization to issue the CQUERY command.

The following examples activate the RACF FACILITY class, define the profile for the PPRC commands, and give user STGADMIN authority to use this profile.

1. Activate the RACF Facility class:

```
SETROPTS CLASSACT(FACILITY)
```

2. Define the profile for PPRC commands, and authorize user STGADMIN to use this profile:

```
RDEFINE FACILITY STGADMIN.ANT.PPRC.COMMANDS UACC(NONE)
PERMIT STGADMIN.ANT.PPRC.COMMANDS CLASS(FACILITY) ID(STGADMIN) ACCESS(READ)
```

#### *IKJTSOxx*

Authorize PPRC TSO commands by adding the command names to the AUTHCMD PARM parameter of the IKJTSOxx member of SYS1.PARMLIB. After you have added the PPRC command names to the IKJTSOxx member, issue the TSO command PARMLIB UPDATE(xx) to activate the new IKJTSOxx member.

### 4.8.3 Considerations for z/VM and VSE/ESA

PPRC control and management in the VM environments is done by means of the ESS Copy Services Web User Interface (WUI). The CLI can also be used if a supported server is connected. Also, the ICKDSF utility can be used to invoke PPRC functions in the VM environment.

### 4.8.4 z/VM

When PPRC is implemented in the VM environment, the following considerations apply:

- All PPRC volumes must be either dedicated volumes or defined as full pack mini disks (including DEVNO-defined mini disks).
- The VM guest directory must include an entry stating STDEVOPT DASDSYS DATAMOVER:
  - The STDEVOPT statement specifies the optional storage device management functions available to a virtual machine.
  - The DASDSYS operand tells the control program (CP) whether the virtual machine is authorized to control and process Concurrent Copy and PPRC established pair channel control words (CCWs).
  - The DATAMOVER option tells the CP whether the virtual machine is authorized to issue Concurrent Copy and PPRC establish pair CCWs.

## 4.8.5 Configuring for VM

The configuration considerations for using PPRC in a VM environment are similar to the z/OS general considerations discussed in 3.10.1, "PPRC connectivity using ESCON ports" on page 126.

In managing the ESCON configurations, remember that the ESCON Manager is not available for VM. Make sure that your documentation shows the latest ESCON connections, and any change is reflected immediately.

ESCON Manager is not specifically designed to operate on an MVS guest in a VM environment. For special configuration guidelines and operating procedures when working with ESCON Manager on an MVS guest in a VM environment, refer to the IBM Redbook: *Planning for ESCON Manager,* GC23-0423.

Real addresses are used as values for addresses in the ICKDSF commands. It is recommended to keep the VM virtual device number the same as the real device number for the devices established in PPRC pairs; this facilitates entering the commands correctly. If you are not able to keep the VM virtual device number the same as the real device number, make sure to accurately document the mapping.

### *Operation under VM*

In the VM environment, PPRC can be managed using the ESS Copy Services WUI. This allows the setup of all the necessary PPRC functions and definitions.

Alternatively, the ICKDSF utility can be used. If ICKDSF is used, then the PPRCOPY command provides the means to initiate and terminate the functions, suspend a copy operation, query the status of volumes, and enable a recovery host to gain control of the secondary volumes.

The ICKDSF command CONTROL CONFIGURE(DISPLAY) can be used to get the SSIDs, serial numbers, CCAs, and LSS numbers that are going to be used in the PPRCOPY commands.

The PPRCOPY QUERY command can be used with or without the PATHS parameter to get information about the status of the PPRC pairs. The output from the PPRCOPY QUERY command is returned to the ICKDSF user.

The following publication provides detailed descriptions of the PPRCOPY commands: *Device Support Facilities User's Guide and Reference,* GC35-0033.

## 4.8.6 VSE/ESA

In the VSE environment, PPRC can be managed using the ESS Copy Services Web user (WUI) interface. This allows the setup of all the necessary PPRC functions and definitions with no limitations.

As in VM/ESA, the ICKDSF utility can also be used to execute PPRC commands in VSE/ESA. If ICKDSF is used, then the PPRCOPY commands are used.

The configuration considerations for using PPRC in a VSE environment are similar to the z/OS general considerations discussed in 3.10.1, "PPRC connectivity using ESCON ports" on page 126.

In managing the ESCON configurations, remember that the ESCON Manager is not available for VSE/ESA. Make sure that your documentation reflects the latest ESCON connections, and any changes must be updated immediately.

## 4.8.7 Software information resources

The software support information varies with some frequency, so it is highly recommended that the latest support information be retrieved from the Web sites where it is posted and updated.

Support information for the ESS can be found at the URL:

`http://www.storage.ibm.com/hardsoft/products/ess/supserver.htm`

### *z/OS and OS/390*

For support information when implementing PPRC in the z/OS environment, the DFSMS SDM Copy Services page should be referred to:

`http://www.storage.ibm.com/software/sms/sdm/index.html`

From the previous page, the technical update page can be reached for the latest APARs and for downloading the required publications. The technical update page is located at:

`http://www.storage.ibm.com/software/sms/sdm/sdmtech.html`

The PSP Upgrade 2105DEVICE should be reviewed for the latest ESS maintenance.

Regarding the publications, remember that they can be found on the Internet. To find z/OS books on the Internet, go to the z/OS home page at:

`http://www.ibm.com/servers/eserver/zseries/zos`

Then from this Web site, you can link directly to the z/OS softcopy books by selecting the **Library** icon. You can also link to IBM Direct to order hardcopy books.

### *z/VM and VM*

The latest ESS support information can be retrieved from:

`http://www.storage.ibm.com/hardsoft/products/ess/supserver.htm`

Support information can also be obtained from the respective PSP buckets. Install service buckets and publications can be retrieved by going to the z/VM main page at:

`http://www.ibm.com/vm`

PPRC is managed in the VM environment either using the ESS Copy Services WUI or the ICKDSF utility. If using ICKDSF, this utility should be at a current release and with the proper maintenance applied.

### *VSE/ESA*

The latest ESS support information can be retrieved from:

`http://www.storage.ibm.com/hardsoft/products/ess/supserver.htm`

Support information can also be retrieved from the respective PSP buckets. Install service buckets and publications can be retrieved going to the VSE/ESA main page at:

`http://www-1.ibm.com/servers/eserver/zseries/os/vse/`

In VSE, as in VM, PPRC is managed either using the ESS Copy Services WUI or the ICKDSF utility. If using ICKDSF, this utility should be at a current release and with the proper maintenance applied.

### *TPF*

The PPRC functions can be invoked and controlled in the TPF environment using the following:

- ESS Copy Services Web User Interface
- ICKDSF utility
- TPF itself
- Other attached servers that support control of PPRC

#### *Web browsers*

The ESS Copy Services interface is a set of Java applets that execute within your Web browser. When a request is made, the Java applets communicate with the microcode running on the ESS clusters to retrieve the current data, submit the request, and display the outcome of the request.

Java technology provides the ability to run programs (applets) inside a Web browser. You must use a browser that contains the proper Java Virtual Machine (JVM) implementation to support these applets. Web browsers vary widely in their JVM support. Consequently, not all browsers are capable of supporting the ESS Copy Services.

The ESS Copy Services Web User Interface supports both Netscape Navigator and Internet Explorer (IE) versions. Netscape 4.76 is the default browser on the ESS Master Console.

Information on supported versions of Netscape Navigator and Internet Explorer can be found in the publication *IBM TotalStorage Enterprise Storage Server Web Interface User's Guide,* SC26-7448.

## 4.8.8 Capacity planning

Capacity planning is essential for successful PPRC implementation. Every write operation to the application site ESS is synchronously mirrored onto the recovery site ESS. This synchronous mirroring comes with an overhead to the write I/O processing of the application. Also, the number of volumes that are mirrored and the write activity rate upon those volumes must be considered in the capacity planning.

All these factors along with the distance between sites should be taken into account when doing the capacity and performance planning. The IBM Field Technical Sales Specialists (FTSS) can provide capacity planning assistance by modeling the effect of different PPRC configurations.

### Identifying volumes to be mirrored

Deciding which volumes are going to be mirrored is part of the capacity planning process. There will be data that is critical for the business operation, as well as other data that is not so important for the business operation. This consideration will play in the determination of which volumes will be in the mirrored farm.

Business impact analysis can be done to determine which applications and associated data are critical for the business operation. This will allow you to identify the data sets, and thus the volumes that must be mirrored.

### Application system I/O performance

The write activity on the primary ESS affects how PPRC impacts the application's I/O activity performance. The following procedure can be used to roughly determine this impact:

1. Look at the peak write rate for the volumes that are to be mirrored.
2. Add this write rate as if they were all DFW (DASD Fast Write), to the total utilization of the ESS.
3. Check if the modeled load provides acceptable DASD I/O performance.

For a more accurate estimation, the IBM Field Technical Sales Specialist (FTSS) can help you with the modeling activity required to find a PPRC configuration that meets the business performance requirements.

### Effect on the recovery system

The ESSs at the recovery site may not necessarily be dedicated to a one-to-one relationship with just one application site ESS. Even some of its volumes may be not part of PPRC relationships, but instead are being used by some applications that run at the recovery site (the recovery site can be an active one and not just a stand-by one). All this I/O activity must be added up when planning the capacity of the ESSs at the secondary site.

Care must be taken also in regarding the access from the host system at the recovery site to the recovery ESS. There must be enough channel paths from the host to the ESS in order to satisfy performance and availability requirements at the recovery site.

### Links

In this section we discuss ESCON and Fibre Channel links.

#### *ESCON*

As a general recommendation, eight ESCON links should be used between the application site and the recovery site ESS. There can be fewer PPRC links if the write activity on the application site ESS is not high. If the write activity is high, then additional links may be needed. The ESS Model 800 with 64-bit ESCON Host Adapters obtains significant gains when adding links, even in addition to the generally recommended eight.

The performance of the PPRC links has been boosted with the latest 64-bit ESCON Host Adapters, which are enhanced with a faster microprocessor, therefore providing increased channel throughput and sequential read bandwidth.

ESCON requires ESCON host adapters in both ESSs. The link can be directly cabled or routed through ESCON directors in the normal way. Because ESCON is a direction oriented protocol, ESCON links can only operate in one direction. Separate physical links must be installed if PPRC paths need to be connected between ESSs using ESCON in the other direction.

PPRC over ESCON supports distances of up to 103 km for synchronous operation. This method is enabled with the PPRC V1 or PPRC V2 features

#### *Fibre Channel*

PPRC over FCP requires a Fiber Channel host adapter in both ESSs. These can be 1 Gb or 2 Gb. The path can then be transported over a SAN network using switches and directors in accordance with normal SAN protocols. As Fibre Channel protocol is bi-directional then the physical links can carry logical PPRC paths in both directions, and through a SAN fabric, to multiple ESSs.

PPRC over FCP supports distances of up to 300 km for synchronous operation. This method is enabled with the PPRC V2 feature.

> **Important:** It is highly recommended that you consult the *Interoperability Matrix* when planning any physical links. This can be found by visiting the ESS homepage:
>
> http://www-1.ibm.com/servers/storage/support/disk/2105.html

The IBM Field Technical Sales Specialist (FTSS) representative will assist in determining the appropriate number of links required for a specific PPRC configuration workload, as well as the effect of distance on the I/O response time.

### Cache

Writes are copied on cache and NVS. For a PPRC write operation, the I/O is completed once the update is in the cache and NVS of the secondary ESS, and the acknowledge comes back to the primary ESS. The ESS Model 800 with up to 64 GB of cache and 2 GB of non-volatile storage (NVS) can deliver excellent levels of performance in a PPRC configuration.

## 4.8.9 Performance considerations

The ESS was originally designed with unique features for making it an exceptional performer when integrated into PPRC solutions. These characteristics are improved even more with the IBM TotalStorage Enterprise Storage Server Model 800 and each new ESS announcement:

- **ESS Model 800:** With this model, up to 125% greater write throughput can be achieved, allowing PPRC to provide excellent response times for synchronous remote copy solutions with random workloads.
- **64-bit ESCON Host Adapters:** These ESCON Host Adapters are enhanced with a faster microprocessor, thus providing an increased channel throughput and sequential read bandwidth. When used in a PPRC configuration on both the primary and secondary ESS, up to a 10% increase in PPRC link throughput for random write operations and sequential bandwidth may be achieved as compared to the previous 32-bit ESCON Host Adapter.
- **Fibre Channel:** This can be used as the communications link between PPRC primary ESSs and secondary ESSs using Fibre Channel protocol (FCP). This allows exploitation of existing Fibre Channel infrastructure and capacity.

#### *ESCON*

The ESCON protocol is more efficient and requires less handshaking than that in previous PPRC implementations. It also permits larger frames of data to be transferred more efficiently across the PPRC links. These ESCON protocol enhancements also cause the ESS to be less sensitive to longer distances between the primary and secondary ESSs, when compared to previous PPRC implementations.

#### *Fibre Channel*

Compared to ESCON, FCP links for PPRC communication reduce the link infrastructure by at least 4 to 1 with equivalent or better performance.

#### *PAV and Multiple Allegiance*

With Parallel Access Volumes (PAV) two or more concurrent writes from the same system image can be driven to the same logical volume, thus reducing the IOSQ contention. This way PAV improves the application’s I/O response time when defined for primary PPRC volumes.

Figure 4-7 illustrates Task1 and Task2 doing writes to the same logical volume at the same time, therefore not increasing the IOSQ time, as would have resulted if Task2 had to wait for Task1 I/O to complete its I/O operation.

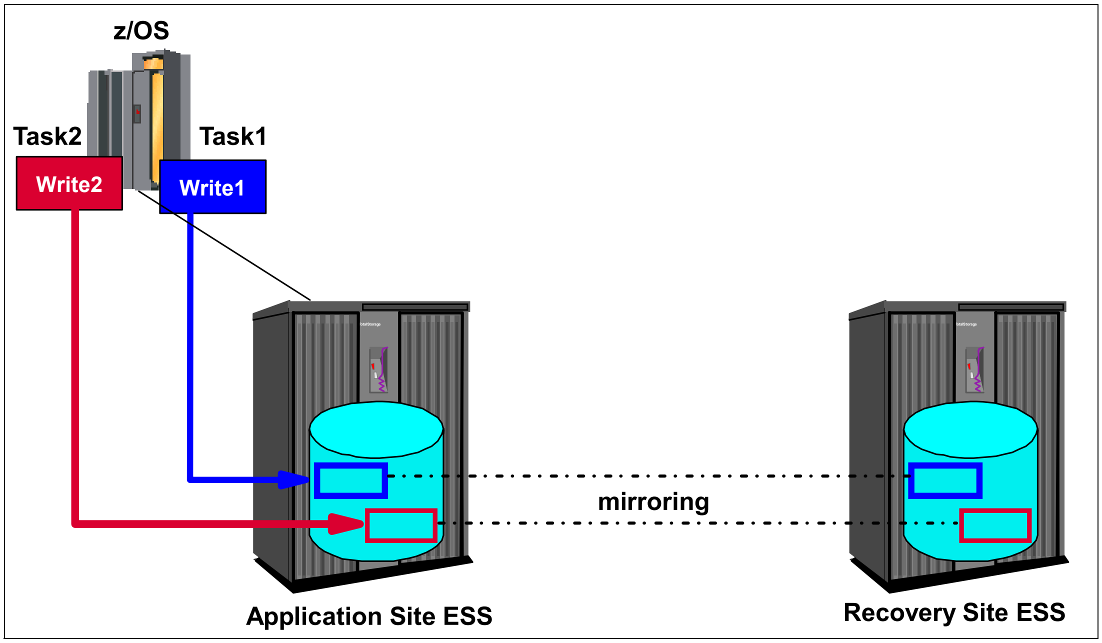



*Figure 4-7 Parallel Access Volumes - PAV*

Multiple Allegiance is another performance feature of the ESS. It allows concurrent access from different system images to the same volume, as illustrated in Figure 4-8.

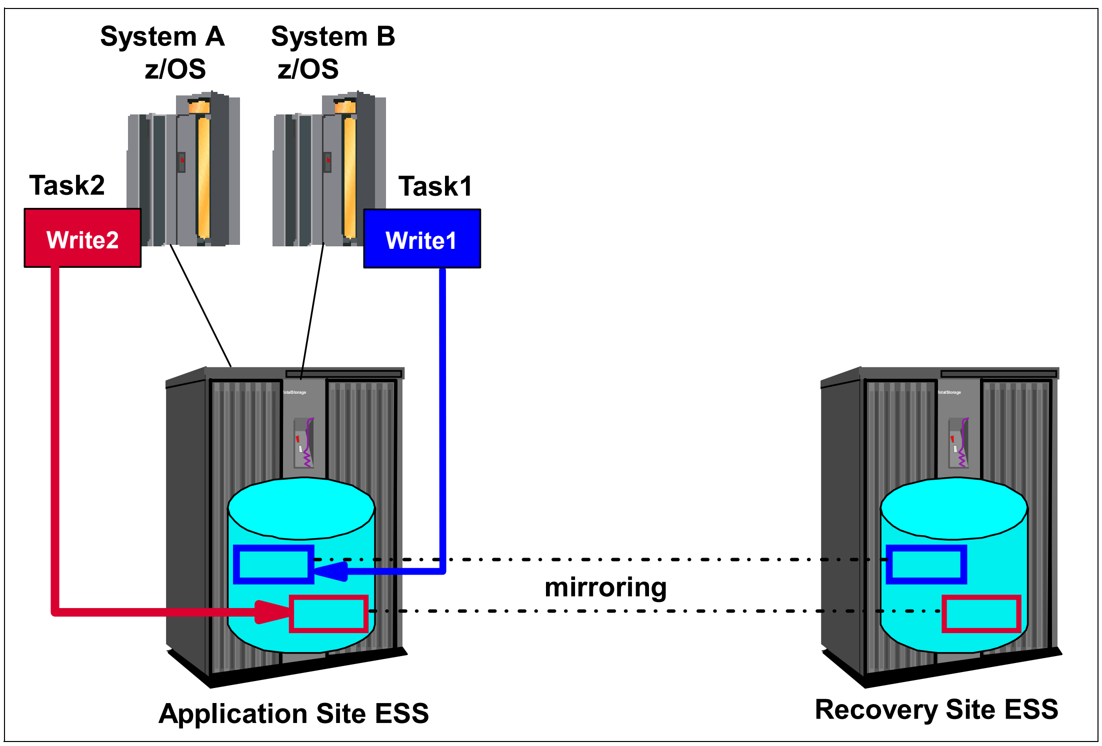



*Figure 4-8 Multiple Allegiance*

Task2 on System A and Task1 on System B are writing to the same volume because ESS allows the writes to occur in parallel maintaining write integrity — as long as there is no extent conflict, and the are paths available.

PAV and Multiple Allegiance can be used on the application primary volumes to enhance the I/O performance of the PPRC volumes, and consequently the application's I/O performance.

PAV and Multiple Allegiance are described in detail in the publication *IBM TotalStorage Enterprise Storage Server Model 800,* SG24-6424.

#### *Performance factors*

Several factors determine the performance of ESS in a PPRC environment. Some of these variables have to do with the configuration being used, such as ESCON or FCP, the distance, and the number of links. Others have to do with the workload characteristics, such as data block size, random versus sequential, and the amount of write activity to the primary ESS.

For a more accurate estimation of the performance of the configuration you are planning, all these factors must be considered, and a proper modeling must be done. Your IBM Field Technical Sales Specialist can help you in these activities.

### Hardware considerations

The following hardware topics should be considered when planning to implement PPRC:

- Both the application and the recovery site disk storage subsystems must be an IBM TotalStorage Enterprise Storage Server.
- The ESCON physical connections between ESSs can be direct fiber optic connections, or with intermediate ESCON Directors. If the application and recovery sites are far from each other, then you can use either channel extenders over WANs or Dense Wave Division Multiplexors (DWDM) over dark fibers. See 3.10.2, "Distance connections" on page 131, which discusses in detail the physical connectivity.
- PPRC over Fibre Channel requires PPRC Version 2 (feature #85xx) and is supported on the ESS Model 800 starting at LIC level 2.3.0. Also, the ESS must have Fibre Channel/FICON host adapters (features #3024 and #3025) installed. See 3.10.3, "PPRC connectivity using Fibre Channel ports" on page 134.
- If the ESS Copy Services Web User Interface (WUI) will be used to manage PPRC, Ethernet connections are required between the ESS subsystems. Chapter 7., "ESS Copy Services Web User Interface" on page 299 discusses in detail the ESS Copy Services WUI.
- Appropriate Licensed Internal Code on the primary and secondary IBM TotalStorage Enterprise Storage Servers is required. This information can be found at:

  http://www.storage.ibm.com/hardsoft/products/ess/supserver.htm
- A compatible secondary volume for each primary volume to be mirrored is needed. The secondary volume must have the identical track capacity and the number of tracks per cylinder, and the same or larger volume capacity as the primary.
- The appropriate PPRC feature code must be enabled on both the application and the recovery ESSs. The PPRC feature code specification depends on the total ESS capacity. Appendix D, "ESS Copy Services feature codes" on page 383 has detailed information on the ESS feature codes.

## 4.8.10 ESS Copy Services server

ESS Copy Services is a combination of software and licensed internal code that runs on each connected ESS cluster in a Copy Services server group — also referred to as a Copy Services Domain. The ESS Copy Services server (CSS) is the ESS cluster that manages the copy services for the group.

For zSeries users, it is necessary to define ESS Copy Service Servers if PPRC and FlashCopy are to be managed through the ESS Copy Services Web User Interface (WUI). In addition, Ethernet connections are required between the managed ESS subsystems — the Copy Services Domain.

The ESS Copy Services Server (CSS) is described in Chapter 7., "ESS Copy Services Web User Interface" on page 299.

## 4.8.11 Production considerations

This section discusses topics to be considered when planning the production environment.

### System IPL volumes

The operating system views PPRC secondary volumes as offline volumes, and permits only a limited set of I/O operations to these volumes. Therefore, if you have to IPL at the recovery site, and you are using TSO commands to invoke PPRC functions, the `CRECOVER` command must be invoked to remove PPRC volumes from their secondary status, and return them to the simplex status. On the other side, if you cannot IPL a system, then you will not be able to submit any TSO commands. So, the following recommendations apply:

- Volumes that contain data sets that are required to IPL the recovery system should not be PPRC secondary volumes. These volumes include the SYSRES, PAGE, and SPOOL volumes, and volumes that have data sets required to initialize JES, VTAM®, and TSO.
- It is recommended to keep at least one IPL-capable system at the recovery site, in order to issue the `CRECOVER` commands, and speed up the recovery process.
- If PPRC is used to copy the system IPL volumes to the recovery site, `CRIT(NO)` should be specified when establishing the volume pairings.

Alternatively, the ESS Copy Services Web User Interface (WUI) can be used to terminate the secondary volumes state. This will reset the secondary volume to simplex mode, at the same time leaving the primary in suspended state.

### Catalogs and control data sets

Disaster recovery time can be reduced if the volumes that contain the master catalog, key user catalogs, and system control data sets are remote copied. This may be a better choice than maintaining a current copy of the catalog, and various control data sets on the recovery system.

### P/DAS

PPRC dynamic address switching (P/DAS) is a software function that provides the ability to redirect all application I/O from one PPRC volume to another PPRC volume with minimal application impact. P/DAS works with PPRC volumes, only.

P/DAS allows application-transparent switching of I/O activity to support the following tasks:

- Planned outages (device or subsystem)
- Device migration
- Workload movement

When using PAV, a P/DAS swap will cause the base and aliases of the primary volume to become offline, and the base and aliases of the secondary volume to become online. If you plan to use PAV on the application ESS, we recommend that you replicate the base and alias addresses definitions on the recovery ESS configuration, and also on the HCD definitions for the recovery ESS.

P/DAS is explained in detail in *z/OS DFSMS Advanced Copy Services,* SC35-0428.

## System managed storage

By using system managed storage (SMS), data can be grouped into specific storage groups, thus controlling what volumes are available as PPRC pairs.

Volumes that are PPRC-eligible will be separated by storage group from volumes that are not being used for PPRC. Any enabled volume in a storage group is eligible for allocation. If non-PPRC volumes are included in a storage group with PPRC volumes, this may result in having only part of an application's data sets, or part of a multivolume data set or database at the secondary site.

If adding volumes to an SMS storage group, the procedures must be updated to include checks to see whether the storage group is being remote copied. If space problems occur, procedures must be in place not only for adding volumes to the storage group, but also for establishing the pairing. If this is not done, the secondary site will be corrupted and useless for recovery.

## DFSMShsm

If any PPRC-eligible volumes are also eligible for either primary space migration or interval migration, then the backup copies of these migration-eligible data sets must be made available to the disaster recovery site. The need for these backup copies can be better understood when comparing the DFSMShsm™ migration process to the remote copy process.

DFSMShsm migration from a primary volume consists of the following steps:

1. A move or delete from the primary volume
2. A change in catalog pointers
3. The reallocation of the data on another DASD device or tape volume

If also copying the primary volume with PPRC, the above DFSMShsm actions appear as a delete of the primary as far as remote copy is concerned. Since the secondary is a track-for-track image of the primary, the delete performed against the primary during the migration process is then reflected to the remote copy secondary volume, which is then deleted.

Any of the following methods can be used to avoid this situation:

- If these data sets are SMS-managed, the ACS routines can be modified in order to assign either a management class, or a storage group construct with attributes to prevent automatic or command migration.
- The ADDVOL command can be issued for the DFSMShsm managed volumes that are going to take part in the PPRC configuration, with the NOAUTOMIGRATION attribute.
- Including the DFSMShsm CDSs and the ML1 volumes in the PPRC configuration. Also, the migration tapes have to be duplicated by using the DUPLEX Tape of DFSMShsm, or the TAPECOPY command.

It is recommendable in all cases to include the volumes where the DFSMShsm CDSs reside in the PPRC configuration.

## ESCON Manager

ESCON Manager is a program offering that uses a TSO ISPF interface to emulate the functionality of a hardware console attached directly to the ESCON Director. It checks the path status on all connected host systems when a disruptive operation is attempted. As PPRC links between ESSs cannot be seen from the host systems, neither ESCON Manager nor the hardware configuration manager (HCM) can check the connections.

The ESCON Director allows you to document links. It is recommended to use this function to define the PPRC links in the active matrix, and fully document them in the operational procedures.

## Resource Measurement Facility (RMF)

Application updates on the primary PPRC volumes, have an overhead because of the extra steps involved in the Synchronous PPRC mirroring. Resource Measurement Facility (RMF) does not detail this data although the I/O operation Response Time reported by RMF includes this overhead.

DASD response time is the sum of queue, pending, connect, and disconnect time:

- Queue (IOSQ) is the time when a request is being delayed in the z/OS system.
- Pending (PEND) time is the time an I/O request waits in the channel subsystem.
- Connect (CONN) time is the time during which the device is actually connected to the CPU (channel, control unit, DASD) and transferring data.
- Disconnect (DISC) time after a command is given to the device, the channel and the controller disconnect. The device seeks the correct location on the disk to read or write the data.

Disconnect time value may increase in a PPRC environment while volumes are in duplex state; the PPRC overhead will be reflected in the disconnect time.

To evaluate how PPRC can affect the response times, a workload analysis must be completed and a model run. RMF data must be collected for volumes which are going to become primary PPRC volumes. The selection of the RMF report interval depends on the user environment, but it should be representative of either a critical processing period or a peak interval. For medium and large zSeries environments with multiple system images, it is recommended to collect RMF data over a week or even more.

The collected data (for example, I/O rates, the individual components of response time, and cache hit ratios) can then be used as input for Disk Magic to determine the expected impact of PPRC. Disk Magic is an IBM internal tool. The IBM Field Technical Sales Specialist (FTSS) can be contacted to assist you in this planning process.

## DASD migration

PPRC is a very effective tool to migrate data from one DASD volume to another DASD volume. The target volume must be of the same type as the old volume. It also must have the same capacity, or larger. The target and the source volume can be in the same ESS, or in different ESSs.

The following example using TSO commands describes how a data migration can be done using PPRC:

1. Vary the secondary devices offline to all system images.
2. Issue the CESTPATH command and CESTPAIR commands with the COPY option to establish the PPRC paths and pairs.
3. Issue CQUERY TSO commands to monitor the initial copy process. The CQUERY command returns information about the state of the PPRC volume, which changes from simplex to pending, to duplex. Wait until all pairs are in duplex state.
4. Stop all applications that are using the primary volumes.

5. Issue the CDELPAIR commands to return the PPRC primary and secondary volumes to simplex state. Optionally, a CDELPATH command can be issued if you already finished with the PPRC paths between the two ESSs.
6. If the PPRC secondary volumes have more cylinders than the primary PPRC volumes, run ICKDSF to refresh the VTOC with the additional space information (see “Refreshing the VTOC” on page 164).
7. If the primary and secondary volumes are under the same system image, vary the original primary volumes offline and the secondary volumes online.
8. Restart all applications, which now use the new DASD volumes.

It is also possible to use P/DAS in this example (see on page 188).

### Planning for testing

It is recommended to create some test scenarios in order to test the PPRC implementation. These test scenarios will help to verify the operational procedures by simulating a failure, and looking at the operating system console messages.

## 4.9 Monitoring Asynchronous PPRC

Asynchronous PPRC can now be monitored and managed by:

- ICKDSF query commands
- ICKDSF messages
- Web User Interface
- SNMP traps
- RMF performance statistics

GDPS, Multiple Device Manager Replication Manager, TSO, and the ANTRQST macro are planned to support Asynchronous PPRC.

### ICKDSF query commands

The ICKDSF Query commands are described in section 4.3, “Using ICKDSF to control Copy Services” on page 156. You can use automation procedures to check for special conditions and highlight important messages to the operator if failure conditions are detected. The most important queries are included in this section.

The QUERY ASYNCCOPY command provides the following important information on ESS box level about an Asynchronous PPRC session:

- Time of the last Consistency Group formed
- Session status
- Information about the three timers
- Master and Subordinate relations
- Error conditions

The QUERY OUTOFSYNCSTATE command provides information on how many of the volumes in the queried ESS that are part of the session have out-of-sync cylinders. This can give you an indication of how the PPRC-XD process is proceeding.

The QUERY SESSIONDEVICE returns the volumes and their state for that addressed LSS.

### ICKDSF messages

ICKDSF messages with begin with ICK. They can be checked and highlighted or suppressed by Automation tools. New messages have been introduced to support FlashCopy, PPRC and Asynchronous PPRC. See *Device Support Facility User's Guide and Reference,* GC35-0033.

### SNMP traps

In a z/OS environment it is not usual to monitor systems with SNMP (Simple Network Management Protocol). If ESSs are used with mixed open systems and z/OS it may be an option to monitor the ESS and Copy Services functions with this protocol. To activate this function you have to enter a trap destination address in the problem notification panel of the ESS specialist. This address can be a Tivoli Enterprise™ Console (TEC) that receives error conditions.

For details see the *IBM TotalStorage Enterprise Storage Server Web User's Interface Guide,* SC26-7448, *and IBM TotalStorage Enterprise Storage Server: User's Guide,* SC26-7445.

The Enterprise Storage Server is delivered with a utility CD that also contains the Management Information Base (MIB). For Asynchronous PPRC new trap types are introduced. An ESS can send these traps with information about Consistency Groups, session status, Master status and other error conditions to the TEC.

## 4.9.1 RMF enhancements: performance statistics reports for ESS links

With ESS LIC version 2.3.x or higher and APAR OA04877 for RMF, new performance statistics of the Enterprise Storage Server for FC PPRC adapters or servers connected via FC adapters can be collected and reported. The reports may be used for the analysis of the external link usage and for performance planning of PPRC and XRC. Currently there is no support available for ESCON adapters.

These link statistics are saved in a new subtype 8 of SMF record 74. The RMF Postprocessor is enhanced by a new ESS Link Statistics report. You can find the detailed description of the field format of the SMF Record in the new APAR.

### RMF Monitor I options

In order to enable or disable the gathering of the RMF Monitor I you simply specify these parameters in the PARMLIB options member for RMF:

ESS / NOESS

NOESS is the default. ESS specifies that Enterprise Storage Server link statistics should be collected. As ESS data gathering involves cache activity measurement, it is recommended to specify ESS *and* CACHE options. If ESS and NOCACHE is specified, cache data gathering is performed implicitly without writing SMF 74 subtype 5 records. Options CACHE and ESS can be specified on any system sharing the selected devices in a Sysplex. Therefore specify options ESS and CACHE together on one selected system to avoid duplicate data gathering.

This APAR updates PARMLIB members ERBRMF00 and ERBRMF02 for Monitor I gathering sessions. ERBRMF00 specifies the NOESS option, while ERBRMF02 specifies ESS.

## RMF Postprocessor Report for link statistics

The syntax of the reports and the values for the exception conditions are described in the *z/OS V1R5.0 Resource Measurement Facility User's Guide*, SC33-7990.

To produce this report, specify the REPORTS(ESS) option in the Postprocessor job. This report contains statistics about the I/O operations for each adapter of an ESS. See Example 4-38, which shows the throughput of two PPRC links and two adapters connected to server HBAs.

The following link types are currently reported in RMF:

- SCSI protocol read and write for Fibre Channel links
- PPRC send and receive activity for Fibre Channel

The report contains values for:

- Transfer rate in MB per second
- Number of MB/s per operation
- Operations per second
- Response time per operation
- I/O intensity

*Example 4-38 New RMF Link Performance Report*

```
1                                          E S S   L I N K   S T A T I S T I C S

PAGE    1
             z/OS V1R4               SYSTEM ID GDP5            START 05/10/2004-16.35.00  INTERVAL 000.04.59
                                     RPT VERSION V1R2 RMF      END   05/10/2004-16.40.00  CYCLE 1.000 SECONDS
OSERIAL NUMBER 0000023953   TYPE-MODEL    2105-800       CDATE  05/10/2004     CTIME 16.35.00      CINT  00.04.59
0------ADAPTER------   --LINK TYPE--       BYTES      BYTES          OPERATIONS      RESP TIME        I/O
 SAID  TYPE                                /SEC       /OPERATION     /SEC            /OPERATION       INTENSITY
00008  FIBRE 2Gb       NO DATA TO REPORT OR ZERO
0000C  FIBRE 2Gb       NO DATA TO REPORT OR ZERO
00024  FIBRE 1Gb       SCSI READ             0.0        0.0             0.0             0.0              0.0
                       SCSI WRITE            8.1M     262.1K           30.9            18.7            579.9
                                                                                                      ------
                                                                                                       579.9
00028  FIBRE 2Gb       NO DATA TO REPORT OR ZERO
0002C  FIBRE 1Gb       PPRC SEND             5.0M      27.0K          183.5            29.2           5356.5
                       PPRC RECEIVE          0.0        0.0             0.0             0.0              0.0
                                                                                                      ------
                                                                                                      5356.5
00084  FIBRE 1Gb       SCSI READ             0.0        0.0             0.0             0.0              0.0
                       SCSI WRITE            8.7M     262.3K           33.0            19.0            626.8
                                                                                                      ------
                                                                                                       626.8
00088  FIBRE 2Gb       NO DATA TO REPORT OR ZERO
0008C  FIBRE 1Gb       PPRC SEND             4.9M      26.8K          184.5            29.1           5362.3
                       PPRC RECEIVE          0.0        0.0             0.0             0.0              0.0
                                                                                                      ------
                                                                                                      5362.3
000A8  FIBRE 2Gb       NO DATA TO REPORT OR ZERO
000AC  FIBRE 2Gb       NO DATA TO REPORT OR ZERO
```

This ESS Link Statistics report provides information for all FC adapters that have activities in the report interval. The report shows two outgoing PPRC links with 5 MB/s data rate during the time interval of 5 minutes and an average I/O rate of 183 I/Os and 262KB I/O block size.

The *Response Time per Operation* is the entire time from sending out a data block until the return receipt of the receiver arrives. On a Fibre Channel multiple data blocks can be sent in sequence regardless of the return receipts of a previous blocks. Therefore the *Number of Operations per Second* times the *Response Time per Operation* can be more than 1000 ms per second. This value is called the *IO Intensity* of the adapter and is displayed in the right column.

## 4.9.2 RMF Exception report and Overview report

If you need to find workload exception conditions of your PPRC links, in order to analyze performance behavior or to obtain the peak write workload, you can use the EXCEPTION RMF report to search a wide time range of your RMF data. Use the IFASMFDP dump program to extract the SMF record 78 subtype 8 from stored long term data and run the Exception report on the extracted data set.

The Overview report shows the values of all selected adapters for intervals that meet the condition used. Example 4-39 presents the throughput for two PRRC links and two Host connected ports.

The syntax of RMF and the detailed description of the exception conditions is available in the *z/OS V1R5.0 Resource Measurement Facility User's Guide*, SC33-7990.

These new conditions are provided on an ESS adapter level. You need to specify an ESS serial number and an adapter ID as qualifier, for example:

- SAID002C - a user specified name of the condition
- ESRTPSD - the condition name - PPRC send transfer rate in this case
- SERN - the ESS serial number
- SAID - the System Adapter Identifier of the ESS adapter

Control statements:

**OVW(SAID002C(ESTRPSD(SERN(0000023953),SAID(02C))))**

**OVW(SAID008C(ESTRPSD(SERN(0000023953),SAID(08C))))**

**OVW(SAID0024(ESRTSWR(SERN(0000023953),SAID(024))))**

**OVW(SAID0084(ESRTSWR(SERN(0000023953),SAID(084))))**

*Example 4-39 Example of Overview report*

```
1                                               R M F   O V E R V I E W   R E P O R T

PAGE  008
             z/OS  V1R4               SYSTEM ID GDP5            START 05/07/2004-17.25.00  INTERVAL
00.00.59
                                      RPT VERSION V1R2 RMF      END   05/10/2004-16.40.00  CYCLE
1.000 SECONDS

 NUMBER OF INTERVALS 332           TOTAL LENGTH OF INTERVALS 05.31.10
-DATE    TIME    INT    SAID002C  SAID008C  SAID0024  SAID0084
 MM/DD HH.MM.SS MM.SS
 05/10 16.30.00 00.59       0.0       0.0       0.0       0.0
 05/10 16.31.00 01.00       0.0       0.0       0.0       0.0
 05/10 16.32.00 01.00       0.0       0.0       0.0       0.0
 05/10 16.33.00 00.59       0.0       0.0       0.0       0.0
 05/10 16.34.00 01.00     161.7K    159.5K      0.0       0.0
 05/10 16.35.00 00.59     166.0K    176.9K      0.0       0.0
 05/10 16.36.00 00.59     635.4K    593.2K     28.8      21.5
 05/10 16.37.00 01.00      10.0M      9.9M     19.6      20.0
 05/10 16.38.00 01.00       8.1M      8.0M     18.7      19.3
 05/10 16.39.00 01.00       5.9M      5.9M      4.4       4.5
```

The Exception report shows only those time intervals where the values exceed a user defined limit.

The Exception report in Example 4-40 only shows you the time intervals when PPRC throughput at the ESS adapter SAID 0024 is greater than 4 MB/s *and* FC throughput at the ESS adapter SAID 008C (Server Adapter) is more than 8MB/s. So you can quickly find out if your configuration has reached its limits.

Control statements:

**EXCEPT(SAID0024(ESTRSWR(SERN(0000023953),SAID(0024)),GE,4000000))**

**EXCEPT(SAID008C(ESTRPSD(SERN(0000023953),SAID(008C)),GE,8000000))**

*Example 4-40 Example of Exception report*

```
1                                         R M F   E X C E P T I O N   R E P O R T

PAGE  001
             z/OS  V1R4              SYSTEM ID GDP5           START 05/07/2004-17.25.00  INTERVAL 00.03.48
                                     RPT VERSION V1R2 RMF     END   05/10/2004-16.40.00  CYCLE 1.000 SECONDS
0
 NUMBER OF INTERVALS 1119          TOTAL LENGTH OF INTERVALS 70.55.11
-CONTROL   DATE    TIME    INT  EXCEPTION       ACTUAL
 STMT NAME MM/DD HH.MM.SS MM.SS THRESHOLD       VALUE  EXCEPTION DESCRIPTION    NAME VALUE
 --------- ----- -------- ----- ------------ --------- ----------------------- ---------------------
SAID008C  05/10 15.48.00 01.00 GE 8000000       22.3M TRANSFER RATE PPRC SEND -SERN(0000023953) SAID(008C)
SAID008C  05/10 15.49.00 01.00 GE 8000000       29.5M TRANSFER RATE PPRC SEND -SERN(0000023953) SAID(008C)
SAID008C  05/10 15.50.00 00.59 GE 8000000       17.7M TRANSFER RATE PPRC SEND -SERN(0000023953) SAID(008C)
SAID008C  05/10 15.51.00 00.59 GE 8000000       13.8M TRANSFER RATE PPRC SEND -SERN(0000023953) SAID(008C)
SAID008C  05/10 15.52.00 01.00 GE 8000000       15.5M TRANSFER RATE PPRC SEND -SERN(0000023953) SAID(008C)
SAID008C  05/10 15.53.00 00.59 GE 8000000       18.3M TRANSFER RATE PPRC SEND -SERN(0000023953) SAID(008C)
SAID008C  05/10 15.54.00 01.00 GE 8000000       19.8M TRANSFER RATE PPRC SEND -SERN(0000023953) SAID(008C)
SAID0024  05/10 16.37.00 01.00 GE 4000000       20.3M TRANSFER RATE SCSI WRITE-SERN(0000023953) SAID(0024)
SAID008C  05/10 16.37.00 01.00 GE 8000000        9.9M TRANSFER RATE PPRC SEND -SERN(0000023953) SAID(008C)
SAID0024  05/10 16.38.00 01.00 GE 4000000       17.9M TRANSFER RATE SCSI WRITE-SERN(0000023953) SAID(0024)
SAID008C  05/10 16.38.00 01.00 GE 8000000        8.0M TRANSFER RATE PPRC SEND -SERN(0000023953) SAID(008C)
```

## 4.9.3 Using Asynchronous PPRC Utilities for ICKDSF Users

There are plans to provide REXX scripts to summarize the output of the Asynchronous PPRC Queries to ease the recovery in disaster cases. It also supports the creating of jobs, issuing the ICKDSF commands to a large number of disks and checking for FlashCopy conditions.

These scripts are planned to be supplied on the CDs shipped with the ESS at LIC 2.4.0.

## 4.9.4 Example for starting and monitoring an Asynchronous PPRC Session

This section provides an example for the setup of an Asynchronous PPRC session managed by ICKDSF and monitored by the Web User Interface. See Figure 4-9 for details of the configuration used. There are z/OS and AIX systems attached to volumes that are members of the session but we focus on the z/OS Systems.

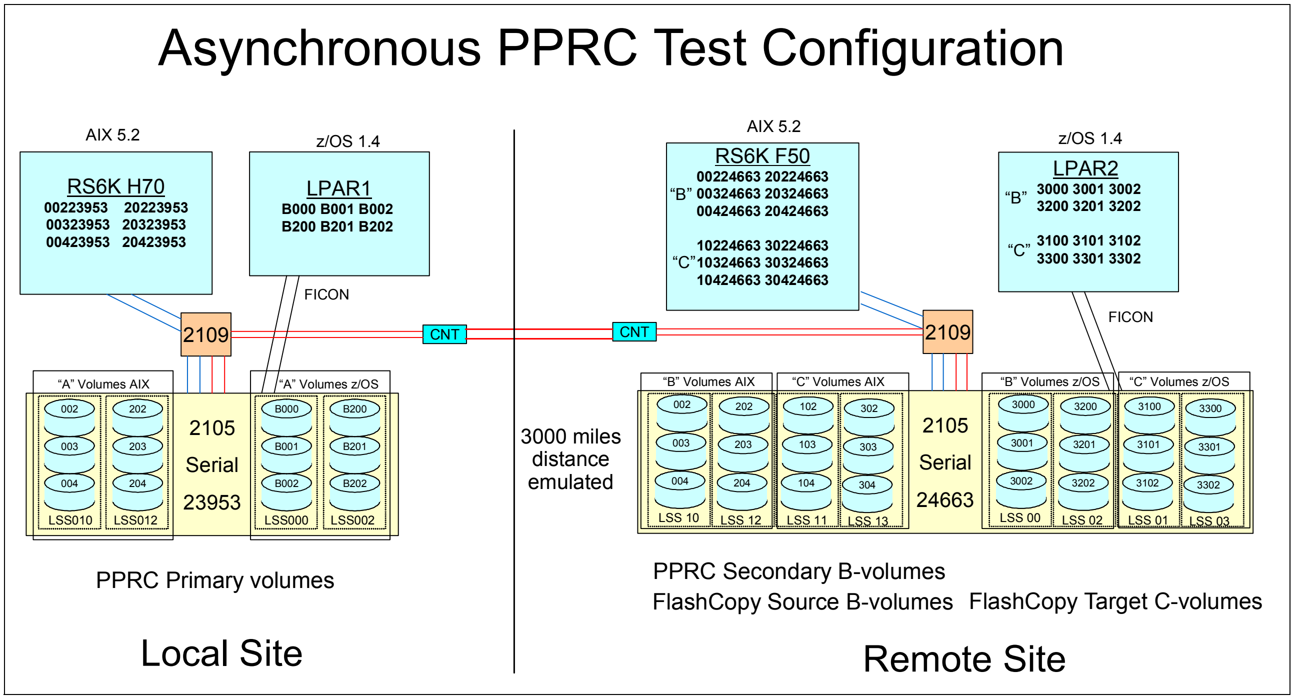



*Figure 4-9 Test configuration for Asynchronous PPRC*

Three volumes in each of two logical subsystems are used in the primary ESS for each operating system. We use one ESS pair only and therefore the primary ESS is also the Master and no Subordinate is used. The distance of 3000 miles is generated by a device with the corresponding signal delay in the line between the Channel Extenders.

**Step 1:** Establish two Fibre Channel PPRC Paths from the local site to the remote site ESS. See Example 4-41 for the ICKDSF commands. These commands have to be used for every logical subsystem.

*Example 4-41 Establish PPRC paths*

```
PPRCOPY ESTPATH                    -
         DDNAME(VLB000)            -
         PRIMARY(X'1340',23953)    -
         SECONDARY(X'1070',24663)  -
         LSS(X'00',X'00')          -
         CGROUP(YES)               -
         WWNN(5005076300C09B29,5005076300C09DEF) -
         FCPPATHS(X'002C002C',X'008C008C')

  PPRCOPY ESTPATH                  -
         DDNAME(VLB200)            -
         PRIMARY(X'1342',23953)    -
         SECONDARY(X'1072',24663)  -
         LSS(X'02',X'02')          -
         CGROUP(YES)               -
         WWNN(5005076300C09B29,5005076300C09DEF) -
         FCPPATHS(X'002C002C',X'008C008C')
```

**Step 2:** Establish six PPRC Pairs in Extended Distance mode. See Example 4-42 for the ICKDSF commands, and Figure 4-10 for the result in the Web User Interface.

*Example 4-42 Establish PPRC-XD pairs*

```
PPRCOPY ESTPAIR                          -
         DDNAME(VLB000)                   -
         PRIMARY(X'1340',23953,X'00')     -
         SECONDARY(X'1070',24663,X'00')   -
         LSS(X'00',X'00')                 -
         OPTION(XD)                       -
         MODE(COPY)

  PPRCOPY ESTPAIR                         -
         DDNAME(VLB001)                   -
         PRIMARY(X'1340',23953,X'01')     -
         SECONDARY(X'1070',24663,X'01')   -
         LSS(X'00',X'00')                 -
         OPTION(XD)                       -
         MODE(COPY)

  PPRCOPY ESTPAIR                         -
         DDNAME(VLB002)                   -
         PRIMARY(X'1340',23953,X'02')     -
         SECONDARY(X'1070',24663,X'02')   -
         LSS(X'00',X'00')                 -
         OPTION(XD)                       -
         MODE(COPY)


  PPRCOPY ESTPAIR                         -
         DDNAME(VLB200)                   -
         PRIMARY(X'1342',23953,X'00')     -
         SECONDARY(X'1072',24663,X'00')   -
         LSS(X'02',X'02')                 -
         OPTION(XD)                       -
         MODE(COPY)

  PPRCOPY ESTPAIR                         -
         DDNAME(VLB201)                   -
         PRIMARY(X'1342',23953,X'01')     -
         SECONDARY(X'1072',24663,X'01')   -
         LSS(X'02',X'02')                 -
         OPTION(XD)                       -
         MODE(COPY)

  PPRCOPY ESTPAIR                         -
         DDNAME(VLB202)                   -
         PRIMARY(X'1342',23953,X'02')     -
         SECONDARY(X'1072',24663,X'02')   -
         LSS(X'02',X'02')                 -
         OPTION(XD)                       -
         MODE(COPY)
```

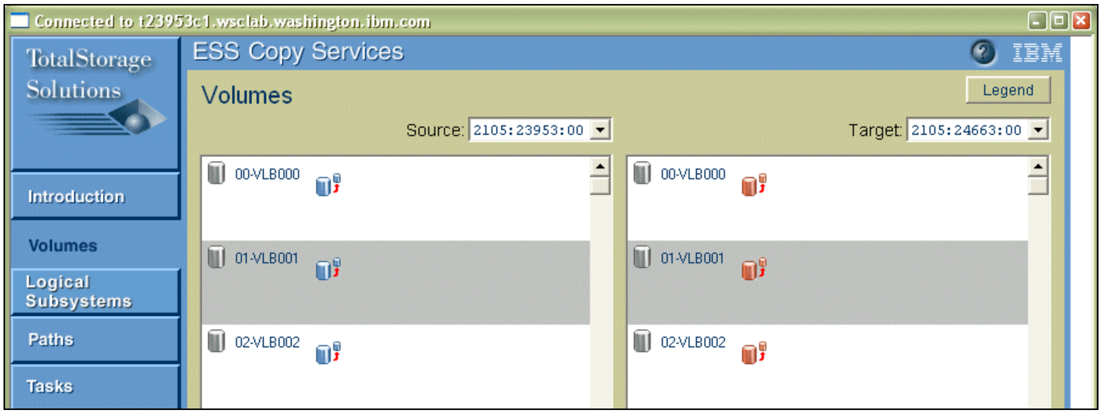



*Figure 4-10 PPRC-XD pairs*

**Step 3:** Establish FlashCopy pairs using Inband FlashCopy at the remote site with the options: *nocopy*, *change recording*, *inhibit target writes*, and *persistent*, which can be addressed to the PPRC secondary volumes. See the ICKDSF commands in Example 4-43 and the presentation of the result in the WUI in Figure 4-11, where you can see that change recording is set on at the FlashCopy Source.

*Example 4-43 Establish FlashCopy Pairs*

```
FLASH   ESTAB                                  -
          DDNAME(VLB000)                        -
          SOURCEVOL(X'00',X'00',X'1070',24663)  -
          TARGETVOL(X'01',X'00',3100)           -
          INHIBITTARGETWRITES(YES)              -
          CHANGERECORDING(YES)                  -
          MODE(NOCOPY)

  FLASH   ESTAB                                 -
          DDNAME(VLB001)                        -
          SOURCEVOL(X'00',X'01',X'1070',24663)  -
          TARGETVOL(X'01',X'01',3101)           -
          INHIBITTARGETWRITES(YES)              -
          CHANGERECORDING(YES)                  -
          MODE(NOCOPY)

  FLASH   ESTAB                                 -
          DDNAME(VLB002)                        -
          SOURCEVOL(X'00',X'02',X'1070',24663)  -
          TARGETVOL(X'01',X'02',3102)           -
          INHIBITTARGETWRITES(YES)              -
          CHANGERECORDING(YES)                  -
          MODE(NOCOPY)

  FLASH   ESTAB                                 -
          DDNAME(VLB200)                        -
          SOURCEVOL(X'02',X'00',X'1072',24663)  -
          TARGETVOL(X'03',X'00',3300)           -
          INHIBITTARGETWRITES(YES)              -
          CHANGERECORDING(YES)                  -
          MODE(NOCOPY)

  FLASH   ESTAB                                 -
          DDNAME(VLB201)                        -
          SOURCEVOL(X'02',X'01',X'1072',24663)  -
```

```
         TARGETVOL(X'03',X'01',3301)           -
         INHIBITTARGETWRITES(YES)              -
         CHANGERECORDING(YES)                  -
         MODE(NOCOPY)

FLASH    ESTAB                                 -
         DDNAME(VLB202)                        -
         SOURCEVOL(X'02',X'02',X'1072',24663)  -
         TARGETVOL(X'03',X'02',3302)           -
         INHIBITTARGETWRITES(YES)              -
         CHANGERECORDING(YES)                  -
         MODE(NOCOPY)
```

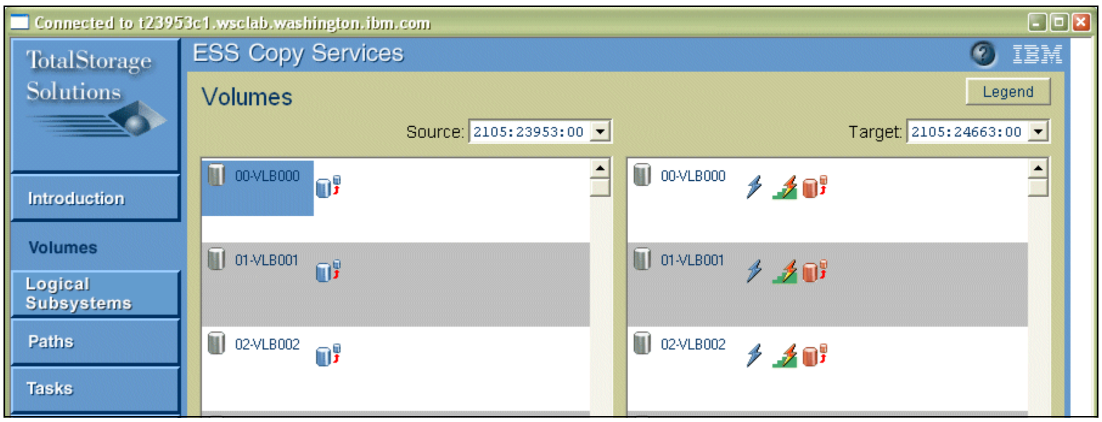



*Figure 4-11 Establish Inband FlashCopy with nocopy, changerecording, inhibit write to target*

**Step 4:** Define a Session to the primary LSS 00 and 02; see Example 4-44.

*Example 4-44 Define an Asynchronous PPRC session*

```
PPRC     DEFINESESSION             -
         DDNAME(VLB000)            -
         SESSIONNO(50)             -
         OPEN
PPRC     DEFINESESSION             -
         DDNAME(VLB200)            -
         SESSIONNO(50)             -
         OPEN
```

**Step 5:** Add six primary volumes to the session; see Example 4-45. The state of the volumes becomes *join pending* as long as the first pass of the PPRC-XD process has not yet finished.

If you have to add many volumes, you can use ranges of volumes in the populate command.

*Example 4-45 Add volumes to the session*

```
PPRC    POPULATESESSION              -
          DDNAME(VLB000)               -
          SESSIONNO(50)                -
          VOLCOUNT(1)                  -
          RANGE(YES)                   -
          RVOLLIST(X'00',X'02')        -
          JOIN

PPRC    POPULATESESSION              -
          DDNAME(VLB200)               -
          SESSIONNO(50)                -
          VOLCOUNT(1)                  -
          RANGE(YES)                   -
          RVOLLIST(X'00',X'02')        -
          JOIN
```

**Step 6:** Start the Asynchronous PPRC session as shown in Example 4-46. The session number is 50, the Maximum Coordination Time is set to 50ms, the Maximum Drain Time is four minutes and the Consistency Group Interval Timer value is ten seconds.

In Figure 4-12 you can watch the new icons for the joining volume and Figure 4-13 shows details of the primary volume 00.

*Example 4-46 Starting the Asynchronous PPRC session*

```
PPRCOPY STARTASYNCCOPY               -
          DDNAME(VLB000)               -
          SESSIONNO(50)                -
          MAXCOORDTIME(50)             -
          MAXDRAINTIME(240)            -
          CGINTERVALTIME(10)           -
          START
```

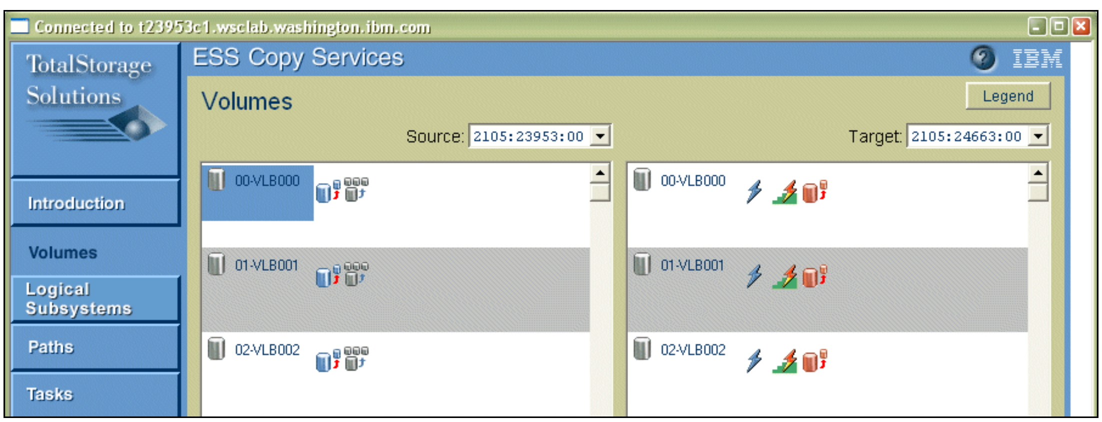



*Figure 4-12 Asynchronous PPRC Session started, primary volumes join pending*

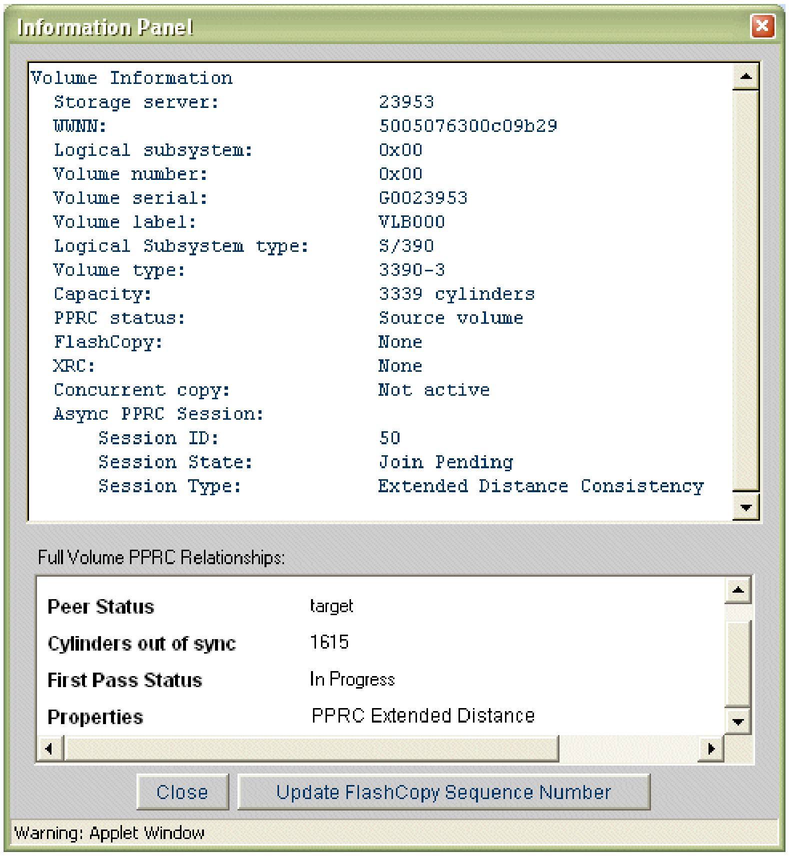



*Figure 4-13 PPRC-XD primary volume first pass in progress, volume status: join pending*

The detailed status of volume 00 in LSS 00 of the target ESS 24663 is presented in Figure 4-14. You can see details of the PPRC and FlashCopy relationships and notice that there is no Consistency Group number yet assigned.

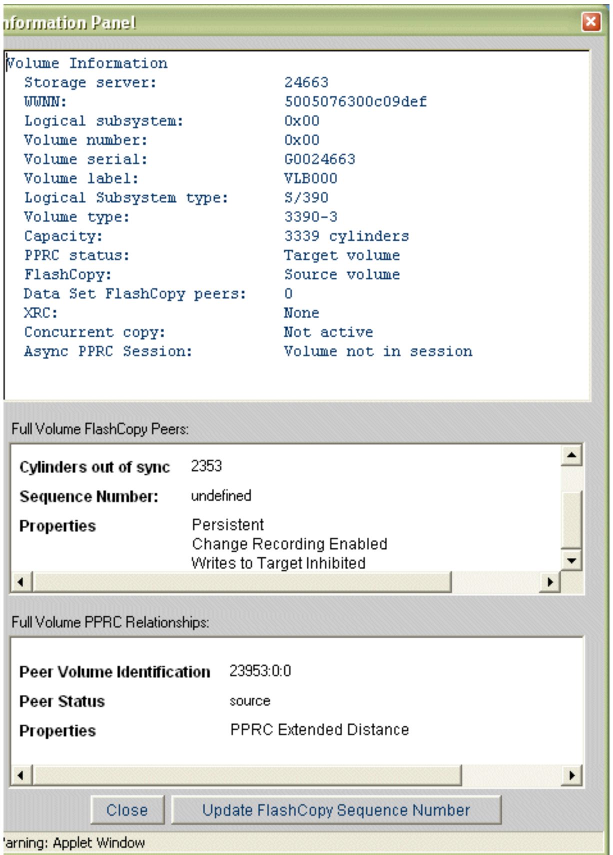



*Figure 4-14 Secondary volume Status*

Figure 4-15 shows the LSS view with Asynchronous PPRC. LSS 00 is the Master LSS that was addressed to start the session and a special new icon is shown that was introduced to identify this LSS. All subsequent session commands should be communicated to this LSS.

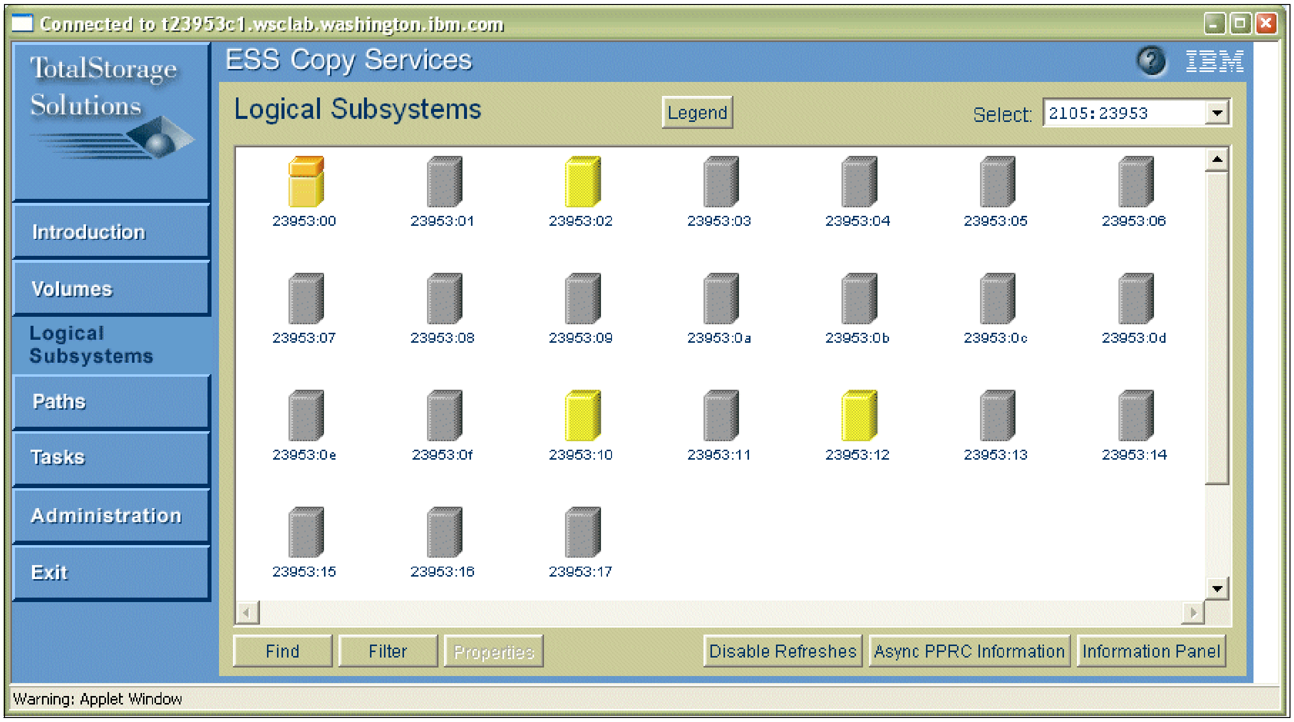



*Figure 4-15 Logical Subsystem view of Asynchronous PPRC*

In Figure 4-15, if you click the button **Async PPRC Information**, you get the information shown in Figure 4-16. The rising value of the *Successful Consistency Group Count* shows that the creation of Consistency Groups is running.

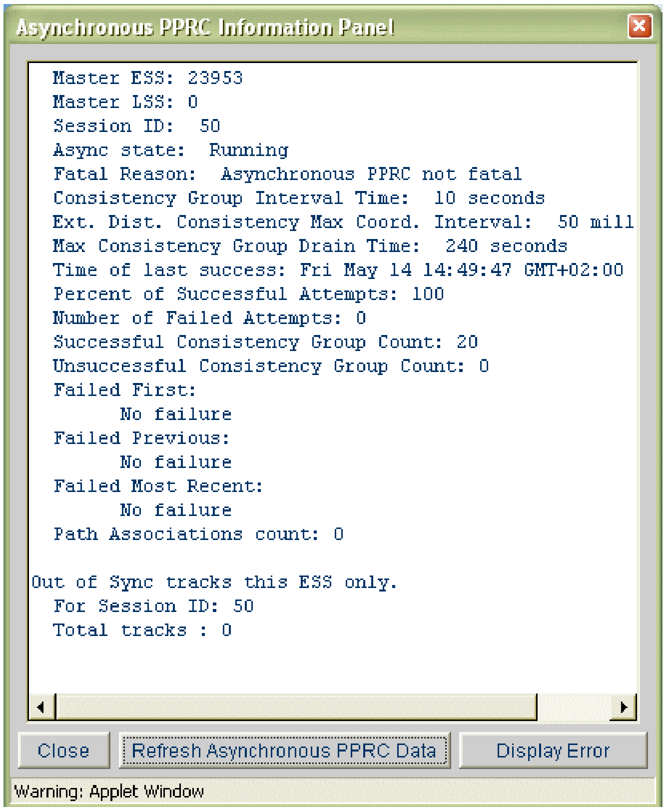



*Figure 4-16 Status of the Asynchronous PPRC session*

Figure 4-17 shows the progress of joining volumes.

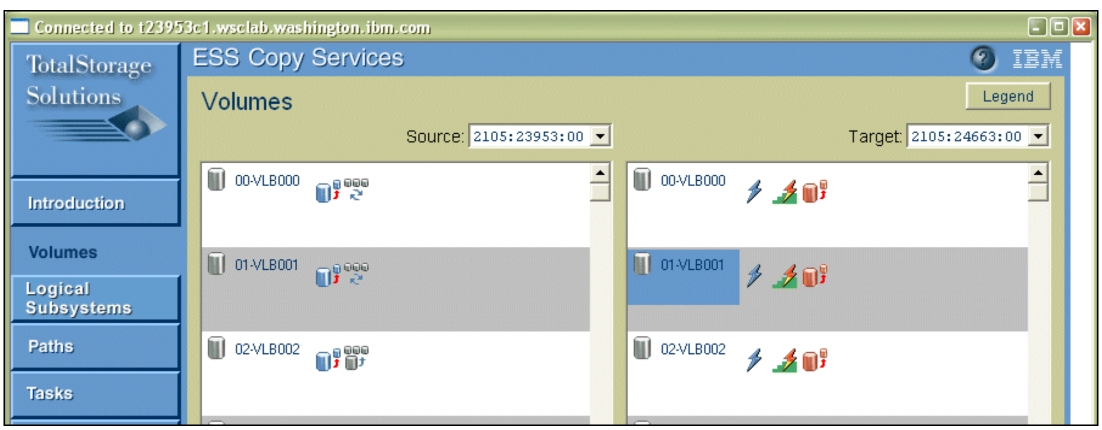



*Figure 4-17 Primary volumes 00 and 01 joined the session, volume 02 is still join pending*

In Figure 4-17, Volumes 00 and 01 have already joined the session, but volume 02 did not yet. A detailed volume query of an already joined volume 00 is provided in Figure 4-18.

Once all of the volumes join the session, Asynchronous PPRC periodically creates Consistency Groups and increments the Consistency Group number that you can check at the PPRC secondary volumes; see Figure 4-19, as well as the FlashCopy target volumes, in Figure 4-20.

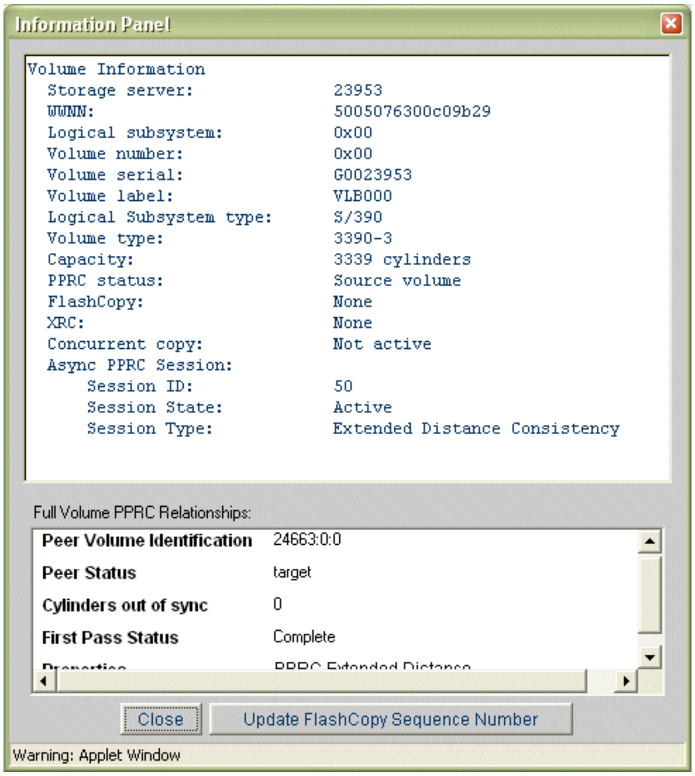



*Figure 4-18 PPRC-XD primary volume with first pass complete and part of the session*

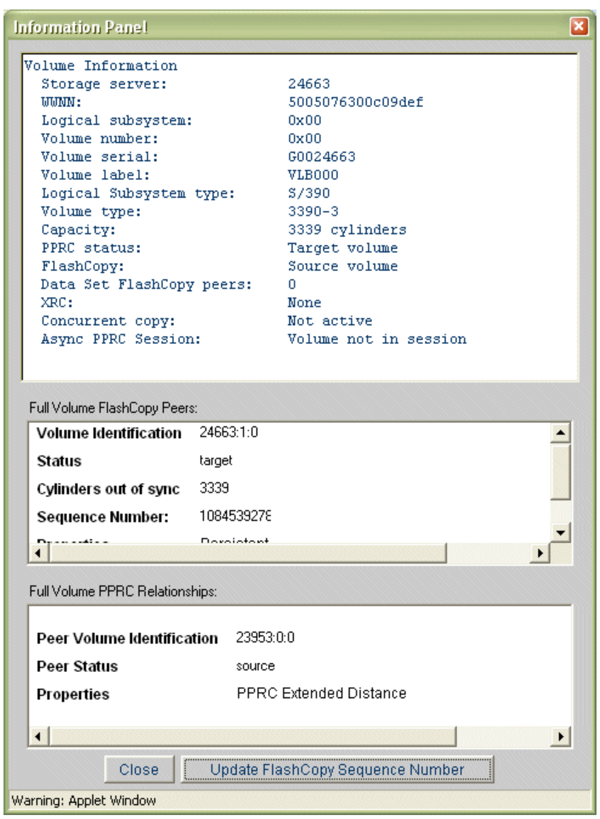



*Figure 4-19 PPRC target volume shows the FlashCopy sequence number*

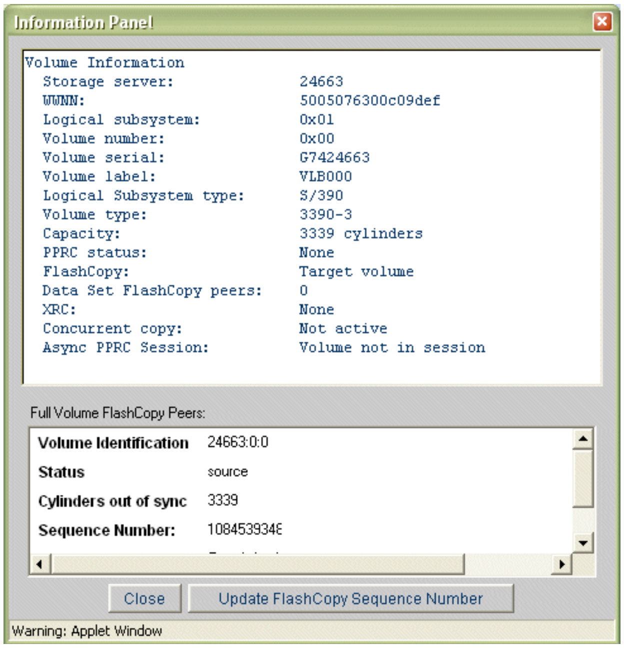



*Figure 4-20 FlashCopy target volume with Consistency Group number*

Finally, you now can watch the session on the Asynchronous PPRC information panel; see Figure 4-21. If there are failures or the Consistency Groups can not be created anymore or creation is delayed, you can see the number of failed attempts in the information panel. The number of out-of-sync tracks for the selected ESS is displayed at the bottom and you can check if this is periodically increasing or decreasing as Consistency Groups are created and drained to the remote site.

To update the view, you have to click the **Refresh Asynchronous PPRC Data** button.

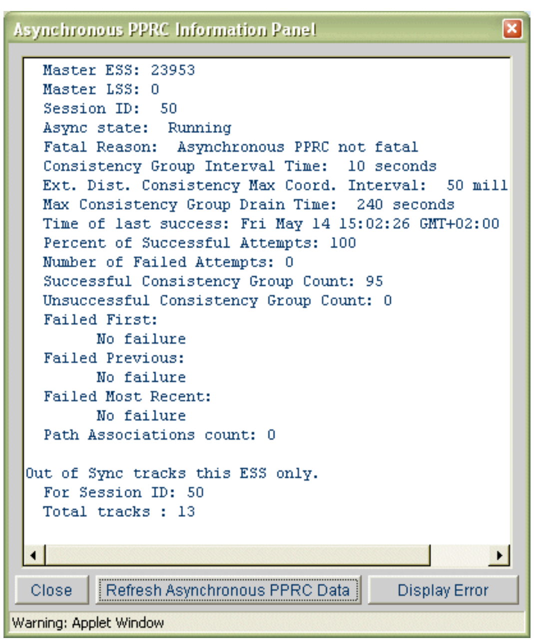



*Figure 4-21 Asynchronous PPRC information panel, all volumes in session*

5

# Extended Remote Copy

This chapter describes the characteristics and operation of Extended Remote Copy (XRC), also known as IBM TotalStorage z/OS Global Mirror. Also discussed are the considerations for its implementation on the IBM TotalStorage Enterprise Storage Server (ESS).

When planning to implement and use XRC, refer to the following publications, which complement the information presented in this chapter:

- *IBM TotalStorage Solutions for Disaster Recovery,* SG24-6547
- *z/OS DFSMS Advanced Copy Services,* SC35-0428
- *DFSMS Extended Remote Copy Installation Planning Guide*, GC35-0481
- *DFSMS Extended Remote Copy Reference Information for Advanced Users*, GC35-0482

## 5.1 Introduction

XRC is a copy function available for the z/OS and OS/390 operating systems. It involves a System Data Mover (SDM) that is found only in OS/390 and z/OS. XRC maintains a copy of the data asynchronously at a remote location, and can be implemented over unlimited distances. It is a combined hardware and software solution offering data integrity and data availability that can be used as part of business continuance solutions, for workload movement, and for data migration. XRC is an optional feature on the ESS; see Appendix D, “ESS Copy Services feature codes” on page 383.

## 5.2 Terminology

The following terms are commonly used throughout this chapter:

- **Primary/application site/system:** The site and systems where the production data and applications normally run are referred to as the *primary site* or *primary systems*. Application site and application systems have the same meaning.
- **Secondary/recovery site/system:** We normally refer to the site and systems where the recovery or test data and applications run as the *secondary site* and *secondary system*. Recovery site and recovery systems have the same meaning. However, we prefer the more generic terms secondary site and systems, as XRC can be used for data and workload migration, as well as in a disaster/recovery solution.
- **XRC volume pairs:** XRC will copy *primary volumes* from the primary site to the *secondary volumes* at the secondary site. The primary volume and its corresponding secondary volume makes an XRC *volume pair*.
- **Primary ESS:** The ESS where the production data resides, which contains the primary volumes of the XRC volume pairs, is referred to as the *primary ESS*. It is also sometimes referred to as the *primary subsystem*, or *primary storage control*, or by referring to its *primary logical control units* (LCUs).
- **Secondary ESS:** The ESS where copies of the primary volumes reside is referred to as the *secondary ESS*, or *secondary subsystem*, or *secondary storage control*, or by referring to its *secondary logical control units* (LCUs).

## 5.3 XRC overview

In this overview we explain XRC concepts and how the different components in XRC relate. XRC is a software centric remote copy implementation. A DFSMSdfp™ component called System Data Mover (SDM) will copy the writes issued to primary volumes to the secondary devices. Although the main XRC implementation consists of host resident software, special XRC support is required in the ESS that attaches the XRC primary volumes.

## 5.3.1 XRC history and development

The first implementation of XRC has been available since early 1996. An enhanced version, often referred to as XRC version 2, was released in late 1997 and brought substantial improvements in the areas of performance and operability.

### XRC Version 2

Most of the XRC Version 2 enhancements were included when it was initially released in 1997, but several further improvements have been made available since then through APARs to SDM and microcode upgrades in the primary storage subsystems.

The following is a short summary of the enhancements introduced in XRC Version 2:

- XSUSPEND command; this function is often referred to as planned outage support. It allows for a planned shutdown and restart of SDM without the need for full volume re-synchronization.
- ERRORLEVEL parameter of the XSTART and XADDPAIR command for providing options to control the behavior of the XRC session under error conditions.
- XSET command for changing XRC session parameters dynamically.
- MIGRATE parameter on the XSTART command for reducing resource utilization when XRC is used for data migration.
- XQUERY command, enhanced with additional reporting options.
- Floating utility device so the SDM can dynamically change the device address during an offload operation. The best candidate is passed to SDM based on algorithms in the primary storage subsystem.
- Multiple reader support for having multiple storage control sessions per SDM.
- Dynamic workload balancing for dynamically adjusting the activity from primary systems when SDM *falls behind* in the offloading process.
- Journal data set enhancements to improve performance.
- Control data set allocation as physical sequential for better performance.

Most of the characteristics previously listed will be covered in more detail in following sections of this chapter.

### XRC and ESS: XRC Version 3

When ESS is used as the primary storage subsystem, XRC also supports unplanned outages. ESS will maintain a hardware bitmap of the tracks changed on primary volumes by the primary systems. If an outage occurs, only the changed tracks need to be copied to the secondary volumes when the connection between the ESS and the SDM is reestablished, and by this a full resynchronization of the volumes is avoided. The SDM also exploits new ESS CCWs for improved performance. There are also a number of enhancements to the support for channel extenders.

## 5.3.2 XRC FICON support

The support provided for XRC FICON is shown in Figure 5-1.

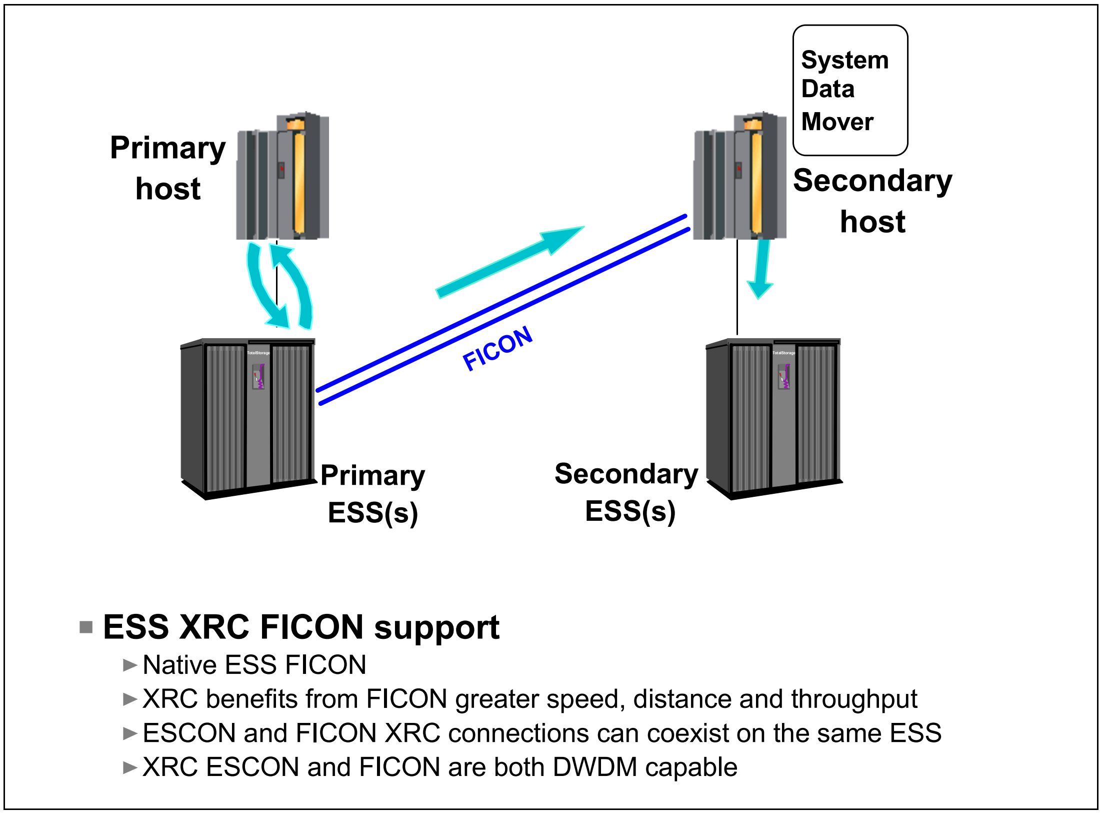



*Figure 5-1 XRC FICON support*

The SDM benefits greatly from the higher bandwidth FICON channels provided to read the updates from the primary ESS as illustrated in Figure 5-1. The transfer bandwidth for a single thread read for XRC can be around three times better than with ESCON due to the increased data rate; a FICON channel can support up to five or more times the throughput of an ESCON channel.

The improved bandwidth of FICON together with the longer unrepeated distance support of up to 10 km (and up to 100 km with repeaters) results in a considerable improvement over ESCON channels. With ESCON, the recommendation was channel extenders through telecom lines beyond 25 km. With FICON, if the infrastructure is available, it may not be necessary to use channel extenders at distances of up to 100 km. These capabilities position XRC as a disaster recovery solution for the metropolitan area implementations as well.

## 5.3.3 Multiple Extended Remote Copy (MXRC)

Multiple Extended Remote Copy (MXRC) allows you to run up to five XRC sessions within a single LPAR; the sessions may or may not be coupled.

### 5.3.4 Coupled Extended Remote Copy (CXRC)

The CXRC capability is shown in Figure 5-1.

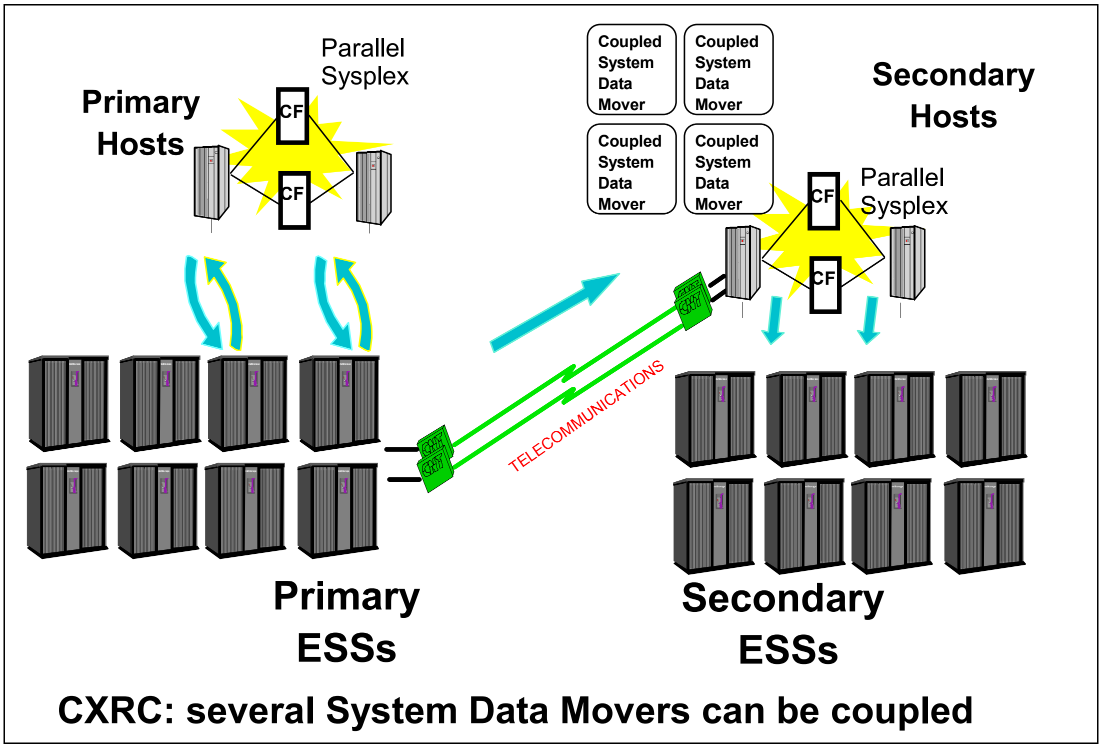



*Figure 5-2 Coupled Extended Remote Copy (CXRC)*

Coupled Extended Remote Copy (CXRC) expands the capability of XRC so that very large installations that have configurations consisting of thousands of primary volumes can be assured that all their volumes can be recovered to a consistent point-in-time (see Figure 5-2). Up to 14 XRC sessions can be coupled to a single master session. The multiple SDMs coordinate their Consistency Group processing such that the recovery of all secondary volumes can be done to a single point of consistency.

In a disaster recovery situation, for the recovered data of a particular application to be usable, the data across all the volumes that make up the application set must be recovered to a consistent point-in-time. CXRC provides this capability by allowing multiple XRC sessions to be coupled with a coordination of consistency between them. This way, all the volumes in all the coupled sessions can be recovered to a consistent point-in-time.

## 5.4 XRC components

This section introduces the main components of the XRC environment.

### 5.4.1 Primary storage subsystem

The primary storage subsystem is a collection of volumes that is designated to be copied to a secondary site. This collections of volumes may be all the volumes at the primary site, or a subset of them. An XRC primary volume must be part of a primary storage subsystem that is XRC capable — like the ESS with the optional XRC feature enabled. Attaching the primary volumes are one or more host applications running on one or more z/OS (or OS/390) system images.

## 5.4.2 Secondary storage subsystem

The secondary site storage subsystem is a collection of volumes that holds copies of the primary volumes. Each XRC primary volume has a corresponding secondary volume. The XRC secondary volumes can be part of any storage subsystem supported by the secondary system, and by the z/OS system where the SDM is running.

## 5.4.3 System Data Mover (SDM)

The System Data Mover (SDM) is part of the DFSMSdfp software, and must have connectivity to the primary volumes and to the secondary volumes. When primary systems write to the primary volumes, the SDM manages the process of copying those updates to the secondary volumes.

Multiple XRC sessions can be in effect per z/OS (or OS/390) system (up to 5 with or without coupling them). Multiple instances of SDM on separate z/OS images are also possible. Each SDM will have one XRC session that is responsible for a group of volumes. SDM maintains the updates' sequence consistency for the volumes participating in the XRC session, across LCUs in the ESS and across ESSs (as well as with other primary storage subsystems which support XRC). In a CXRC environment, the update sequence integrity is managed across the coupled SDM sessions.

The SDM for XRC operates in at least two system address spaces, ANTAS000 which is automatically started during IPL, and ANTASnnn (an ANTASnnn session will be started for each XSTART command). These address spaces are unswappable.

ANTAS000 handles TSO commands that control XRC. If this address space is cancelled, it is automatically restarted with no impact on XRC operations. If the address space for some reason is not automatically restarted, you can submit a job to restart it (see "Restarting XRC address space" on page 268).

ANTASnnn manages the movement of data from the primary to the secondary volumes. It manages the journal data sets and controls the application of updates to secondary volumes. If this address space is cancelled, the XRC session for this SDM will be terminated.

## 5.4.4 Journal, control, and state data sets

The *journal data set*, *control data set*, and *state data set* are used by the SDM to harden — on disk — consistent time groups of updated records received from the primary volumes, and to control the process of applying them to the secondary volumes thus maintaining sequence consistency. SDM creates Consistency Groups and writes them to the journal data sets.

The control data set has pointers into the journal data sets indicating the last set of Consistency Groups written to the secondary volumes and the amount of data written to the journal.

The state data set maintains a list of XRC primary and secondary volumes, and is updated whenever a volume pair status for that session is changed.

## 5.4.5 Master data set

The master data set is only required for a coupled XRC (CXRC) environment. It ensures recoverable consistency among all XRC subsystems contained within the CXRC system. At regular intervals the SDM writes data to the master data set. This data includes the last Consistency Group written to the secondary volumes, and the last Consistency Group written to the journal data sets. In the same I/O operation, the SDM reads the status of the last

Consistency Group written to the secondary volumes, and the last Consistency Group written to the journal data set by all other SDMs in the coupled sessions. It also writes information about its own journal and write consistency time.

## 5.5 XRC operation: data flow

For an overview of the XRC components, see Figure 5-3.

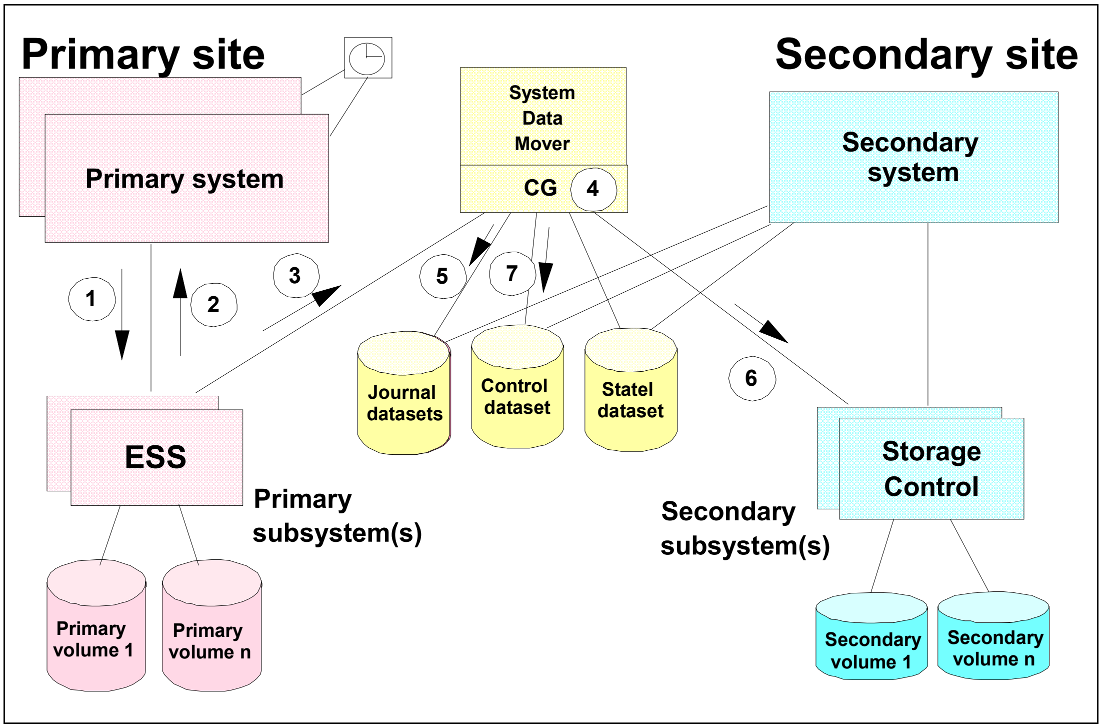



*Figure 5-3 XRC data flow*

Figure 5-3 illustrates a simplified view of the XRC components and the data flow logic. But before discussing the logic, let us understand a key issue for the characteristic consistency of this process: *timestamping*.

### Common time reference

When an XRC pair is established, this is signalled to the primary system, and the host system DFSMSdfp software starts to time stamp all write I/Os to the primary volumes. This is necessary to provide data consistency across multiple LCUs.

If those primary volumes are shared by systems running on different CECs, an IBM Sysplex Timer® is required to provide a common time reference. If all the primary systems are running in different LPARs on the same CEC, the system time-of-day clock can be used.

You should be aware that the primary systems use Universal Time Coordinated (UTC) time.

XRC is implemented in a cooperative way between the ESSs on the primary site and the DFSMSdfp host system software component System Data Mover (SDM). The logic for the data flow is as follows (refer to Figure 5-3):

1. The primary system writes to the primary volumes.

2. The application I/O operation is signalled *completed* when the data is written to primary ESS cache and NVS; this is when *channel end* and *device end* are returned to the primary system. Thus, the application write I/O operation has completed, and now the updated data will be mirrored asynchronously as per the following steps.
3. The ESS groups the updates into record sets which are asynchronously off-loaded from the cache to the SDM system. As XRC uses this aynchronous copy technique, there is no performance impact on the primary applications' I/O operations.
4. The record sets, perhaps from multiple primary storage subsystems, are processed into Consistency Groups (CGs) by the SDM. The CG contains records that have their order of update preserved across multiple LCUs within an ESS, across multiple ESSs and across other storage subsystems participating in the same XRC session. This preservation of order is absolutely vital for dependent write I/Os such as databases and their logs. The creation of CGs guarantees that XRC will copy data to the secondary site with update sequence integrity.
5. When a CG is formed, it is written from the SDM real storage buffers to the journal data sets.
6. Immediately after the CG has been hardened on the journal data sets, the records are written to their corresponding secondary volumes. Those records are also written from SDM's real storage buffers. Because of the data in transit between the primary and secondary sites, the currency of the data on secondary volumes lags slightly behind the currency of the data at the primary site.
7. The control data set is updated to reflect that the records in the CG have been written to the secondary volumes.

## 5.6 XRC volume states

The following terms used in this chapter describe XRC volumes and volume pairs, and the states in which they operate:

**Primary volume** The volume you want XRC to mirror to a remote location.

**Secondary volume** The mirroring volume that receives the updates that are made to the primary volume.

**Target volume** Another name for the secondary volume when you use XRC as a data migration tool.

**Active volume** A primary volume in an XRC session that receives application data updates that are also copied to a remote location. An active volume can be in duplex, seqcheck, copy, or pending status.

**Suspended volume** An established primary volume in an XRC session whose data XRC is no longer copying to a remote location. The volume remains in the XRC session.

**Utility volume** A primary XRC volume that is available to the XRC data mover for reading data from a LCU. You have the option to copy an XRC utility volume to a remote location.

**XRC volume pair** An established primary and secondary (or target) volumes that contain identical data.

**Copy volume pair** A volume pair that XRC is currently synchronizing. Copy volume pairs are not part of the XRC session, and therefore are not eligible for recovery.

**Duplex volume pair** A volume pair that has been synchronized or resynchronized, or that was established with the NOCOPY option specified. The pair is active, and is in neither suspended nor pending states.

**Pending volume pair** A volume pair that XRC has not yet synchronized or re-synchronized. Pending volume pairs are not part of the XRC session, and are therefore not eligible for recovery.

**Seqcheck volume pair** A volume pair that contains non-timestamped writes. This temporary condition indicates that data consistency is uncertain, for this volume pair, in the event of a Disaster Recovery. The pair automatically returns to the duplex state when XRC successfully writes a timestamped update to the secondary volume.

**Suspended volume pair** An established volume pair in an XRC session where XRC is no longer copying data from the primary volume to the secondary volume. The volume pair remains in the XRC session.

## 5.7 XRC sessions

The SDM operates the XRC sessions. The following definitions describe the XRC session types:

**XRC session** Describes an XRC environment

**Active session** The data mover has been initiated with an XSTART command and is available to copy changed records from primary volumes to secondary volumes. In an active session, XRC accepts the XADDPAIR, XDELPAIR, XQUERY, XSUSPEND, XSET, and XEND commands.

**Inactive session** The data mover has been ended by an XEND command, and is no longer active. There is no ANTASnnn address space. In an inactive session, the only commands accepted by XRC are the XSTART and XRECOVER commands.

**Suspended session** An XSUSPEND(session) command with the TIMEOUT option has suspended the XRC session. The ANTASnnn address space is no longer active. You can issue an XSTART command to restart the suspended session or a XRECOVER command to recover the session at a secondary site.

For a storage control that is not an ESS, and therefore does not support hardware bit mapping, you must issue the XSTART command prior to the timeout expiration of the primary storage control. If the timeout expires, the storage control ends its XRC session. Volumes will require a full synchronization when added back to the session.

If the timeout expires in a storage control that supports hardware bit mapping, like the ESS, the hardware bitmap continues to monitor the volumes. When you restart the session and issue an XADDPAIR command, XRC reads the bitmap and resynchronizes only the changed tracks.

**Recovery session** A XRECOVER command has initiated the data mover, which is available to update secondary volumes with unapplied journal records. SDM only accepts the XQUERY command after you have issued the XRECOVER command. The XRC session returns to the inactive mode when the recovery function is complete.

**Coupled session**. An individual XRC session has been added to a master session.

### XRC sessions operation

When an XRC session is started, it is given a session-id and volume pairs can be added to it. A volume pair can only belong to one XRC session, but a primary volume in one session can be the secondary in another, and a secondary volume can be primary in another session.

An XRC session can map a volume from any primary ESS logical control unit (LCU) to any secondary subsystem. Furthermore, primary volumes behind an ESS LCU can map onto any number of secondary subsystems, and volumes behind multiple primary ESSs can map onto a single secondary subsystem.

Control for starting, ending and recovering XRC operations is at the XRC session level. If multiple XRC sessions are required, multiple instances of the SDM must be started. Depending on how many SDMs are required you may need multiple z/OS images, because each z/OS system supports up to five SDMs only.

If you are considering running multiple XRC sessions, you should be aware of the fact that to provide data consistency across XRC sessions, you will need to establish a CXRC environment.

## 5.8 XRC storage control sessions

When you start an XRC session (using the TSO command XSTART) and establish the first volume pair (using the TSO command XADDPAIR), an *XRC storage control session* (XRC SC session) is created within the ESS LCU where the primary volume belongs. This XRC storage control session in the LCU has a counterpart in the SDM. It is possible to have multiple XRC storage control sessions within an LCU per SDM. SDM will use a separate utility device and run a separate subtask per XRC storage control session, making it possible to transfer data in parallel between one LCU and one SDM. This functionality is also referred to as *multiple reader support*.

You can use the SCSESSION parameter in the XADDPAIR command to route a primary volume to a specific storage control session within an LCU. The primary volume will be part of the default XRC storage control session when no SCSESSION parameter is specified.

Figure 5-4 illustrates an example with multiple *XRC sessions* per LCU and multiple *XRC storage control sessions* per XRC session per LCU. Each XRC storage control session is responsible for a group of volumes and uses a separate *utility device*.

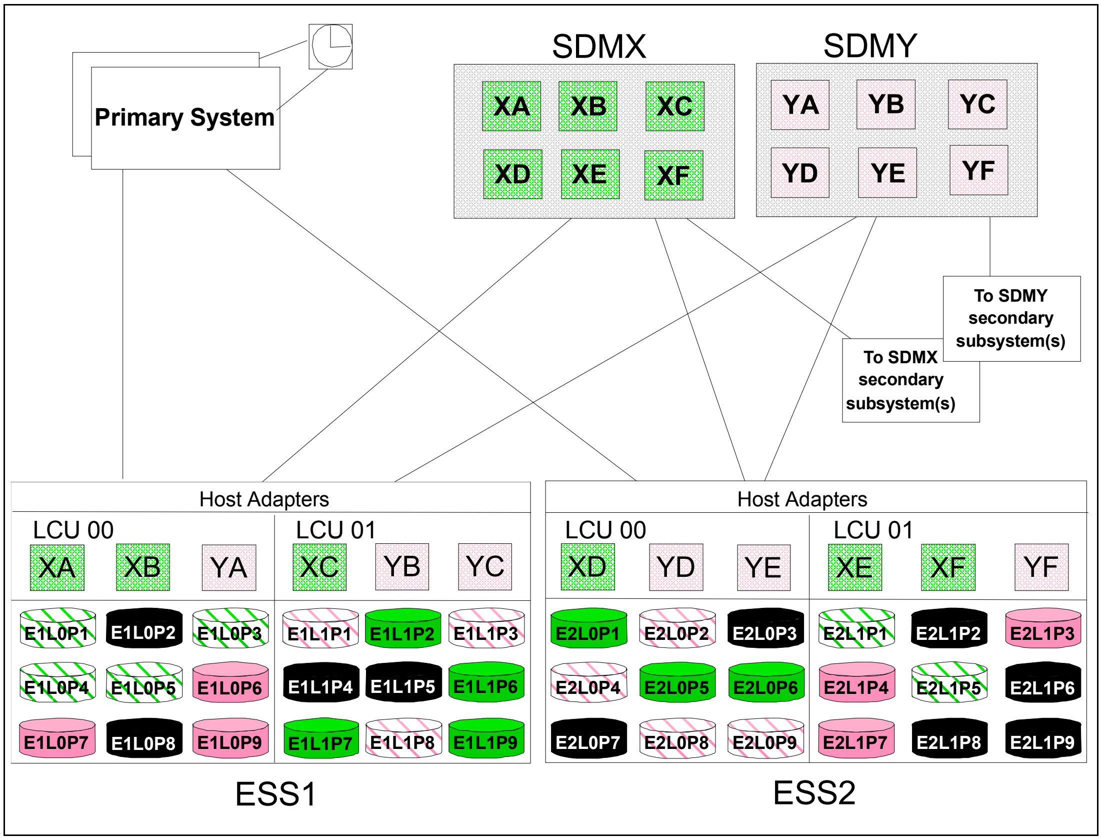



*Figure 5-4 XRC sessions and XRC storage control sessions*

In the example illustrated in Figure 5-4 we have two SDMs (SDMX and SDMY) and two ESS primary subsystems (ESS1 and ESS2) each with two LCUs (LCU 00 and LCU 01):

- ESS1/LCU 00 has three XRC storage control sessions: XA and XB in the SDMX session and YA in the SDMY session.
- ESS1/LCU 01 has three XRC storage control sessions: XC in SDMX session and YB and YC in SDMY session.
- ESS2/LCU 00 has three XRC storage control sessions: XD in SDMX session and YD and YE in SDMY session.
- ESS2/LCU 01 has three XRC storage control sessions: XE and XF in the SDMX session and YF in the XRCD session.

## Concurrent Copy sessions

Concurrent Copy (CC) also uses a data mover when creating a point-in-time copy. This data mover runs in an address space, ANTMAIN, which is automatically started at IPL. Whenever we use the term system data mover or SDM in this chapter, we are referring to XRCs SDM. Concurrent Copy is discussed in Chapter 6, “Concurrent Copy” on page 273.

The CC data mover communicates with a *CC storage control session* in the ESS LCU, so in fact we can have two types of storage control sessions in the ESS LCU: *XRC storage control sessions* and *CC storage control sessions*. When we refer to a storage control session in this chapter, we are referring to XRC storage control sessions.

### XRC session and XRC SC session: Configuration rules

The following configuration rules apply when defining XRC and XRC SC sessions:

- A maximum of 64 combined XRC Storage Control sessions and CC sessions for each primary site ESS LCU
- A maximum of 80 XRC SC sessions in total (across multiple ESS LCUs) for each SDM

## 5.9 Utility devices

For any host system to read data from a disk storage control unit in z/OS (or S/390), that storage control unit must provide a device address. This is a requirement of the S/390 I/O architecture and applies also when reading from the ESS.

So, when the SDM issues channel commands to read updates from the primary storage control unit cache, it has to specify a device address even though the data in the cache in reality *belongs to* several different devices. This device address used by the SDM to offload updates from primary subsystem cache is referred to as the *utility device*.

If SDM has multiple XRC storage control sessions in one LCU, SDM needs one utility device per storage control session.

It is important that the utility device is not heavily used by the primary system because this could prevent SDM from getting access to the device, and the offload process would slow down.

### Utility device selection method

You can specify the method of selection of the utility devices using the XSET command with the UTILITY parameter. You can choose that a specific (*fixed*) device be used, or you allow the ESS to determine which utility device (*floating*) to use.

#### *Fixed utility device*

For *fixed utility devices* XRC always reads data from the storage control using this device, thus eliminating application contention for devices. The following recommendations apply to these devices:

- Select dedicated fixed utility devices that are not used for any other purpose.
- Ensure that the device is not reserved by other applications.
- Ensure that the first device added to a session for a particular storage control be a fixed utility device.
- Define the fixed utility device as a single cylinder device if the storage control is an ESS. Specify a single fixed utility device per storage control session.

#### *Floating utility devices*

XRC can use a *floating utility device* with the ESS. This means that the microcode in the ESS will continuously monitor the usage of utility device candidates, select the least used device, and pass this address to the SDM. SDM will then use this device address on the next offload operation.

When using floating utility devices, all primary devices in the XRC storage control session are eligible to be used as a utility device, plus all devices added to the storage control session with the XRCUTL parameter in the XADDPAIR command.

XRCUTL devices are not really part of the session because they are not copied to secondary volumes, but they can be used to extend the range of eligible utility devices. For the ESS, you have the option to define a single cylinder custom volume, and then specify this volume in the XADDPAIR command with the XRCUTL parameter.

**Tip:** In a channel-extender configuration, it is highly recommended that you define fixed rather than floating utility devices. In these configurations, the telecommunication links tend to be the most vulnerable. If they fail, it is essential that you do not leave a production volume in the status of an XRC utility device. If an application attempts to reserve a primary volume that is currently being used as a utility device, message `IEA482I WAITING FOR CONTROL UNITS` is written to the system log. The application program will not be able to proceed until the storage control session is suspended. Also, you can define utility devices on additional paths to improve resilience and performance.

## 5.10 Consistency groups

This section discusses in more detail the characteristics of timestamping and the Consistency Groups.

### Timestamping process

Maintaining the update sequence for applications whose data is being copied in real time is a critical requirement for applications that execute dependent write I/Os. If data is copied out of sequence, serious integrity exposures could render the recovery procedures useless. XRC uses special algorithms to provide update sequence consistency for all data. The starting point for maintaining update sequence integrity is when a record is first written by an application system to a primary volume of an XRC managed pair. When the record is written, the LCU maintains the data (including the time-stamp information) and transfers it to SDM along with other updated records. This process is shown in Figure 5-5.

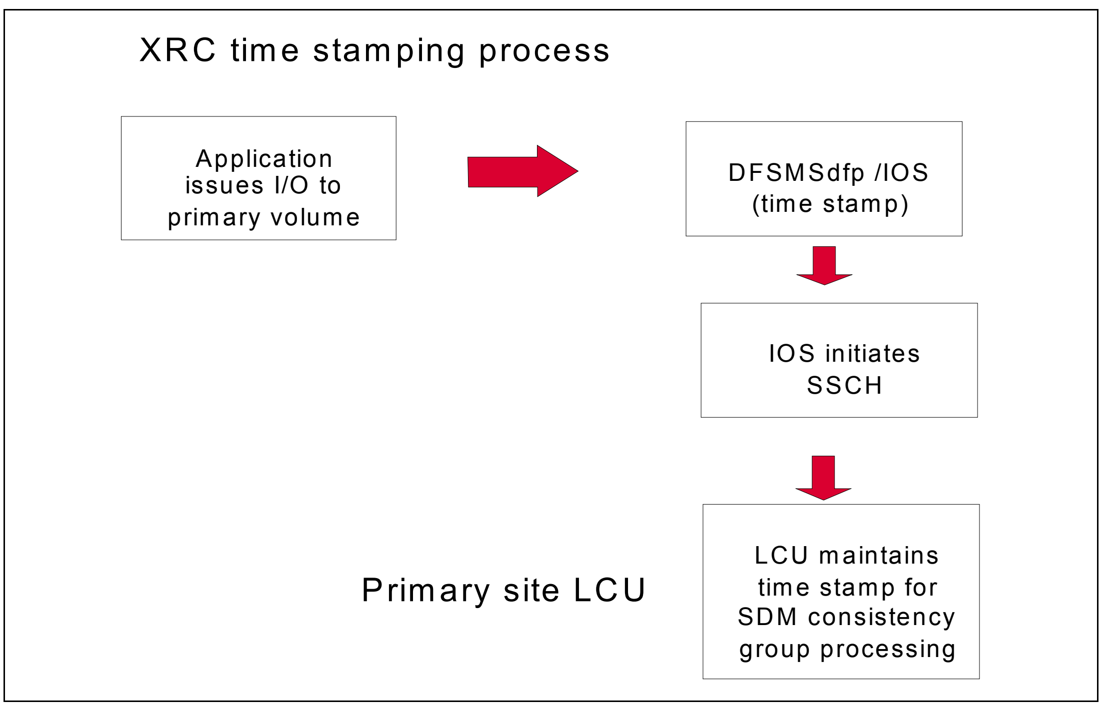



*Figure 5-5 XRC timestamping process*

Whenever an XRC-managed pair of volumes is established by the **XADDPAIR** command, all attached primary systems are informed of their duplex status. Thus, DFSMSdfp can be selective when performing the timestamping process. Only those records that are being written to XRC primary volumes are timestamped for future analysis by SDM. Identification of records that require this special treatment is a result of the pack change interrupt, which is issued when the volumes come under XRC control after the XADDPAIR command is executed. The pack change interrupt allows all attached hosts to update their internal control blocks to reflect the XRC status of the volumes, and thus provides an efficient way for DFSMSdfp to identify which records should be timestamped, and which should not.

The timestamping code is an extension of the IOS SSCH code, so it applies to all data. Deferred writes held in main storage buffers for database applications are processed in SSCH order. This ensures that the timestamping process delivers update sequence integrity support accurately and efficiently.

## XRC update sequence consistency example

The example shown in Figure 5-6 illustrates how the Consistency Groups are created.

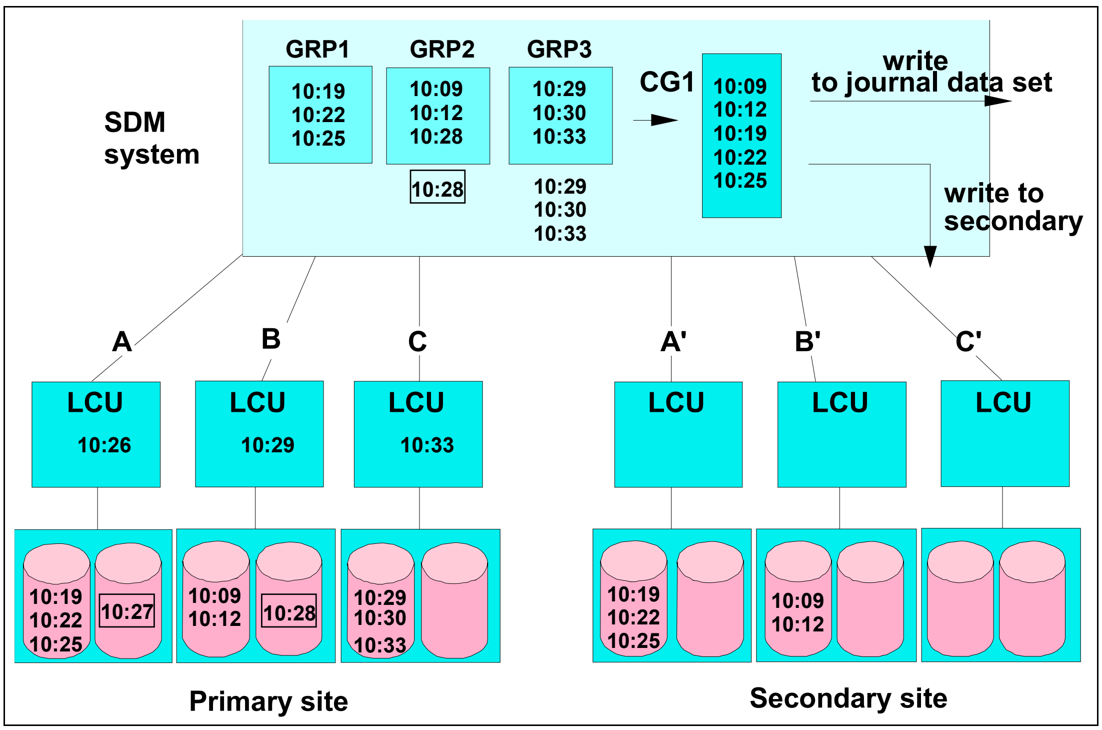



*Figure 5-6 Creation of Consistency Group - CG1*

In Figure 5-6 we see a configuration with six primary volumes attached to three LCUs at the primary site, and six secondary volumes attached to three LCUs at the secondary site. The SDM system has connectivity to the three LCUs at the primary site and the three LCUs at the secondary site. We assume a one-for-one relationship between the LCUs and the primary volumes at the primary site, and the LCUs and secondary volumes at the secondary site (A, B, C, copied to A’, B’, C’, respectively).

The applications running on the primary system update records on the primary volumes at certain times. In our example, the first volume attached to LCU(A) at the application site has three records updated at 10:19, 10:22, and 10:25. Similarly, the times recorded for the other five volumes are the times when their records are updated (ignore the secondary site disk for the time being).

Two record updates are highlighted on two volumes at the primary; we will use these two records as our dependent writes. The first record is the log update on the second volume attached to LCU(A), which takes place at 10:27. The second record is the database update on the second volume attached to LCU(B), which takes place at 10:28.

Now, we define a point-in-time when SDM reads the updated records from each primary LCU. Notice the time in each of the three LCUs at the primary site. We assume that this is the time that SDM will read the records that have been updated from that LCU.

So, at 10:26, SDM will read updates from LCU(A); at 10:29, from LCU(B); and at 10:33, from LCU(C). We will look at each LCUs individually and describe the record transfer to SDM and the CG processing that takes place when all LCUs have sent their updates to SDM.

- SDM reads from LCU(A) at 10:26:

  At 10:26, SDM will read three records from the LCU(A) — the records written at 10:19, 10:22, and 10:25. SDM will not read the highlighted log record because it has not been written at this point-in-time (the log will be written at 10:27). We deliberately stress test SDM by creating a scenario that could easily result in copying data out of sequence. This group of records will be stored in SDM's real storage buffers and is designated as GRP1 on the diagram. SDM will not write this data yet, because it must determine whether the other LCUs have dependent record updates that are included in the session.

- SDM reads from LCU(B) at 10:29:

  At 10:29, SDM will read three records from LCU(B) — the records written at 10:09, 10:12, and 10:28. Note that the highlighted database update (10:28) is included in this group of records, which is described as GRP2 in the diagram. This poses a potential problem because SDM has read the database update, but not the log update. If SDM simply writes data as it receives it, the serious data integrity exposure described before would prevail. SDM's ability to avoid these exposures becomes clear as we proceed with this example.

- SDM reads from LCU(C) at 10:33:

  At 10:33, SDM will read three records from LCU(C) — the records written at 10:29, 10:30, and 10:33. This group of updated records is called GRP3. At this stage, SDM has received responses from all three LCUs that have volumes in the XRC session. The next phase of providing the remote copy function is to perform the update sequence integrity check for all of the updated records read from all of the primary storage subsystems in the XRC session.

- SDM creates CG:

  SDM now has three groups of records that it uses to create a CG. The CG is the unit of data transfer to the journal data sets at the secondary site. It contains records that SDM has determined can be safely written to the secondary site without risk of out-of-sequence updates. To compile the CG, SDM uses all of the record groups read from the LCUs as input. In our case, we have three record groups that SDM uses as input to produce a single CG output.

  SDM compares the maximum timestamp for each individual record group (GRP1 = 10:25, GRP2 = 10:28, GRP3 = 10:33) and selects the smallest value (10:25) to calculate which records should be included in the CG. SDM calculates that all records written at or before this smallest value time can be written together to the remote site journals as a CG.

  In our example, five records (10:09, 10:12, 10:19, 10:22, and 10:25) qualify for inclusion in the first Consistency Group of CG1. Using this calculation, SDM ensures that dependent write records are not written *ahead* of time as demonstrated by the fact that the database update at 10:28 has not been included in the CG. This is vital because the log update (10:27) is located behind a different LCU(A) and has not yet been read into SDM's real storage buffers for inclusion in the *Consistency Group creation algorithm*.

Now, we see the importance of using the maximum timestamp in each individual record group, and selecting the minimum of all of the maximums across record groups as the CG *upper limit* timestamp.

SDM has *retained* the database update along with other possibly dependent write sensitive records in its real storage, until it calculates that it is safe to include them in a future CG. In the diagram, these four records are listed beneath their respective original record groups (10:28 in GRP2 and 10:29, 10:30, and 10:33 in GRP3).

- SDM writes CG to journal data sets:

  Having created a group of records that can be safely written together, SDM then writes this group to the journal data sets at the secondary site. Thus SDM can harden the updates as quickly as possible in one I/O to a single data set.

- SDM writes the updates to the secondary volumes:

  After the CG has been written to the journal data set, SDM immediately writes out the individual records to their appropriate secondary volumes. This transfer takes place from SDMs real storage buffers (the journal data sets are only read during recovery using the XRECOVER command). Updating the secondary volumes could involve several I/Os, because the record updates could be directed to several volumes located behind several LCUs. In this example, the CG comprises five records directed to two volumes attached to two LCUs. The figure illustrates that the five records have been successfully copied to their corresponding secondary volumes — 10:19, 10:22, and 10:25 written to the first volume of LCU(A'), and 10:09 and 10:12 written to the first volume of LCU(B').

Figure 5-7 concludes the example on update sequence consistency by stepping forward in time and describing how the dependent write updates are copied to the secondary site.

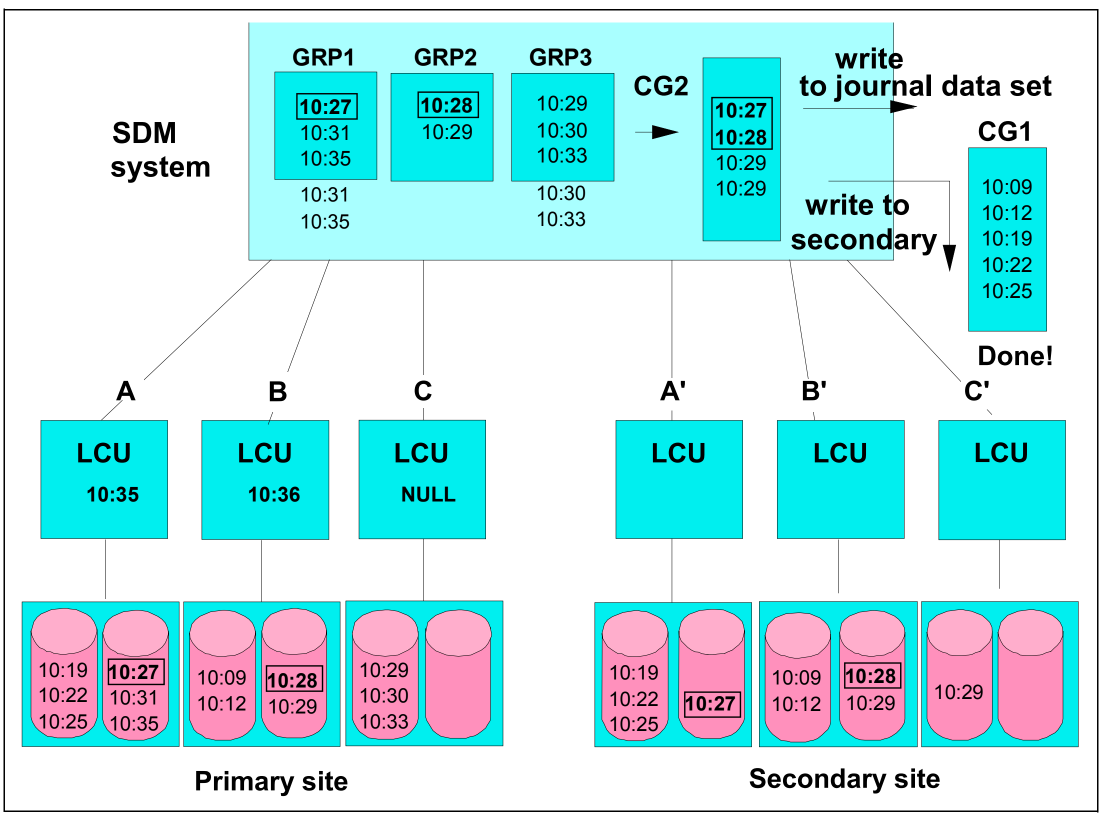



*Figure 5-7 Creation of Consistency Group - CG2*

SDM copies data in real time, so it constantly transfers updated records from perhaps multiple LCUs in an XRC session. In our example, we simulate moving the *clock forward* in time and tracing the steps taken by SDM in the production of its second CG.

Notice that LCU(A) and LCU(B) at the primary site now have different times indicating when SDM reads the updated records. We proceed as before:

- SDM reads from LCU(A) at 10:35:

  At 10:35, SDM reads three records from LCU(A) (10:27, 10:31, and 10:35). Note that the log update (10:27) has now been read. This group of three records has been designated GRP1 in the diagram.

- SDM reads from LCU(B) at 10:36:

  At 10:36, SDM reads one record from LCU B (10:29). This record can now keep the database update record (10:28) company in SDM real storage (these two records are in GRP2 in the diagram). Remember that the database update record (10:28) was not included in the previous CG and was kept in real storage by SDM along with other *residual* records.

- SDM detects NULL response from LCU(C):

  Finally, SDM completes its cycle of updated records collection by establishing that the third LCU(C), which also connects volumes in this XRC session, has *nothing to report*. The NULL response must be sent even if the LCU has no updates, because it indicates that the LCU has not suffered an outage, thus the record groups retrieved to date from other LCUs can be used to build a valid CG.

- SDM creates CG:

  SDM creates the second CG using the same algorithm as before. We can see that the two dependent writes (the log update at 10:27 and the database update at 10:28) have been *captured* in this CG.

  Notice that the record group transferred from LCU(B) produced a maximum timestamp, which also turned out to be the smallest of the three GRP maximums. Notice also that this value (10:29) enabled all records written before or at this time to be included in the CG — thus allowing the log update and the database update to accompany each other on the journey to the secondary site when written to the journal data sets.

  The diagram shows that when SDM writes the records to the secondary volumes, the relationship between the contents of the log and the database, which is vital for recovery, has been protected. The record residue (10:31 and 10:35 from GRP1, and 10:30 and 10:33 from GRP3) will be processed during the next SDM record group collection cycle.

## 5.11 Single command for recovery

One of the most valuable features of XRC is its ability to do a single command recovery. After a disaster that effects the primary site, the secondary site must be made ready for takeover. This involves getting the secondary volumes to a state where their contents are usable in anticipation of application restart. XRC has a single command that performs this task, the XRECOVER command.

XRECOVER commits the last available Consistency Groups to the secondary volumes, reports the consistency time for the secondary volumes, and then clips the volume serial number of each secondary device to that of its matching primary device.

## 5.12 Dynamic workload balancing

With XRC Version 2, a dynamic workload balancing algorithm was introduced. The objective of this mechanism is to balance the write activity from primary systems, and SDM's capability to offload cache during write peaks or temporary lack of resources to SDM, and with minimal impact to the primary systems.

In situations where the SDM offload rate falls behind the primary systems write activity, data starts to accumulate in cache. This is dynamically detected by the primary ESS microcode, and it responds by slowly but progressively reducing available write bandwidth for the primary systems, thus giving the SDM a chance to catch up.

The ESS implements device level blocking. The update rate for a volume continues unrestricted unless a volume reaches a residual count threshold waiting to be collected by the SDM. Whenever that threshold is exceeded, application updates to the single volume are paused to allow the SDM to read them from the cache of the subsystem.

By using the DONOTBLOCK parameter of the XADDPAIR command, you can request XRC not to block specific devices. This option can be used for IMS WADs, DB2 logs, CICS® logs, or spool data sets which use small blocksizes, do numerous updates, and are critical to application response time.

## 5.13 Planned outage support

As the ESS provides XRC Version 3 *hardware bit map* support, suspension of an XRC session has no primary system impact since the ESS maintains a *hardware bit map* of updates, thus not consuming cache resources while sessions are suspended.

### 5.13.1 Suspending an XRC session

The SDM supports two types of suspension. One is an XSUSPEND VOLUME(ALL) suspension. With this, the SDM address space is not terminated. The ANTASnnn address space continues to run and with primary disk subsystems not supporting hardware bit maps the SDM continues to read updates. With the ESS as the primary disk subsystem the hardware bit map is used to track updated tracks. No updates are applied to the journals or secondary volumes.

The second type of suspension is a *session suspension*. This is accomplished by using the XSUSPEND TIMEOUT command. This command is issued when you want to terminate the SDM for a planned activity such as a maintenance update, or moving the SDM to a different site or a different LPAR.

The XSUSPEND TIMEOUT command will end the ANTASnnn address space and inform the involved LCUs that the XRC session is suspended. The ESSs will then use the hardware bit map to record changed tracks, and will free the write updates from the cache.

When the XRC session is restarted with the XSTART command and volumes are added back to the session with the XADDPAIR command, the hardware bit map maintained by the ESS while the session was suspended will be used to resynchronize the volumes, and thus full volume resynchronization is avoided.

### 5.13.2 Suspending volumes

The XSUSPEND VOLUME command will accept a list of volumes (up to 100 volume pairs can be specified) or ALL (meaning all volumes in the session). If ALL is specified, XRC will suspend all volumes in the XRC session at the same consistency time. If a volume list is specified then XRC will suspend all the volumes in the list at the same consistency time. If it is necessary to suspend more than 100 pairs, but not all the volumes in the session, then multiple XSUSPEND VOLUME(volser,volser,...) ATTIME() commands with the same ATTIME will cause the volumes to suspend with the same consistency time. Be sure to pick an ATTIME far enough in the future to allow all commands to be processed.

When volumes are suspended with the XSUSPEND VOLUME command, the ANTASnnn address space remains active; for the suspended volumes, the hardware bit map maintained by the ESS will be used later during the resynchronization process.

## 5.14 Unplanned outage support

The ESS, with its capability of maintaining hardware bit maps, supports *unplanned outages*. This is the capability of avoiding full resynchronization of volumes after an unexpected failure on one or more of the components in the SDM data path.

When data starts to accumulate in cache, there can be several reasons:

- A temporary write peak from the primary systems
- A temporary lack of resource for the SDM
- An outage in one or more of the components necessary for the SDM data moving process (for example, a failure in channel extender links, or failure of the SDM address space itself)

The dynamic workload balancing algorithm (described in 5.12, "Dynamic workload balancing" on page 224) is designed to reduce the effects for situations described in 1 and 2 above, but if this is an outage of a vital SDM component, this mechanism does not help.

When the data in the cache exceeds a predefined level, a time interval starts to decrease. This time interval value is set by the TIMEOUT parameter in the XSET command (not to be confused with the TIMEOUT parameter in the XSUSPEND command!), and can be set individually for each LCU.

The TIMEOUT interval is reset every time the SDM reads from the storage control session. If there is no draining of the cache, the data will continue to accumulate, and will eventually reach a predefined high threshold.

A *long busy* condition will then be presented to the primary systems, and I/Os will be queued in the primary hosts. The duration of the *long busy* condition is the remaining part of the TIMEOUT interval.

When the interval expires, this is what will happen: The SC session is suspended. The hardware bit map will be used to reflect the changed tracks, and the cache resources will be freed. The outage can be of any duration, the primary systems continue to run without any impact from the XRC environment, and the ESS keeps track of the write updates in the hardware bit map. This bit map is used at the later time when the problem is fixed and the volume pairs are eventually added back to the XRC session with the XADDPAIR command.

## 5.15 XRC requirements

This section provides references about the software and hardware requirements when using XRC with ESS.

### 5.15.1 Hardware requirements

XRC hardware requirements are as follows:

- S/390 hardware must be used.
- For the primary ESSs, the XRC optional feature must be enabled. Support is provided by XRC-capable Licensed Internal Code (LIC). Refer to Appendix D, "ESS Copy Services feature codes" on page 383 for further detail on applicable feature codes.
- Primary systems must have a common time reference. XRC uses timestamped record information to ensure that updates are not copied out of sequence at the secondary site:
  - In environments where the primary systems are on different CECs, an IBM Sysplex Timer is required.
  - When you have only one primary system, or the primary systems are running on different LPARs on the same CEC, the system time-of-day clock is used.
- A compatible secondary volume must be available for each primary volume you want to copy. The secondary volume must have the identical track capacity and number of tracks per cylinder as the primary volume. The capacity of the secondary volume can be the same or larger than that of the primary volume.

### 5.15.2 Software requirements

The data mover function requires current OS/390 and z/OS releases with the latest maintenance applied. For the latest maintenance information, refer to the Web site:

http://www.storage.ibm.com/software/sms/sdm/index.html

Then click the **Technical Updates** link. Or, you can go directly to:

http://www.storage.ibm.com/software/sms/sdm/sdmtech.html

### 5.15.3 VM considerations

When XRC manages volumes that are also attached to a VM system, you must define all XRC volumes to VM as unsupported disks. You *cannot* define them as minidisks.

Note that for a disk device that is defined as unsupported under VM, the SET PATH GROUP command is unable to reestablish the path group once a path has been removed.

## 5.16 Initial XRC planning

Before you do further planning, you have to decide which volumes are going to be included in the XRC solution, because this will determine the sizing of your XRC resources. You also have to evaluate your data consistency requirements. This section discusses the considerations when addressing these two topics for the initial planning for XRC.

## 5.16.1 Selecting volumes for remote copy

In this section, we discuss some considerations regarding which volumes to include in the remote copy solution.

#### *Volume level copying*

XRC copies the entire contents of a volume. As a result, all data sets on a primary volume are copied to the secondary volume. This support is for all data types, independent of the applications.

### What data should be copied?

Before you start up your XRC environment, you have to identify the data you will want to mirror onto the secondary site. This may differ depending on whether you use XRC for data or workload migration, or for a disaster recovery solution.

In the remainder of this section, we are assuming that XRC is used for Disaster Recovery.

You should identify the type of data required for a successful recovery. Many installations with an existing secondary site may have completed this task already.

The SYSRES, master catalog, SPOOL volumes, as well as other data sets required to initialize the secondary system (including those used to start up JES, TSO, and RACF) can be copied by XRC. The idea is that even if we do not need the data on some of these volumes, we do need either the data sets correctly allocated and the space formatted (pages, spool, SMF man data sets) available for the production jobs to use (temporary storage groups). Some of these volumes can be initially synchronized and then suspended, so XRC stops copying their updates.

However, the focus for remote copy is on your application data sets that are updated regularly, and are essential for your company. If a disaster occurs, critical application volumes must be made available to your primary application as quickly as possible, depending on the speed of recovery required. XRC offers a single command recovery strategy, which makes it the fastest method for recovering data at the secondary site. A single XRECOVER command recovers all the volumes in an XRC session at once, and all of the volumes are consistent to a single point-in-time.

### Application volumes

Different applications at the primary site may have different priorities. Some of these applications may not warrant the investment required for remote copy. If only a subset of applications is to be copied, the volumes belonging to those applications must be identified and included in the XRC configuration. Even if all applications are required at the recovery site, there still may be volumes that need not be copied.

Because applications deal with data sets and not volumes, multi-volume data sets require special attention. Multi-volume data set types include data sets that reside on multiple volumes, striped data sets, and VSAM spheres. If you want a copy of a multi-volumes data set at the secondary location, you therefore have to copy all volumes on which this data set resides.

### System volumes

The data sets described in the following sections should be given special consideration, as they might be eliminated from a particular installation's XRC configuration.

### *Page*

Page data sets are of no use during recovery at the recovery site. They are owned by the host application site. However, if we do not have the page data sets then we will not be able to IPL the production system at the recovery site until we create and catalog new ones. What can be done is to initially synchronize and then suspend the page data sets. You do not want to have the page data sets in duplex due to the type of I/O that they use.

### *Spool*

The spool data sets should be copied to the recovery site if they are necessary for recovery. Copying the spool increases the amount of copy activity to the secondary. If the contents of the spool are easily re-created, consider excluding spool volumes from the XRC configuration. Alternatively, you can initially synchronize the spool volumes and then suspend them as with the page data sets.

### *Temporary data sets*

Volumes containing temporary data sets (that is, those that exist for the duration of a job or step) can be re-created at the secondary by resubmitting jobs. These volumes should be excluded.

### *SYSRES volume*

The change activity against data sets residing on the SYSRES often does not affect the applications. However, some installations share the SYSRES volume with active data, so the decision on whether to copy the SYSRES volume can depend on the data stored there. Because XRC copies data at the volume level, all data residing on that volume is copied.

If the SYSRES volume is not copied by XRC operations, another method must be used to ensure that a copy is available at the secondary site. Nevertheless, commonly the SYSRES as well as other system volumes will be copied.

### *Master catalog and user catalogs*

Catalog changes are limited, but they are essential for recovery, and therefore they must be considered for XRC copying.

### *Program libraries*

All program libraries that contain modules for XRC applications are essential for recovery and should be considered for XRC copying.

### *Control data sets*

Control data sets, such as RACF, SMS, HSM, and RMM are essential for recovery and should be considered for XRC copying.

### *HSM migration level 1 volumes*

Consider whether migrated data is required at the secondary site. If not, exclude HSM migration level 1 volumes. If migrated data is required but is not critical, these volumes could be copied daily by other means to reduce the amount of copying XRC has to do as migrations and recalls occur.

## 5.16.2 The ERRORLEVEL parameter

As part of the planning process, you should evaluate your data consistency requirements in the event of a failure. To specify how XRC should process certain error conditions, use one of the following ERRORLEVEL subparameters:

- **VOLUME**
- **SESSION**
- **group_name**

You specify the default error level for the entire session with the ERRORLEVEL parameter on the XSTART command, but you can change the error level for specific volume pairs with the XADDPAIR ERRORLEVEL parameter.

- The VOLUME option suspends an XRC volume pair that encounters a device error. XRC continues to process all other volumes in the session. You must correct the error condition and then resynchronize the volume so it is consistent with the other volumes in the session.

  If a volume that is suspended due to an error contains data that is dependent on data on other volumes, data consistency cannot be assured in the event of a disaster. In those cases you should use group_name or SESSION (see below) as the error level specification.

- The SESSION option suspends all volume pairs in an XRC session. You must correct the error and resynchronize all the suspended pairs.

  If a volume with error level SESSION encounters an error, all volumes in the XRC session will be suspended. If a failure occurs on a single LCU, and that LCU attaches to any duplex volume pair with error level SESSION specified, then all volume pairs in the XRC session are put into the suspended state.

  XRC also suspends volume pairs that are in pending state, and the pairs remain where they are in the synchronization process.

- The group_name option allows you to associate any valid TSO name with one or more volume pairs. If a volume with error level group_name encounters an error, XRC will suspend all volume pairs which have the same group name specified.

  An error that occurs on a single LCU that attaches to any duplex volume pair that has group_name specified causes XRC to suspend all volume pairs with the same group name specified.

  Use group_name to associate volumes belonging to an application to ensure consistency in the application.

Table 5-1 summarizes what we have discussed so far on how ERRORLEVEL options determine volume pair suspension in the event of different types of failures.

*Table 5-1 ERRORLEVEL and suspension of volume pairs*

| ERRORLEVEL options | Single volume failures | Primary LCU failures | SDM failures |
|---|---|---|---|
| VOLUME | One duplex volume pair is suspended. | All active volumes on the LCU are suspended. | All duplex volume pairs in the XRC session are suspended. |
| SESSION | All duplex volume pairs in the session are suspended. | All active volumes in the session are suspended. | All duplex volume pairs in the XRC session are suspended. |
| group-name | All duplex volume pairs in the group are suspended. | All active volumes on the LCU are suspended, and all active volumes in this group on any LCU are suspended. | All duplex volume pairs in the XRC session are suspended. |

# 5.17 Capacity and performance

Planning the resources required in an XRC environment is essential for a successful XRC implementation. A smooth data flow is crucial, and in this section we provide some guidance in the capacity and performance planning process.

## 5.17.1 SDM requirements

The SDM manages the XRC activity for all volume pairs in an XRC session.

### Volumes

Up to five XRC sessions are allowed in each z/OS (or OS/390) system image. If you have more then 1500 to 1800 volumes in a single XRC session, you may need to split these into two or more XRC sessions. In addition to the workload balancing split, also note that unless you use CXRC there is no write sequence consistency between XRC sessions, so you must consider splitting the volumes between XRC sessions on an application basis also.

### Virtual storage

XRC operates in up to six address spaces on a single LPAR. ANTAS000 is always active and processes XRC commands. Up to five XRC sessions may be started at a time. Each XRC session that is started on a system occupies its own address space, ANTAS001 through ANTAS005.

The ANTAS000 address space uses 65 MB of virtual storage, which includes a 20 MB trace data space. An ANTASnnn (started when the XSTART command is issued) has the following virtual storage requirements:

- Approximately 20 MB is needed for base code.
- Approximately 30 MB is needed for trace data spaces.
- Volume initialization requires 360 KB for each concurrent volume initialization task.
- About 35 MB for each active XRC storage control session, up to a maximum of 1.7 GB. SDM uses this space for buffers to hold data read from the primary LCU. These data buffers are required from the time when the data is first read from the ESS cache, through the Consistency Group formation process, the journaling process, and the writes to the cache and NVS of the secondary subsystem.

XRC needs a minimal amount of space (0.2 KB) in the extended common service area (ECSA). All other storage used is within the XRC address spaces.

### Real storage

The main real storage requirements for SDM will be for storage buffers used during I/O operations. Those pages will be fixed for the duration of the input or output operation.

The actual real storage required at a given point depends on how much concurrent activity is occurring. You should provide enough real storage on the SDM system to support the projected peak I/O load.

For example, as SDM requires up to 35 MB data buffers from the time the data is read from the ESS cache until the data is written to the secondary volumes, you should plan for a real storage requirement equal to 35 MB times the number of XRC storage control sessions. Part or all of this can be permanently fixed real storage, using the PAGEFIX parameter of the XSET command. All storage beyond the PAGEFIX limit is fixed and freed as required. Specifying a high PAGEFIX value will reduce the MIPS required for fixing and freeing pages for those data buffers.

**Tip:** It is highly recommended that all storage used by the SDM is pagefixed.

### Paging space

The SDM requires paging space to support the address spaces, and you should plan for up to 2 GB for this space, depending on your virtual storage requirements (see “Virtual storage” on page 230). We recommend that you use expanded storage for paging.

### Processor (MIPS) requirements

The processor consumption is usually expressed as the number of MIPS required per 100 writes/sec to primary volumes from primary systems (XRC writes).

This number varies mainly depending on the blocksize and the write I/O rate. With a larger blocksize, the SDM has to move more data, and consequently uses more resources. At a high I/O rate, SDM tends to be more efficient than at a low write rate. There may be some increase in the MIPS requirement when using channel extenders.

For further discussion, refer to the following Web site:

http://www.storage.ibm.com/software/sms/sdm/sdmtech.html

### SMS

If the journal data sets are striped (which we recommend), they must be on SMS managed volumes. The state data set is a PDSE. Journal data set allocation is discussed later in 5.18, “Journal, control, state, and master data sets” on page 233.

## 5.17.2 Primary ESS

You have to assess the capacity of the primary ESS. In addition to reads and writes issued from the primary systems, the ESS must also be able to handle the reads associated with the SDM offloading process. The data written to primary volumes by primary systems will be grouped into record sets in cache, and those record sets will be offloaded by the SDM system with the read record set channel command.

The cache requirement should also be evaluated. There is no separate copy in the cache of the data to be offloaded, the ESS merely builds a logical queue using a directory structure.

When the SDM operation is balanced (meaning that the SDM does not get delayed when retrieving data from the primary ESSs and committing those updates to the journal data sets and secondary volumes) the primary ESS needs little more cache than what is normally required to satisfy primary application performance requirements.

As long as the SDM manages to retrieve the updates before the ESS caching algorithms would have removed them out of cache, minimal additional cache is required.

However, if the primary systems write rate is (temporarily) more than the SDM can absorb, or if there is a (temporary) disruption in the SDM data flow, the cache will be used as a buffer (see 5.14, “Unplanned outage support” on page 225).

Our recommendation is that you evaluate the cache size required to satisfy your primary application needs, and then plan for XRC buffers. This may mean you should install more cache (if possible), or increase your cache-to-backstore ratio by spreading the volumes across multiple ESSs. Your IBM Field Technical Sales Specialist can help you determine the cache needed for XRC by running the Disk Magic modeling tool.

### 5.17.3 Secondary storage subsystem

The secondary storage subsystem can be any subsystem supported by the SDM system, as long as it provides volumes with the same track capacity, the same number of tracks per cylinder, and at least the same capacity as its corresponding primary volume.

If a secondary volume has more capacity than a primary volume, it has to be carefully considered if a copy-back procedure is implemented.

The secondary site needs the same number of XRC volumes as the primary site.

Although there is no requirement to use ESSs as secondary storage subsystems, doing so will give you all the benefits of ESS's performance characteristics. In addition, it also makes it possible to use XRC in a copy-back implementation. With ESSs at the secondary, you can also combine XRC and FlashCopy functions as described in 5.26, “Using XRC with FlashCopy and Tertiary Volumes” on page 271.

Ideally, the number of LCUs at the secondary site should match the number of LCUs at the primary site, in order to provide configuration symmetry.

The secondary site storage controls must be configured to handle all of the primary volume writes plus the I/Os to the journal, control and state data sets as a minimum. In addition they must also be capable of supporting the I/O activity related to the primary application's requirements in a disaster recovery situation.

### 5.17.4 SDM to primary site bandwidth

One of the key elements in the sizing of an XRC configuration is to determine the required bandwidth between the SDM and the primary ESSs.

You could use ESCON or FICON channels to connect the sites if the distance allows it; otherwise, you have to use channel extenders and telecommunication lines.

Regardless of connection technology used, you have to analyze the workload to determine the peak requirements. The peak period should cover a short duration, so that lower workload levels do not lower the average peak workload. Typically, you should analyze on-line requirements during the 5 minutes peak period, and batch requirements during the 15 minutes peak. Taking this approach, you can still benefit from the dynamic workload balancing function in XRC, without really compromising application throughput or response time.

The challenge can be to find the peak intervals, because the period with the highest write activity is not necessarily the period with the highest bandwidth requirement. Typically, the write activity during daytime processing can be high, but this processing uses short write blocksizes. The required bandwidth can therefore be higher during batch processing, even if the write rate is lower, because of the larger blocksizes.

You could use report generator products to identify peaks of write I/Os and data transfer based on information in SMF record type 74. Subtype 1 (device activity) contains information about I/O rates and connect times, and subtype 5 (cache activity) contains information about the number of reads and number of writes per device. Make sure that the known peak days, like the end of month processing, are included in the analysis.

### 5.17.5 Utility device usage

ESS can use either fixed or floating utility devices, as described in 5.9, “Utility devices” on page 218. Our recommendation is to use fixed utility devices.

If you are using floating utility devices and there are many primary volumes in an XRC SC session, there should be a good chance for ESS to find volumes with low utilization to use as a utility device. However, with the possibility in ESS to define small custom volumes, you could define a 1 cylinder volume and add this volume to the SC session with XRCUTL parameter in the XADDPAIR command. Then you know that there will always be one device available to be used as a utility device with no activity and no reserves.

Normally, there will be a relation between the number of XRC primary volumes and the amount of data to be offloaded by the SDM. Consequently, there will be a relation between the number of primary volumes in an ESS LCU and the number of utility devices to be used.

If you are running performance monitors like RMF on the SDM system, you could check the device utilization of the primary volumes during peak offloading periods.

### 5.17.6 XADDPAIR processing

The XADDPAIR command will, unless you specify the NOCOPY parameter, instruct SDM to make the initial copy of the primary volumes to the secondary volumes. XADDPAIR is also used to resynchronize suspended volumes.

XRC allows updates to the primary volumes during this synchronization or resynchronization process, and copies the updates to the secondary volumes.

**Note:** You can use the QUICKCOPY option on the XADDPAIR command to copy only allocated tracks from the primary volume to the secondary. However, as a reserve is issued against the primary volume during initial processing, this option should not be used in a channel extender environment, as if the links fail the reserve may not be released, thus impacting the applications.

Here are some comments on XADDPAIR elapsed times:

- Concurrent active applications will elongate the elapsed time for XADDPAIR operation.
- Concurrent active applications can be negatively impacted (residual counts can grow as well as response times) by the XADDPAIR operation.

SDM does monitor the update and XADDPAIR activity, and attempts to throttle the XADDPAIR activity such that the update activity is minimally impacted. Even though this control is present, it is important to monitor the system activity during XADDPAIR times, and reducing the SCSYNCH and SYNCH parameters may be required. See 5.22.13, “SCSYNCH and SYNCH parameters in XSET command” on page 262.

## 5.18 Journal, control, state, and master data sets

We often call the journal, control, and state data sets for XRC, the *control data sets*.

The control data sets are used by the SDM (except that the journal and control data sets are not used when using the SESSIONTYPE(MIGRATE) parameter in the XSTART command) during normal processing, and have to be allocated and cataloged on volumes that the SDM system can access. The data sets are also used during XRECOVER processing, so the system issuing the XRECOVER command has to have access to those data sets.

The master data set is only used in a CXRC environment. It is not required otherwise.

### 5.18.1 Defining the XRC journal data set

The SDM uses the journal data set(s) as temporary storage, to *harden* a Consistency Group until all of its data is committed to the secondary devices. The bandwidth required to write the Consistency Groups to the journal data sets is the same as the bandwidth required to offload the updates from the primary LCUs.

Journaling is an area that can be easily become a bottleneck and contribute to SDM slowing down. If not properly configured, the result could be that the Consistency Groups would accumulate in SDM's address space, because the journal data sets did not have the necessary performance.

Journal data sets are allocated in pairs (odd/even). When one of the data sets (the active one) of the pair is full, the full data set will become inactive and the paired inactive data set will become active (replacing the just filled or previous active data set). The even/odd pair can therefore be allocated on the same volume because they will never have to compete with each other. SDM assigns a separate internal subtask for each pair of allocated journal data sets, and thereby allows for parallel transfer of CGs.

A maximum of 16 journal data sets can be specified for an XRC session.

- These are the required specifications:
  - Specify the following journal data set name:
    - SYS1.XCOPY.session_id.JRNLnn or `hlq.XCOPY.session_id.JRNLnn`

      Where **nn** has to be **01**,**02**.... up to **16**. If you define journal data sets **01**,**02**,and **04**, for example, XRC ignores **04** because you did not define data set **03**.
  - You have to specify at least two journal data sets.
  - Each journal data set must be a fixed blocked sequential data set with the following attributes:

    DCB=(RECFM=FB,LRECL=7680,BLKSIZE=7680,DSORG=PS)
- These are the recommended specifications:
  - Allocate journal data sets as single extent data sets.
  - Define an even number of journal data sets, preferably four to eight for smaller XRC configurations (for up to about 750 volumes). For large XRC configurations, define from eight to 16 journal data sets.
  - Define the journal data sets as striped data sets.

    When you define journal data sets as striped data sets, you enable the spreading of its I/O operations across multiple volumes and storage controllers.
  - Journal data sets should be large enough to allow the SDM to offload to secondary volumes during peak data load periods (the size of journal data sets is discussed later in this section).
  - Allocate the same amount of space for all journal data sets.
  - Minimize non-XRC activity to journal data set volumes.
  - Do not put journal data sets on the same volumes as the control and state data set.
- For optimal performance, here are some important considerations:
  - Allocate the journal data sets on high performance volumes.
  - Do not put journal data sets behind storage controllers that also process secondary volumes.
  - Spread the striped journal data sets across many disk volumes.

The allocations shown in Figure 5-8 will optimize journal performance for 2, 4, and 8 journal data sets.

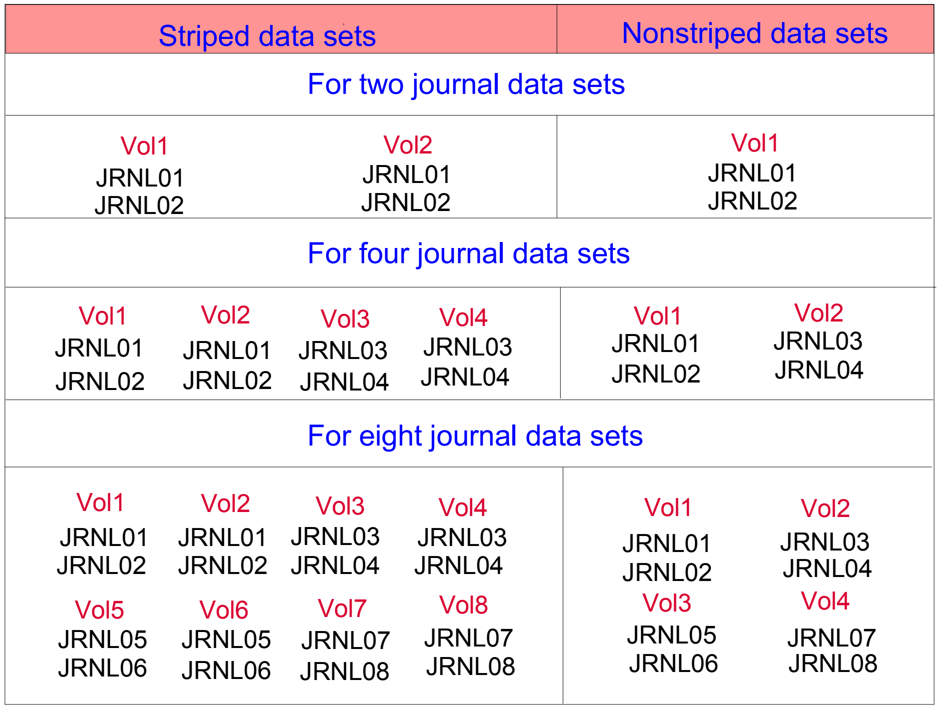


*Figure 5-8 Allocation of journal data sets*

- Size of journal data sets:

  The aggregate, minimum size for the allocated journal data sets should be at least twice the size of the total SDM buffer. For example, if you have 10 XRC storage control sessions running in an SDM, and 576 buffers per storage control session (which is the current default), SDM would have 5760 full track buffers available. This represents the equivalent of 384 cylinders, and so 768 cylinders should be allocated.

  If eight journal data sets are allocated, each with two stripes, then allocate a minimum of 100 cylinders per data set (50 cylinders per stripe).

  > **Tip:** The minimum recommended allocation is 100 cylinders for each journal data set.

Remember that all journal data sets should be of equal size. It is recommended to allocate the maximum journal configuration with 300 cylinders per data set (150 cylinders per stripe) and 16 journals.

## 5.18.2 XRC journal data set: Striped data set definition

Again, we recommend that you define the journal data set as striped data set. In order to use striping, you must request a data class, which specifies a data set name type of Extended.

Extended format data sets, for most purposes, have the same characteristics as physical sequential data sets. Records are not necessarily stored in the same format or order as they appear. An extended format data set can be referred to as a striped data set if its data is interleaved across multiple volumes. This is called sequential data striping.

Sequential data striping is exclusive to system-managed storage. Extended format data sets can be allocated only on system-managed volumes. An extended format data set is not allocated in a storage group containing mixed device types.

For non-guaranteed space requests, if the Sustained Data Rate field is blank or `0`, the target number of stripes is 1 regardless of whether Guaranteed was specified or not. For example, if the storage class specifies `SDR=0` Guaranteed Space and a Volume Count of three is specified, DFSMS will allocate a single-stripe, three volume data set. If the value in the Sustained Data Rate field is greater than 1, it is divided by 4 for 3390s, or by 3 for 3380s, to determine the stripe count.

Thus, for an `SDR` value of 24, a storage group of 3390s would have a stripe count of 6. The volume must be able to satisfy the requested primary space.

Depending on the number of eligible volumes in the candidate storage groups, the actual number of stripes allocated to a data set can be less than the derived one. As the stripe count is decreased, the primary space for each stripe increases, except for the guaranteed space requests.

The `SDR` attribute is ignored if the data set name type, extended required or extended preferred, is not specified in the data class. It is also ignored if you specify Guaranteed Space = YES. In this case, the number of stripes equals the larger of the unit and volume count.

### 5.18.3 XRC Control data set definition

The control data set contains control records indicating:

- The last set of data that was written to the secondary volumes, that is, the last Consistency Group.
- The amount of unwritten data that exists in the journal (this means the amount of data that has been written to the journal as CGs, but has not yet been completely written to the secondary volumes).
- The location of the unwritten data in the journal data sets.

You should use the following data set name:

`SYS1.XCOPY.session_id.CONTROL` or `hlq.XCOPY.session_id.CONTROL`

The data set must be allocated without defining secondary extents, and you should place the data set on a different volume from the journal data sets.

You can allocate the control data set in one of two ways, as a sequential data set or as a PDSE; SDM detects which type of control data set you have allocated, and supports the specified data set type.

- We recommend sequential allocation, this provides much higher performance than the PDSE allocation.

  Use the following attributes:

  ```
  DCB=(RECFM=FB,LRECL=15360,BLKSIZE=15360,DSORG=PS)
  ```

  You should allocate 2 tracks for the sequential control data set.
- You can allocate the control data set as a PDSE, but this is not recommended.

  If you choose PDSE, define the data set with the following attributes:

  ```
  DCB=(RECFM=FB,LRECL=4096,BLKSIZE=4096,DSORG=PO),DSNTYPE=LIBRARY
  ```

  You should allocate approximately 480 KB.

## 5.18.4 XRC state data set definition

The state data set contains the status of the XRC session and the status of the volumes in the session. This data set also contains bit maps of changed tracks for primary volumes. The state data set is updated whenever an XADDPAIR, XDELPAIR, XSUSPEND, XRECOVER or XEND command is issued, or whenever a volume state changes.

You should use the following name for the state data set:

`SYS1.XCOPY.session_id.STATE` or `hlq.XCOPY.session_id.STATE`

The state data set is a PDSE, and you should use the following attributes:

```
DCB=(RECFM=FB,LRECL=4096,BLKSIZE=4096,DSORG=PO), DSNTYPE=LIBRARY
```

Recommended minimum allocation: Allocate 10 tracks for each XRC storage control session.

Allocate one additional track for each volume pair in the session. For volumes that have a larger capacity than a 3390 Model 3 (3339 cylinders), allocate one additional track for every 3000 additional cylinders.

## 5.18.5 XRC master data set definition

All of the XRC sessions coupled in a master session continuously write to and read from the master data set. This access allows communication between hosts as data is copied from the primary site to the recovery site. These are the rules that must be considered:

- Catalog the master data set and make it accessible to each host system that processes a coupled session, as well as to the system that processes the XRECOVER command.
- Note that the master data set can reside on the same volume as the state and control data sets, but not on the same volume that contains the journal data sets.
- Allocate the master data set without defining secondary extents.
- Allocate the master data set as physical sequential and not striped.
- It is essential that you allocate the master data set with contiguous tracks as the I/O is assumed not to span multiple extents. Access to the master data set is controlled using GRS enqueues, and therefore, all systems with coupled SDMs must be in the same GRS configuration.
- Allocate the master data set on a single disk device.
- Note that the required size of the master data set is one cylinder, which allows 14 XRC sessions to be coupled. The required size of the master data set is fixed, which also fixes the maximum number of sessions that can be coupled into one master session.
- Preallocate the master data set size before use of the XCOUPLE command.
- Only the space that is allocated at the time the XCOUPLE ADD command is issued will be available for XRC use.
- Place the master data set in a user catalog that contains only entries for the master data set.
- Allocate the master data set with one cylinder primary space and zero cylinders secondary space as follows:

  DCB=(RECFM=FB,LRECL=15360,BLKSIZE=15360,DSORG=PS),SPACE=(CYLS,(1,0))
- XRC uses the master data set name in one of the following forms:
  - `SYS1.XCOPY.msession_id.MASTER`
  - `mhlq.XCOPY.msession_id.MASTER`

Where:

***msession_id*** Specifies the same name that you issued with the MSESSION parameter of the XCOUPLE ADD command. The name can be up to eight characters long. The name you specify for the msession_id cannot be the same as any session_id.

***mhlq*** Specifies the name of the master data set high-level qualifier that is specified on the `MHLQ` parameter of the `XCOUPLE ADD` command. The `MHLQ` must be one to eight characters, and can be any name acceptable to TSO. SYS1 is the default value.

# 5.19 SDM placement

See Figure 5-9 for an overview of SDM placement in regard to the connections involved.

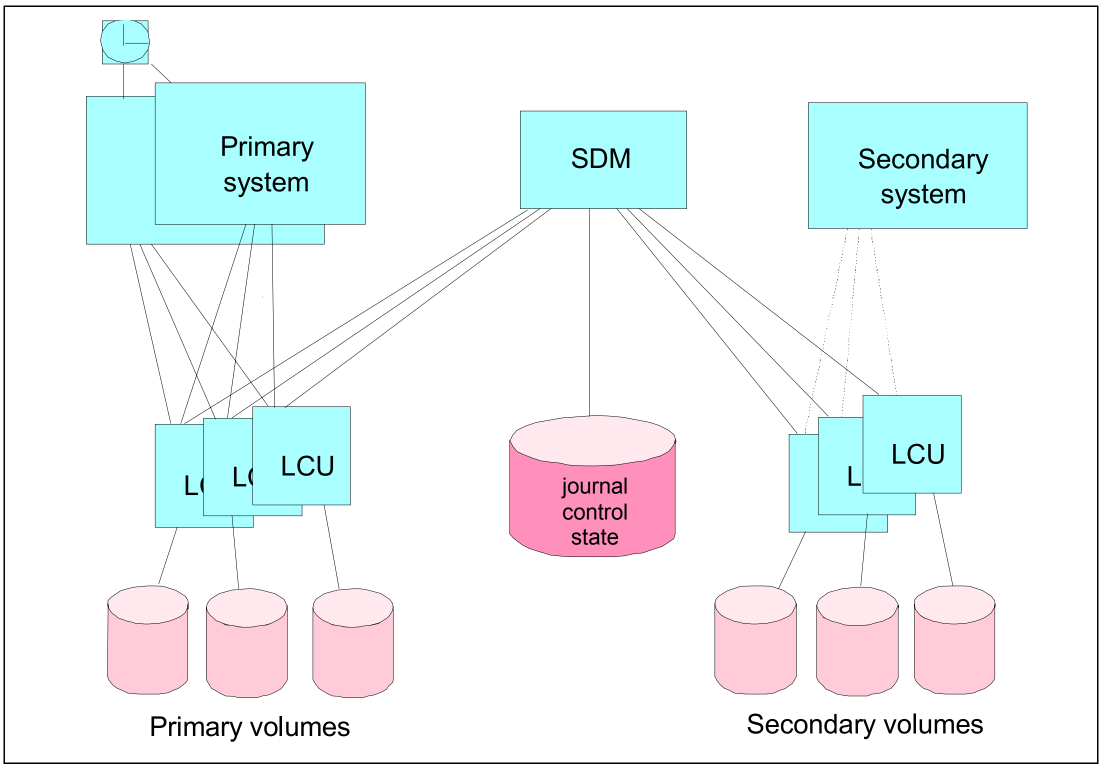



*Figure 5-9 XRC connections*

The SDM can be located at the secondary site, the primary site, or an independent site. Figure 5-9 illustrates the connections required between the components involved in an XRC solution, *independent* of where the SDM is located.

Connectivity requirements may differ between normal XRC operation and recovery operation.

## Normal operation

The connectivity requirements while in normal operation are as follows:

- **Primary systems:** The primary z/OS systems reside on single or multiple CECs, in one or more LPARs, and need only connectivity to primary volumes.

  If the primary systems are running on different CECs, they also need connection to a sysplex timer.

- **Secondary system:** The secondary system does not need to be connected to any other XRC components during normal operation.
- **SDM:** The SDM needs to be connected to all primary volumes, to all secondary volumes and to the journal, control, and state data sets. The journal, control, and state data sets will generally reside at the secondary site.

### Recovery operation

During recovery operation, that is, when executing the XRECOVER command, the SDM only needs access to the secondary volumes and the journal, control, and state data sets. Primary volumes, if connected, should be varied off-line.

After the recovery operation has finished, the secondary system needs connection to secondary volumes.

## 5.19.1 SDM at a secondary site

Connectivity requirements when SDM is located at a secondary site are shown in Figure 5-10. The SDM and the secondary system could run in separate CECs, in separate LPARs on the same CEC or in the same z/OS (or OS/390) image.

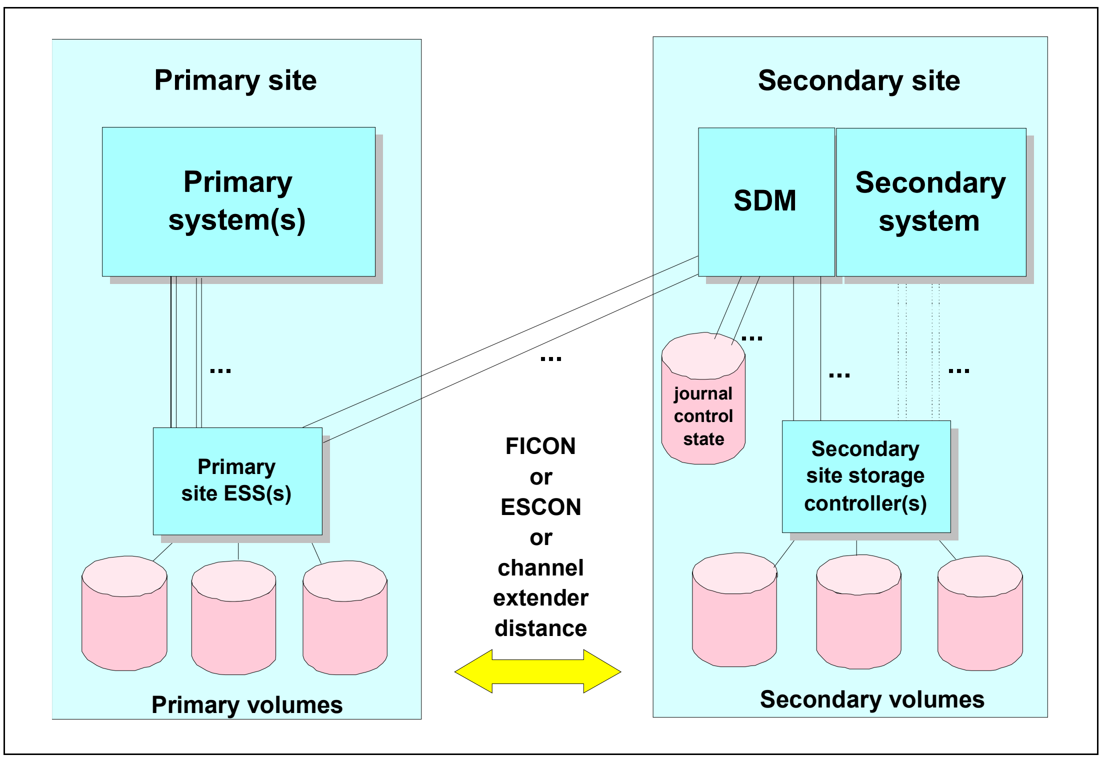



*Figure 5-10 SDM at secondary site*

The advantages we see when locating the SDM at a secondary site are the following:

- Data availability:

  Updated records will undergo three transfers during XRC copying. First, the records are read from the primary site ESS cache and stored in SDM processor storage. Second, they are written from SDM processor storage to the journal data sets. Third, they are written from SDM processor storage to the secondary volumes. The data is considered safe from a primary site outage when it has been journaled at the secondary site.

However, if SDM is located at the secondary site, data that has been read from the primary site ESS cache and stored in SDM buffers is not lost if the primary site subsequently suffers an outage. Although the data has not been hardened on disk, a copy of it still exists in SDM storage, and thus can be journaled from there. If the primary site suffers an outage, only the records that were updated in ESS cache and not read into SDM storage would be lost.

- Performance:

  The primary site ESS and SDM use special communications to transfer data so that multiple records can be read from the ESS to SDM in a single transfer command. However, when writing data to the secondary volumes, normal write chains are used, and this could result in multiple I/Os to multiple volumes.

  It is much more efficient to perform a single data transfer over an extended distance followed by multiple data transfers over shorter distances than it is to perform a single data transfer over a shorter distance followed by multiple data transfers over an extended distance. Thus, performance will be improved if SDM is located at the secondary site because the ESS and SDM special communications eliminate multiple writes over extended distances.

### 5.19.2 SDM at an independent site

SDM can run at a site that is independent of both primary and secondary sites. Locating SDM at an independent site can extend the maximum distance between the primary and recovery sites. This configuration is shown in Figure 5-11.

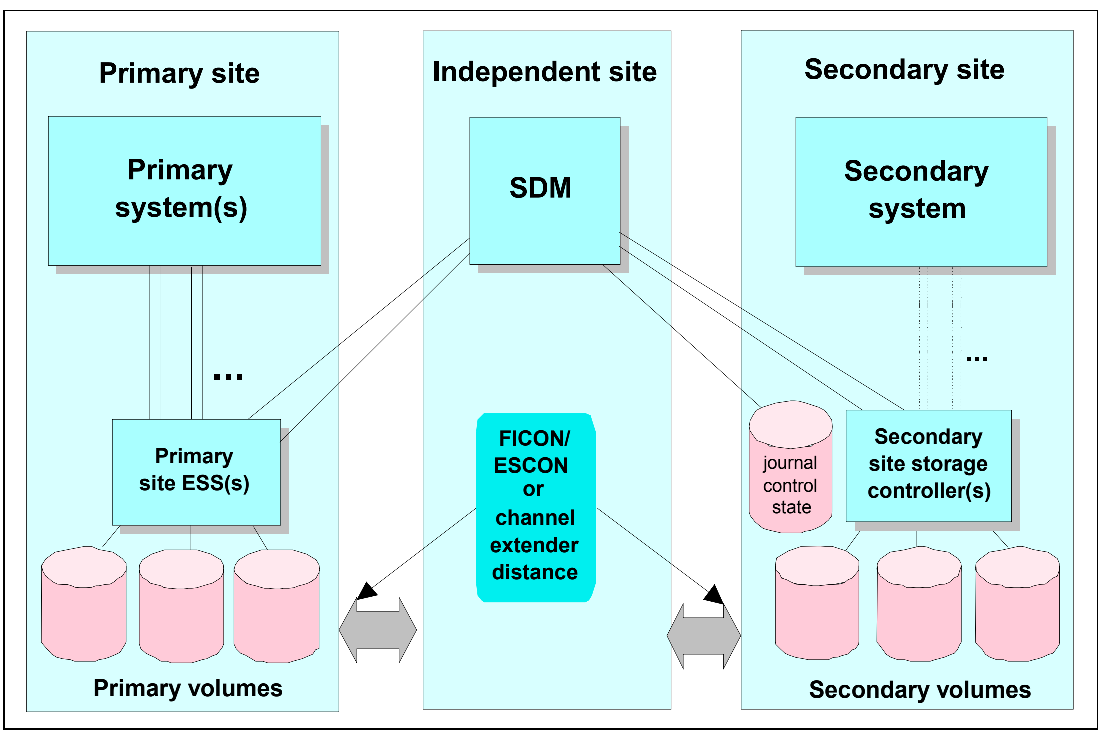



*Figure 5-11 SDM at an independent site*

The following considerations apply when the SDM is located at an independent site:

- Data availability:

  When SDM is physically remote from the application site, records read into SDM storage buffers are not lost if the primary site suffers an outage.

- Performance:

  In this case we may assume that SDM has the dedicated use of processor resources. Thus SDM tasks will always execute with priority, the maximum number of real storage buffers can be defined for SDM operations, and SDM can optimize channel path selection both for reading from the primary site and writing to the secondary site, because there is no contention for channel resource.

  However, an extended distance between the independent site and the secondary site will impact performance for journaling and copying to secondary volumes.

### 5.19.3 SDM at the primary site

The SDM can run in the same z/OS (or OS/390) image as the primary system or in a separate image. The separate image could be in a different LPAR on the same CEC, or on a different CEC.

During a recovery operation, a SDM must be established at the secondary site. This SDM must connect to journal, control, and state data sets, and to all XRC participating secondary volumes in order to do XRC recovery processing.

Connectivity requirements when SDM is located at the primary site are shown in Figure 5-12.

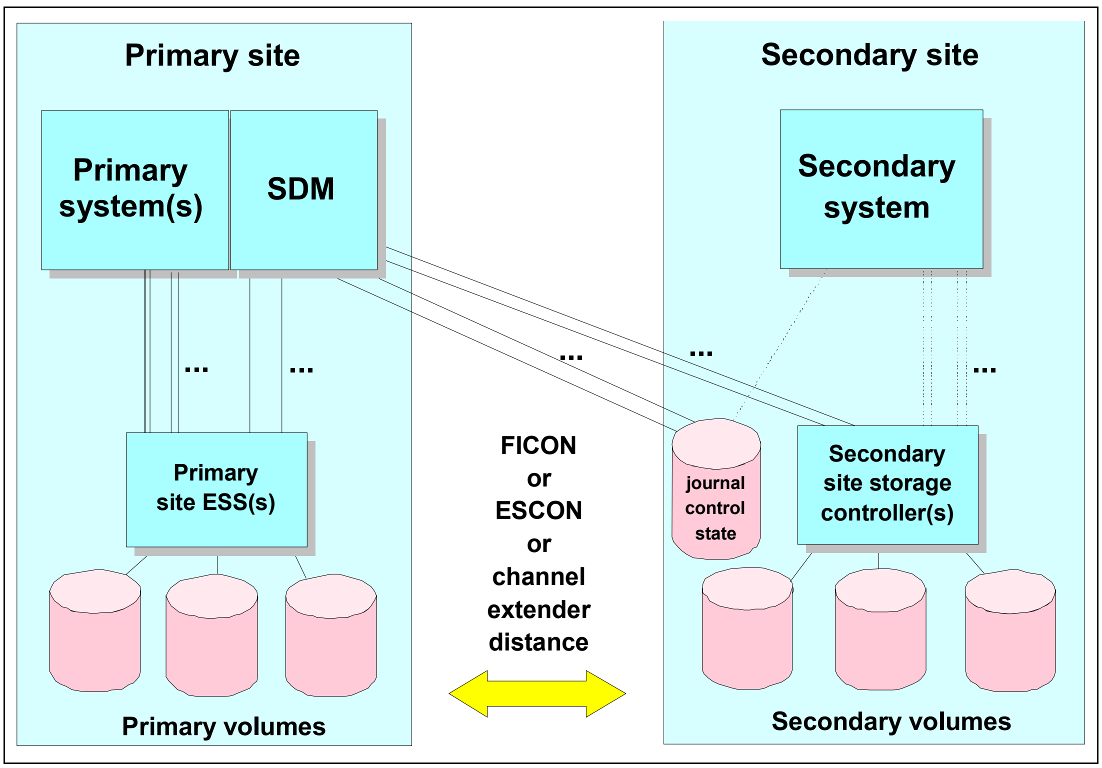



*Figure 5-12 SDM at primary site*

The following considerations apply when the SDM is located at the primary site:

- Data availability:

  When SDM is located at the primary site, updated records read from the primary site ESS are not safe until written to the journal data set at the remote site location. Thus, if the primary site suffers a disaster, record updates in the ESS cache and in SDM storage will be lost.

- Performance:

  The updates read from the primary site ESS will use the special protocol defined for ESS and SDM data transfer, but the journaling and copying to the secondary volumes will incur delays associated with writing over extended distances. And in this case, the updated records have to be transmitted between primary and secondary site twice: once to the journal data sets and another time to the secondary volumes.

- Other considerations:

  To enable recovery, SDM must have access to the journal, state, and control data sets as well as the secondary volumes. If SDM is located at the primary site, which has been rendered inoperative because of the disaster, a second SDM must be started at the recovery site. This is to allow the XRECOVER command to apply any outstanding journal updates to the secondary volumes, and to change the volsers of those volumes.

There may be reasons why you choose to configure SDM at the application site, but the above points should be considered.

# 5.20 Connectivity

This section shows different options for connecting the primary site ESS(s) to the SDM at the secondary site (or an independent site).

## 5.20.1 ESCON channels

Figure 5-13 shows different connection possibilities using ESCON channels. The ESCON architecture allows for a maximum transmission distance of a single link of 3 km with 50 micron multimode (MM) fiber. With the ESCON extended distance feature (XDF) feature, which uses laser transmitters and single mode (SM) fiber, this maximum distance can be extended to 20 km.

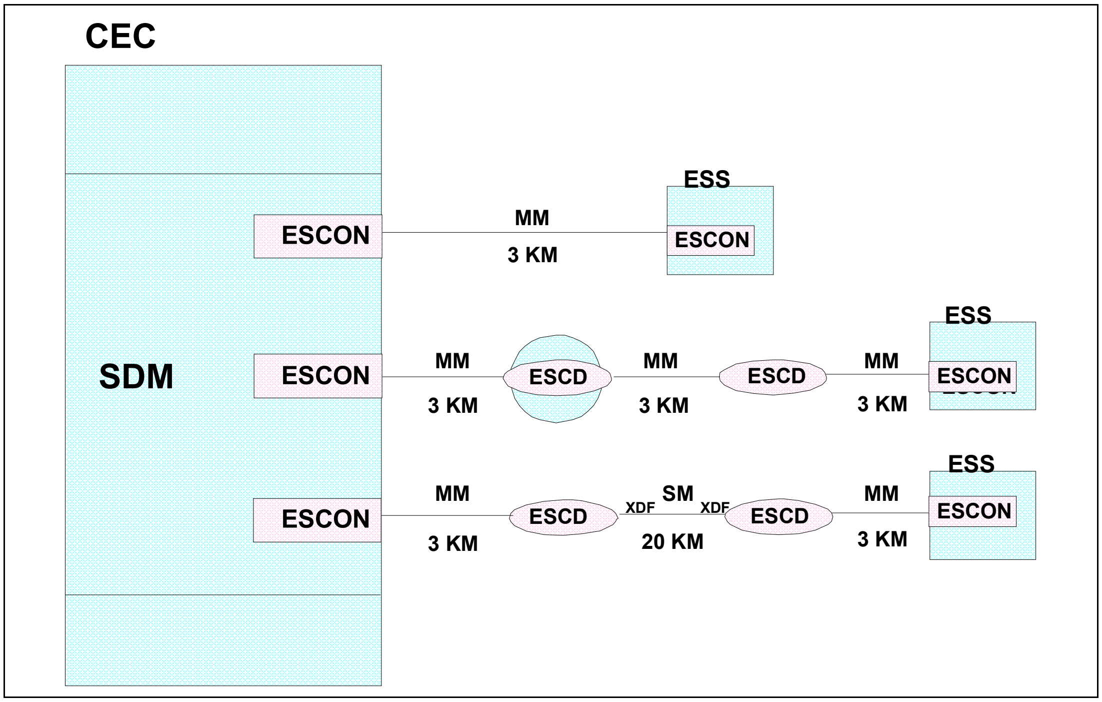



*Figure 5-13 ESCON channels*

ESCON directors are switches for ESCON channels, and can also be used as repeaters to extend the maximum distance. Up to two of them can be cascaded, resulting in a configuration with three connections: CEC to ESCD; ESCD to ESCD; and ESCD to ESS. Each of these can be 3 km apart, or 20 km if using XDF.

A local ESCON connection can have a throughput of up to 17 MB/s, but you must consider that the throughput is a function of distance, and drops with increased distances.

## 5.20.2 FICON channels

The Fiber Connection (FICON) is a high bandwidth channel type that offers more efficient and faster data transfer. Currently, installed single mode and multi mode fiber cables can still be used.

Without repeaters, link distances are up to 10 km for 9-micron single mode fiber, and up to 300 meters and 120 meters for 50 or 62.5 micron multi mode fiber respectively. Each FICON link is capable of supporting the same capacity as up to eight ESCON channels. This enhanced connectivity allows you the option of channel consolidation.

The FICON attachment for ESS is implemented in three ways:

- The first is the FICON Bridge (FCV), in which a FICON channel in a CEC connects to a 9032 Model 5 Director with a bridge card. This enables an existing ESS with ESCON host adapters, to exploit the server FICON channel.
- The second is the native FICON Direct Attachment (FC), where the FICON channel plugs directly into the ESS.
- The last is FICON Switch connectivity. A FICON Switch will enable full dynamic switching of a FICON control unit between multiple channels, as well as multiple FICON control units connected to a single channel.

FICON channels have link rates of 100 MB/s (or 200MB/s with FICON Express), using full-duplex data transfer (it is possible to send data in both directions simultaneously; compared to ESCON, which is a half-duplex protocol). These channels can also implement multiple concurrent I/Os on a single channel.

Data rate droop does not occur with the FICON implementation at the shorter distances as with ESCON. Therefore, FICON is more appropriate for longer distances up to 100 km.

Figure 5-14 shows connection possibilities using FICON channels. Link distances are up to 10 km without repeaters, and maximum end-to-end distance using repeaters is 100 km. These distances can be achieved by cascading two Dense Wave Division Multiplexors (DWDMs) as discussed later in 5.20.3, “Using DWDMs” on page 244.

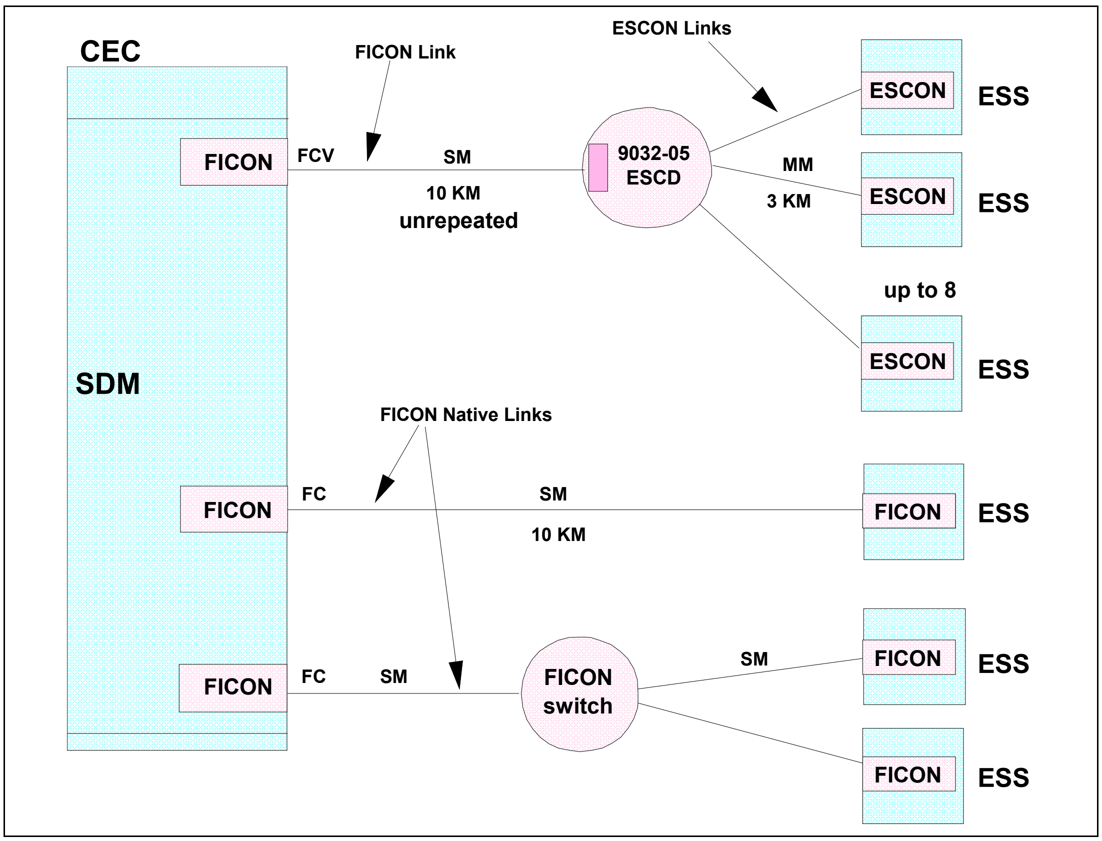



*Figure 5-14 FICON channel connectivity*

## 5.20.3 Using DWDMs

A DWDM is a high speed, high capacity, scalable fiber optic data transport system that uses Dense Wavelength Division Multiplexing (DWDM) technology. DWDM is a technique to transmit several independent bit streams over a single fiber link. It is an approach to opening up the conventional optical fiber bandwidth by breaking it up into many channels, each at a different optical wavelength. Each wavelength can carry a signal at any bit rate less than the upper limit defined by the electronics, typically up to several gigabits per second.

A possible configuration using DWDMs is shown in Figure 5-15. This is a point-to-point configuration with a local site DWDM and a remote site DWDM. Those DWDM units are connected by two pairs of optical fibers, and for availability reasons we recommend that different physical paths are used for each fiber pair. Alternatively, in metropolitan areas, DWDMs may be connected using dark fiber. It is possible to cascade DWDM units, in pairs, to achieve the maximum FICON distance for ESS.

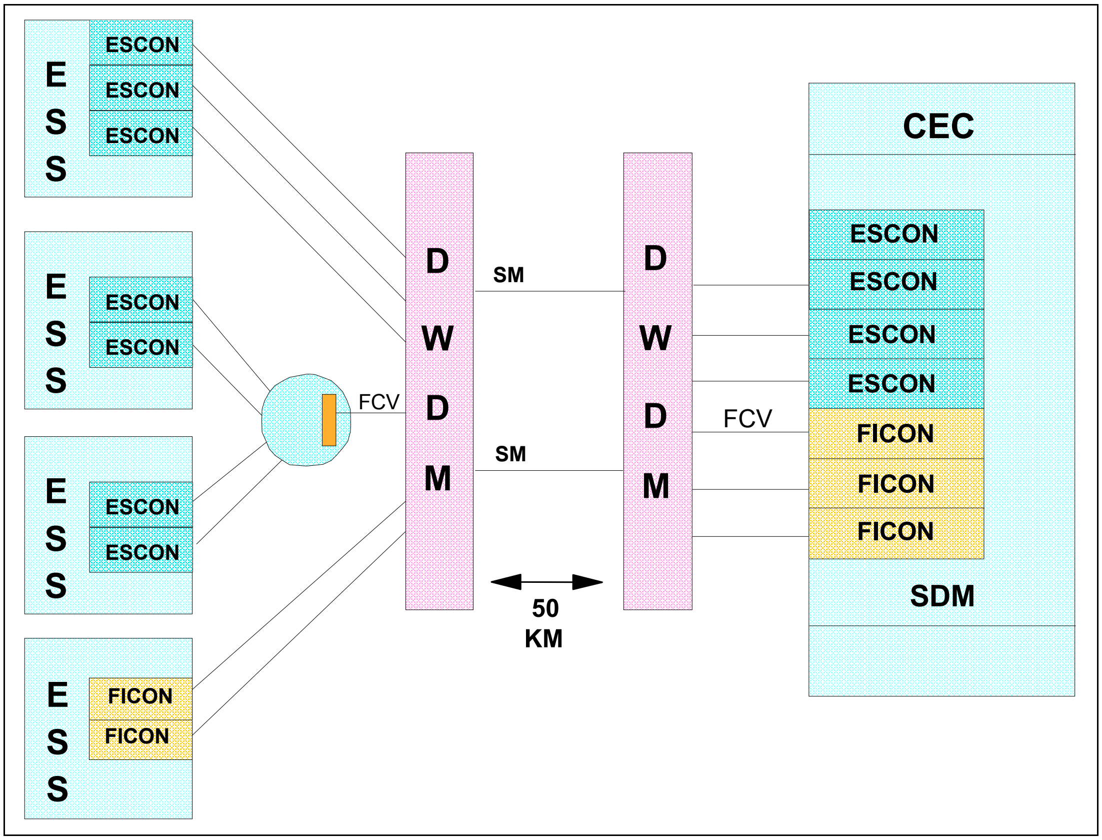



*Figure 5-15 DWDM configuration example*

Note that performance may be affected by the inherent delay caused by signals travelling over long distances, which is a characteristic of the protocol architecture using the DWDM units.

DWDMs provide protocol independence, and a mixture of the following fiber optic data link types may be supported:

- ESCON (both multi mode and XDF single mode links)
- FICON (both FICON native (FC) and FICON bridge (FCV) single mode and multimode links)
- Fibre Channel
- 9037 Sysplex Timer
- Parallel Sysplex Coupling Links
- Asynchronous Transfer Mode (ATM) (both 155 and 622 Mbits/sec)
- Fiber Distributed Data Interchange (FDDI)
- Fast Ethernet
- Gigabit Ethernet (GbE)

Using a DWDM in an ESS XRC configuration can serve two purposes: to extend the ESCON/FICON distance between primary ESS and the SDM at secondary site, and to reduce the number of fiber cables between the sites.

### 5.20.4 Using channel extenders

For XRC, the way to achieve connections beyond ESCON or FICON distances is by using channel extenders. A possible configuration using channel extender technology is shown in Figure 5-16.

IBM does not market this technology directly, but other suppliers are available. The exact configuration and limitations may vary among solutions provided by different suppliers. The channel extender vendor should be contacted for requirements and capabilities.

Figure 5-16 illustrates a configuration with two channel extender units at each site, CE-L (local) 1 and CE-L 2 at the secondary site (local because it is local to SDM and the channels) and corresponding units CE-R(remote) 1 and CE-R 2 at the primary site. The LCUs could be in the same or different ESSs.

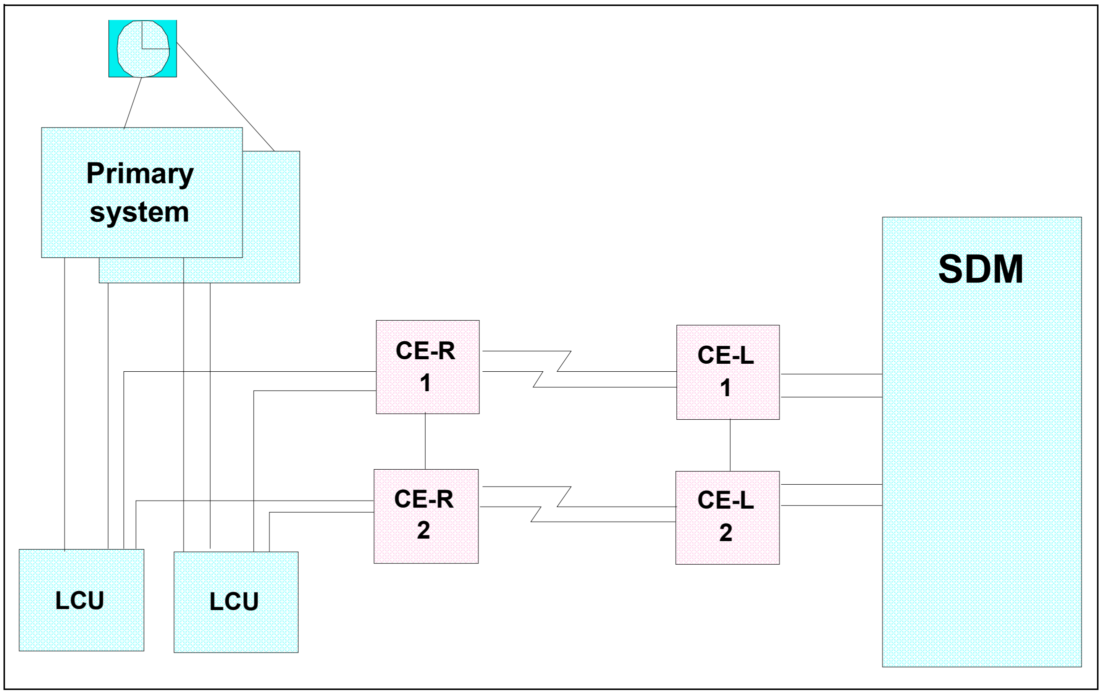



*Figure 5-16 High availability channel extender configuration*

The distances that can be implemented with these types of configurations are only limited by the telecommunication and the channel extender technology capabilities and limitations.

The following notes apply to the example presented in Figure 5-16:

1. When more than one channel extender (CE) unit is installed at a site, they can usually be interconnected with any type of technology supported by the channel extender like ESCON, E3, T3, ATM, FDDI. With this capability, activity can dynamically be routed over the other CE-L/R pair if the links between a CE-L/R pair fail

   In the example in Figure 5-16, if the links between the CE-L/R pair 1 fail, then the connection is dynamically routed using CE-L/R 2.

2. A CE-L/R pair can be interconnected using high speed telecommunication links (for example, T1, DS3, ATM, T3, E3). At least two links should be used between each CE-L/R pair to protect against failure of a link. More links should be added if additional capacity is needed. If there is more than one link between a CE-L/R pair, then those links will usually be exploited as a single resource, that is, during a single data transfer, the channel extender pair will split up the data and transfer an equal part over each link.

3. A channel extender usually does not support dynamic pathing and dynamic reconnect. That is, an I/O issued to a device always uses the same port on the channel extenders, and the I/O has to reconnect through the same port.

   However, you should configure your channel extenders such that each device can be accessed through both CE pairs. Should a CR pair fail, access to the devices would then be retried on the other pair.

# 5.21 XRC management

XRC can be managed either through the set of z/OS (or OS/390) TSO commands, or the application programming interface (API), or through the Remote Copy Management Facility (RCMF) offering.

In this section we are going to give you a description of the TSO commands used to start, control, and recover XRC sessions; what they are used for; and some recommendations regarding use of different parameters.

Like most TSO commands, those commands are deferred requests. The initial message that you receive reports on the validity of the command syntax and that the system has accepted the command for processing. When you specify a userid (MSGROUTEID) in the command, only that ID receives the command processing messages. If this user ID is not logged on, the messages are saved in the TSO BRODCAST data set.

For a more detailed description of the XRC TSO commands, and the required and optional parameters, see *z/OS DFSMS Advanced Copy Services,* SC35-0428.

## 5.21.1 Including XRC TSO commands in procedures

You can include XRC TSO commands in procedures such as REXX CLISTs. Within the procedure, use the MSGROUTID parameter to specify the TSO user ID to which you want XRC to send messages.

XRC routes operational messages to the user ID that issued the command. In order to have these messages appear in the syslog, ensure that the SYS1.PARMLIB CONSOLxx member *permits write to programmer* on the ROUTCODE parameter.

After you issue the XEND, XDELPAIR, and XSUSPEND commands, XRC prompts you (unless you have specified NOVERIFY) to confirm the requests. If those commands are invoked through a CLIST written in REXX, you should either specify NOVERIFY, or place your response to the prompt in the data stack prior to issuing the command. In a batch environment, you can stack the response.

## 5.21.2 Issuing XRC TSO commands

The mode of SDM (see 5.7, “XRC sessions” on page 215) determines which commands you can issue. Table 5-2 lists the XRC TSO commands and indicates for which sessions you can issue them.

*Table 5-2 XRC TSO commands*

| TSO command | This command can be issued when XRC session is: | | | | | | |
|---|---|---|---|---|---|---|---|
| | **Active** | **Inactive** | **Suspended** | **Recovery** | **Coupled Active** | **Coupled Inactive** | **Master** |
| XADDPAIR | Yes | No | No | No | Yes | No | No |
| XADVANCE | Yes | Yes | Yes | No | Yes | Yes | No |
| XCOUPLE ADD | Yes | No | No | No | No | No | No |
| XCOUPLE DELETE | No | No | No | No | Yes | No | No |
| XCOUPLE PURGE | No | No | No | No | No | Yes | No |
| XCOUPLE RELEASE | No | No | No | No | No | No | Yes |
| XDELPAIR | Yes | No | No | No | Yes | No | No |
| XEND | Yes | No | No | No | Yes (1) | No | Yes (2) |
| XQUERY | Yes | No | No | Yes | Yes | No | Yes |
| XQUERY MASTER | No | No | No | No | No | No | Yes |
| XRECOVER | No | Yes | Yes | Yes | No | Yes | No |
| XSET | Yes | No | No | No | Yes | No | No |
| XSTART | No | Yes | Yes | No | No | Yes | No |
| XSUSPEND (Volume) | Yes | No | No | No | Yes | No | Yes (2) |
| XSUSPEND (Session) | Yes | No | No | No | Yes (1) | No | Yes (2) |

Notes:
1. You may issue the XEND and the XSUSPEND commands to a coupled, active session after all volumes have been suspended (AVS) or there are no volumes in the session (NOV).
2. The XEND, XSUSPEND (VOLUME) and XSUSPEND (SESSION) commands issued to a master session will cause the specified command to be coordinated on the member coupled sessions.

## 5.21.3 XSTART command

The XSTART command (Figure 5-17) starts an XRC session by creating the ANTASnnn address space. Volume pairs can now be established. The XSTART command also assigns a name to the session.

The XSTART command is also used to restart a suspended session. When the suspended session becomes active, the suspended volumes are not automatically resynchronized. You have to use the XADDPAIR command to resynchronize the volume pairs within the session.

The XSTART command also validates and opens the state, control, and journal data sets with names that match the session ID when you specify SESSIONTYPE(XRC). When SESSIONTYPE(MIGRATE) is specified, the control and journal data sets are not required.

Figure 5-17 shows a sample XSTART command, where an XRC session is started with the name SBSESS.

```
XSTART SBSESS SESSIONTYPE(XRC) ERRORLEVEL(SESSION)
```

*Figure 5-17 XSTART command*

The SESSIONTYPE parameter can either be XRC or MIGRATE:

- XRC specifies that XRC operate in recovery mode. In this mode XRC makes use of the journal and control data sets to guarantee secondary volume consistency. You must use this mode for disaster recovery solutions.
- In MIGRATE mode, the journal and control data sets are not used. This means that the Consistency Groups are not written to the journal data, and if an error occurs during the write to secondary volumes, XRC cannot guarantee data consistency across the secondary volumes.

  However, the SDM applies updates to the secondary volumes on a Consistency Group basis, and data consistency is maintained on secondary volumes in a controlled environment. In this mode you do not need to allocate the journal and control data sets, but the state data set is required. This mode can be used for migrating data and workload to avoid the I/O activity to journal and control data sets. XRECOVER can be used after ending the session to clip secondary volume serial numbers.

ERRORLEVEL is an important parameter because it specifies how the XRC session responds when an error occurs that causes the session to become unable to process a volume. This may impact the data consistency on secondary volumes. The ERRORLEVEL specified with the XSTART is the default for all volumes in the session, but can be overridden by specifications on the XADDPAIR command. For additional discussion of the ERRORLEVEL parameter, see 5.16.2, “The ERRORLEVEL parameter” on page 228.

## 5.21.4 XADDPAIR command

The XADDPAIR command (Figure 5-18) can be used for:

- Defining new volume pairs to an XRC session. XRC makes an initial copy of the primary volume to the secondary volume unless you specify NOCOPY.
- Adding volume pairs that were previously suspended. XRC automatically resynchronizes suspended pairs and processes them ahead of new volume pairs. XRC will use the hardware bit map in the ESS during the resynchronization process (or the software bit map in the state data set if the primary storage control does not support hardware bit mapping). Instead of specifying the volume pairs by volser, the SUSPENDED parameter can be used to instruct XRC to add all suspended volumes into the session and schedule them for resynchronization.
- Adding a utility device to a storage control session. You specify a utility volume by using XRCUTL instead of secondary volser. XRC does not copy data from a utility volume, and XRC ignores NOCOPY and ERRORLEVEL options.

A maximum of 50 pairs can be specified on a single XADDPAIR command.

XRC allows the primary systems to update the primary volumes while the XADDPAIR copy operation is processing.

```
1. XADDPAIR SBSESS VOLUME(PVOL01,SVOL01) DONOTBLOCK
2. XADDPAIR SBSESS VOLUME(PVOL03,SVOL03) ERRORLEVEL(VOLUME)
3. XADDPAIR SBSESS VOLUME(PMIN01,XRCUTL) SCSESSION(AA)
4. XADDPAIR SBSESS SUSPENDED
5. XADDPAIR SBSESS VOLUME(PVOL03,SVOL03) NOCOPY
6. XADDPAIR SBSESS VOLUME(PVOL04,SVOL04) QUICKCOPY
```

*Figure 5-18 XADDPAIR examples*

Figure 5-18 shows various examples of XADDPAIR commands:

- **Example 1:** DONOTBLOCK

  If the SDM is unable to drain updates from the ESS cache rapidly enough, then the ESS microcode will by default slow down write activity from the primary systems to the primary volumes with high write I/O activity. The DONOTBLOCK parameter specifies that XRC should not block write I/Os issued by the primary systems to the PVOL01 primary volume. See 5.22.11, “DONOTBLOCK parameter in the XADDPAIR command” on page 261 for more information about this parameter.

- **Example 2:** ERRORLEVEL

  This parameter, specified in this case with the XADDPAIR command, overrides the default value specified in the XSTART command. See 5.16.2, “The ERRORLEVEL parameter” on page 228 for more information about the ERRORLEVEL option.

- **Example 3:** XRCUTL

  This specifies that PMIN01 is a utility device, and PMIN01 is not copied to a secondary volume.

- **Example 4:** SCSESSION

  This instructs XRC to assign the volumes (in this case a utility volume) to the storage control session specified (AA). You can assign one or two alphabetic characters for the storage control session value.The SCSESSION designation cannot be changed for suspended volumes.

- **Example 5:** SUSPENDED

  This specifies that XRC should add all suspended volumes back into the session (SBSESS) and schedule them for resynchronization. Suspended volumes keep the same characteristics that existed when the suspension occurred, and you should therefore not use this option to add volumes back into a session if you want to change any of the volume pair characteristics (such as error level, storage control session assignment, or device blocking).

  The SUSPENDED keyword is mutually exclusive with the VOLUME, DONOTBLOCK, NOCOPY, SCSESSION, FULLCOPY, QUICKCOPY, and ERRORLEVEL keywords.

- **Example 6:** NOCOPY

  The NOCOPY option bypasses the initial primary to secondary volume copy; XRC only starts mirroring updates to the primary volume after the volume is added.

  Because of potential data integrity exposures, the NOCOPY option should be used selectively, and only when installation procedures can guarantee that the primary and secondary volumes are exact copies (excluding volsers) at the time the command is issued.

- **Example 7:** QUICKCOPY

  The QUICKCOPY option specifies that only *allocated* tracks will be copied onto the secondary volume when the pair is being established.

XRC ignores QUICKCOPY when establishing utility volumes and when resynchronizing volumes.

To ensure data integrity, the initial processing for QUICKCOPY must issue a reserve and then a release for the primary volume during the initial phase of the synchronization process. If access to the primary volume is through a channel extender, and the connection fails while XRC has the volume reserved, applications at the primary site will not be able to access the primary volume.

### 5.21.5 XQUERY command

The XQUERY command (Figure 5-19) is the primary tool to request information about the XRC environment. XQUERY provides status at the session and volume level for all primary and secondary volume pairs.

Parameters are available to customize the reports, both regarding the type of information and the level of detail. The following are examples of reports available by selecting XQUERY command parameter options:

- XRC session statistics report
- XRC volume pair reporting
- XRC session default setting report
- XRC related storage control statistics report
- XRC related configuration report

The data consistent time in the XQUERY report is the consistency time for volumes in the session at the time the report is given. The timestamp variable specifies the UTC time from the application system. XRC generates this time from the updates received from the application system. Data on the secondary volumes was last consistent up to this timestamp.

XQUERY also reports a delay time. This is the current delay between the last application I/O to a primary LCU, and the update made by the data mover to a secondary volume. This field provides an approximation of the time delay between the primary and secondary volumes.

Output from XQUERY directed to the system log has `ANTQxxxx` message identifiers, and *z/OS MVS System Messages, Volume 1 (ABA-AOM),* SA22-7631 contains more explanations about XQUERY output.

XQUERY output can be directed to a sequential data set, and our recommendation is that you have the command automated, and that you direct the output to a data set for further processing.

The sample command shown in Figure 5-19 will direct the XQUERY output to data set XQUERY.OUT, and this output will be appended to this existing data set. Only the volumes from session SBSESS, which match the serial number prefix PVOL will be reported.

```
XQUERY SBSESS VOLUME(PVOL*) dataset(XQUERY.OUT) DISP(MOD)
```

*Figure 5-19 XQUERY example*

#### XQUERY MASTER in a coupled environment

You can issue the XQUERY MASTER command (Figure 5-20) to a master session to display information about coupled XRC sessions associated with the master session, including the XRC session name, session status, command pending status, journal delta time, RCV/ADV delta time, master session status, master session recoverable time, and master pending command.

Figure 5-20 shows an example of a command where the master session name is **HOSPITAL**, and the high level qualifier of the master session control data set is **SYS2**.

```
XQUERY HOSPITAL MASTER MHLQ(SYS2) dataset(XQUERY.OUTPUT2) DISP(OLD)
```

*Figure 5-20 XQUERY MASTER example*

## 5.21.6 XDELPAIR command

The XDELPAIR command (Figure 5-21) directs the XRC session to stop processing a primary volume and its secondary volume. For each volume, the XDELPAIR command returns a timestamp indicating that the data on the secondary volume is consistent with the data that was on the primary at that time.

You can delete any XRC volume pair with the XDELPAIR command, regardless of the status of the pair. Volume pairs in pending, duplex, copy, seqcheck, and suspend status all become inactive when deleted, and are no longer eligible for recovery.

You may delete up to 100 volume pairs in a single XDELPAIR command. Note that XRC does not process any volumes if there is a command syntax error, or if any of the volumes in the list are not part of the XRC session.

```
1. XDELPAIR SBSESS VOLUME(PVOL01 PVOL02 PVOL03) DRAIN
2. XDELPAIR SBSESS VOLUME(PVOL02) CANCEL
3. XDELPAIR SBSESS VOLUME(ALL)
```

*Figure 5-21 XDELPAIR examples*

Figure 5-21 shows various examples of XDELPAIR commands:

- **Example 1:** As an optional parameter you can specify when the pairs are going to be deleted, by specifying IMMEDIATE, DRAIN, or ATTIME:

  **IMMEDIATE:** This is the default. IMMEDIATE specifies that XRC immediately process the delete request. XRC deletes the volume pair or pairs when the current Consistency Group has been applied. The secondary volumes are consistent up to the UTC timestamp that is reported on the successful completion of the command. This is the only option that can be used for suspended volumes.

  **DRAIN:** The DRAIN function ensures that all timestamped record updates that the SDM has not yet written to the secondary storage control are drained from the primary storage control, and applied to the secondary volume or volumes. All secondary volumes are consistent up to the UTC timestamp that is reported by successful completion of the command.

  This option could be used whenever you have quiesced the application activity on the primary volumes and to ensure that XRC has applied all the updates to the secondary volumes before the volume pairs are deleted.

  **ATTIME:** This option specifies that the XRC will delete the specified volume pairs after applying all updates, up to the specified time, to the secondary volumes. XRC will not apply any updates beyond the specified UTC timestamp to the secondary volumes. All affected secondary volumes are consistent up to the UTC timestamp that is reported on successful completion of the command.

  You could use this option in environments where applications continue to update the primary volumes even after the time specified on the ATTIME keyword has expired.

- **Example 2:** CANCEL

  Specifies that XRC cancel the previous XDELPAIR command with an ATTIME or DRAIN request. The cancel option is only valid when there is a pending XDELPAIR command.

- **Example 3:** VOLUME(ALL)

  Specifies that XRC should delete all eligible volumes in the session, including utility volumes. The XRC session will still be active.

## 5.21.7 XEND command

Use the XEND command (Figure 5-22) to stop all XRC activity to active volumes and to end the XRC session. Issuing the XEND command stops all updates to the secondary volumes at the specified time.

The XEND command returns a timestamp indicating that the data on the secondary volumes of all active pairs is consistent with the corresponding data that was on the primary volumes at the specified time.

In a coupled environment you can issue the XEND command to a master session to have all sessions end with their volumes consistent to the same time.

With XEND you have the same optional parameters as XDELPAIR to specify when the session should end and how the updates should be applied to secondary volumes (ATTIME, DRAIN, IMMEDIATE and CANCEL). See 5.21.6, "XDELPAIR command" on page 252 for further description of those parameters.

```
XEND SBSESS
```

*Figure 5-22 XEND example*

The example in Figure 5-22 ends session **SBSESS** and all its associated volumes. IMMEDIATE is the default.

## 5.21.8 XSUSPEND command

Use the XSUSPEND command (Figure 5-23) to suspend either:

- An XRC session
- One or more volume pairs from the XRC session

When an XRC session is suspended, the ANTASnnn address space is ended. The system data mover is no longer active and the volume pairs are in suspended state. You should use the XSUSPEND command to suspend a session as part of a planned outage when you stop applications or intend the outage to be brief.

In a coupled environment, you may either suspend all XRC sessions coupled to the master session, or suspend all volume pairs from sessions that are coupled to the master session.

The SDM remains active when volume pairs are suspended. SDM continues to read record updates from the primary LCUs, and XRC will use the ESS hardware bit map during resynchronization.

As ESS supports hardware bit maps, the ESS will continue to record changed tracks, and there will be no impact to the primary systems. XRC will use this bit map to resynchronize the volumes when the session is restarted, and volumes added back to the session with XADDPAIR command.

```
1. XSUSPEND SBSESS VOLUME(PVOL03) DRAIN
2. XSUSPEND SBSESS VOLUME(ALL)
3. XSUSPEND SBSESS TIMEOUT(01:15:00)
```

*Figure 5-23 XSUSPEND examples*

Figure 5-23 shows various examples of XSUSPEND commands:

- **Example 1:**

  The VOLUME parameter indicates that this is a volume suspension. The VOLUME parameter is mutually exclusive with the TIMEOUT parameter.

  The IMMEDIATE, ATTIME, and DRAIN keywords allow you to control when the suspension occurs. For more information about those parameters; see 5.21.6, "XDELPAIR command" on page 252.

- **Example 2:**

  When VOLUME(ALL) is specified, XRC suspends all volumes in the session, but does not suspend utility devices.

- **Example 3:**

  TIMEOUT indicates that the entire XRC session should be suspended, and the ANTASnnn address space ended. The value specified in the TIMEOUT parameter specifies the maximum time that the storage control can wait for the XRC session to restart. For more information about this TIMEOUT parameter and the TIMEOUT value, see 5.22.9, "TIMEOUT parameter in XSUSPEND command" on page 260, and 5.13, "Planned outage support" on page 224.

  The TIMEOUT parameter is mutually exclusive with the VOLUME parameter.

### 5.21.9 XSET command

You can use the XSET command (Figure 5-24) to change the characteristics of an active session without requiring that you quiesce or restart the session. The XSET command enhances serviceability and availability by allowing the storage administrator to dynamically change the XRC session tuning values. XRC saves the values in the state data set and uses them to suspend or restart an XRC session.

The XSET command changes the characteristics of the XRC session. The changes made with the XSET command remain in effect for the duration of the session, even through session suspension and restart, unless changed with another XSET command.

```
1. XSET SBSESS PAGEFIX(140) PRIORITY(LOAD)
2. XSET SBSESS SCSYNCH(2,2) SYNCH(4)
3. XSET SBSESS RTRACKS(8000)
4. XSET SBSESS TIMEOUT(00:00:45) SSID(ALL)
5. XSET SBSESS UTILITY(FIX)
```

*Figure 5-24 XSET examples*

Figure 5-24 shows various examples of XSET commands:

- **Example 1:** PAGEFIX and PROIORITY

  PAGEFIX specifies the number of megabytes that are assigned to the SDM as permanently fixed real storage. The value specified here may impact SDM's performance, and this parameter is discussed in 5.22.7, “PAGEFIX parameter in XSET command” on page 260.

The PRIORITY parameter specifies the priority algorithm used by XRC when volumes are synchronized or resynchronized with the XADDPAIR command. This parameter is discussed in more detail in 5.22.12, “PRIORITY parameter in the XSET command” on page 261.

- **Example 2:** SCSYNCH and SYNCH

  The SCSYNCH and SYNCH parameters control the synchronization and resynchronization process in XRC. SCSYNCH specifies the maximum number of volume pairs processed concurrently per LCU, while SYNCH specifies the maximum number of pairs processed concurrently in the XRC session. More information about those parameters are in 5.22.13, “SCSYNCH and SYNCH parameters in XSET command” on page 262.

- **Example 3:** RTRACKS

  The RTRACKS or RFREQUENCY parameters can be used to control when ESS or SDM should reset resynchronization bit maps. More information is in 5.22.10, “RFREQUENCY/RTRACKS parameters in XSET command” on page 260.

- **Example 4:** TIMEOUT

  This TIMEOUT parameter can be used to specify the acceptable primary application impact for *unplanned outage* SDM processing. For comparison, the TIMEOUT parameter in the XSUSPEND command can be used to specify acceptable application impact for a planned outage. This TIMEOUT parameter is discussed in 5.22.8, “TIMEOUT parameter in XSET command” on page 260.

- **Example 5:** UTILITY

  The UTILITY parameter can be used to specify how the utility device is selected for reading data from the primary storage control - either FIX or FLOAT. See “Utility device selection method” on page 218 for further information.

## 5.21.10 XRECOVER command

The XRECOVER command (Figure 5-25) will bring data on the secondary volumes to a consistent, recoverable state where they are equivalent to the primary volumes, as of the given timestamp.

If the session was started with SESSIONTYPE(XRC), the XRC recovery function must continue to have access to the appropriate journal, control, and state data sets that were in use at the time of the failure, or when the XRC session ended. In this case XRECOVER will add valid, non-applied journal data to the secondary volumes. For SESSIONTYPE(MIGRATE), only the state data set is required. In both cases will XRECOVER label the secondary volumes with the volume serial numbers of the primary volumes.

If the XRC session has either been suspended or ended normally by a command, the XRECOVER command reports the timestamp that all data is consistent up to. The XRECOVER command will also automatically generate an XQUERY storage control report. The report includes the last applied timestamp that defines the recovered, consistent state for all volumes.

You should not issue the XRECOVER command to an active XRC session. If a session is active when you issue the XRECOVER command, XRC rejects the command. You must first issue an XSUSPEND or XEND command for the session, and then make all primary volumes off-line to the SDM system.

You can also issue the XRECOVER command to recover a group of interlocked coupled sessions to the same consistency time. In order to provide master session consistency, CXRC applies updates to volumes on the sessions that are behind to allow them to advance to the forward session. CXRC will forward-recover volumes that are suspended because of an error, and that have the necessary journal data available. It will not forward-recover volumes that are suspended because of an XSUSPEND command.

```
XRECOVER SBSESS
```

*Figure 5-25 XRECOVER example*

Figure 5-25 shows an example of a XRECOVER command for bringing the data of session SBSESS to a consistent and recoverable state. You have to specify the session_name that was previously specified on the XSTART command.

### 5.21.11 XADVANCE command

You can issue the XADVANCE TSO command (Figure 5-26) to both coupled and noncoupled sessions.

The XADVANCE command differs from the XRECOVER command in that it obtains a consistent set of secondary volumes without updating the secondary volume VOLSERs. This makes it easier for you to create a time-consistent backup of the secondary volumes before restarting XRC and resynchronizing volumes.

The XADVANCE command requires all volumes in the session to be suspended. It may be used with an active or an inactive session. You cannot use this command when the session is active on another system, or if the session has any non-suspended volumes. The XADVANCE command does not apply any data to volumes that were suspended because of a command, or that were suspended because of errors that occurred before the last Consistency Group.

If an error occurs while writing to a secondary volume that results in all volumes being suspended, that volume will be suspended at a different consistency time than the other volumes in the session. To resolve the time discrepancy, correct the error on the secondary volume, and issue the XADVANCE command to update that volume to the same consistency time as other volumes in the session.

Use the XADVANCE TSO command to update secondary volumes of a session to a time that is consistent with the volumes of other sessions that are coupled to the master session. The XADVANCE command applies secondary volume updates to the minimum master journal time or the pending master command time. This is the same time to which an XRECOVER command would recover the session.

The XADVANCE command produces an XQUERY report that displays YES in the ADV column to indicate if each volume in the session was eligible to have updates applied. A NO in the column indicates that the volume was not eligible to have updates applied. The report also displays NO for volumes that have not reached duplex, that were suspended through a command, or that were suspended at a time before the consistency time that exists as of the beginning of the XADVANCE operation.

Issuing an XADVANCE command to a master session automatically generates an XQUERY MASTER report for the coupled sessions.

```
XADVANCE UMC HLQ(SYS1) MSGROUTEID(OPER3)
```

*Figure 5-26 XADVANCE example*

Figure 5-26 shows an example of an XADVANCE command for session UMC. SYS1 is the high level qualifier of the journal, state, and control data sets that were used with this session.

### 5.21.12 XCOUPLE command

Use the XCOUPLE TSO command (Figure 5-27) to add XRC sessions to, or delete XRC sessions from, a master session. You can also use this command to manage the status of the master session. Specifically:

- The XCOUPLE(ADD) command couples the specified XRC session into a master session, which is then used to coordinate the consistency time and data recovery of multiple XRC sessions. When two or more XRC sessions are coupled, the journal and consistency times of the sessions are maintained such that the entire group of sessions can always be processed by the XRECOVER or XADVANCE commands to a consistent point-in-time. The XCOUPLE(ADD) command is not allowed for sessions where you specified SESSIONTYPE(MIGRATE) on the XSTART command.
- The XCOUPLE(DELETE) command uncouples the specified active XRC session from the master session. In a coupled environment, when the last volume is deleted, the system will automatically perform the XCOUPLE DELETE command to remove the session from the master session.
- The XCOUPLE(PURGE) command is issued to an inactive coupled session. If the master session is active, this command clears information about the coupled XRC session from the master data set.
- The XCOUPLE(RELEASE) command removes the HOLD status of the master session. Use the RELEASE option only if you no longer need to protect the recoverable state of the coupled XRC sessions in the master session and are ready to resume normal remote copy operations.

```
1. XCOUPLE STJOSEPH ADD MSESSION(HOSPITAL) MHLQ(MSTRHOSP)
2. XCOUPLE GENERAL DELETE MSGROUTEID(OPER2)
3. XCOUPLE STJOSEPH PURGE MSESSION(HOSPITAL) MHLQ(MSTRHOSP) HLQ(STJOS1)
4. XCOUPLE HOSPITAL RELEASE MHLQ(MSTRHOSP)
```

*Figure 5-27 XCOUPLE examples*

Figure 5-27 shows various examples of XCOUPLE commands:

- **Example 1:** Couple XRC session STJOSEPH to master session HOSPITAL, whose master session control data set has the high level qualifier MSTRHOSP.
- **Example 2:** Delete XRC session GENERAL from the master session to which it is coupled.
- **Example 3:** Remove information about the inactive XRC session STJOSEPH from the master session.
- **Example 4:** Indicates that all manual, operator-initiated actions have been performed to protect the recoverable status of the coupled XRC sessions associated with master session HOSPITAL. The HOLD status of the master session is to be removed.

### 5.21.13 ANTRQST application programming interface

The ANTRQST macro provides an application program call to the z/OS system data mover's application programming interface (API). This macro allows you to call XRC, PPRC, FlashCopy, and SnapShot copy functions. You may need to ensure that necessary address spaces such as ANTAS000, ANTASnnn, or ANTMAIN are active when you execute the macro.

The ANTRQST macro is described in detail in *z/OS DFSMSdfp Advanced Services*, SC26-7400.

## 5.22 XRC tuning parameters

This section discusses the tuning parameters available for XRC. For additional information on the topics discussed in the present section, the following IBM publication can be referred to: *z/OS DFSMS Advanced Copy Services,* SC35-0428.

XRC offers the flexibility of allowing SDM operations to be tailored to installation requirements. XRC supports the modification of key parameters, either from a PARMLIB data set or through the XSET command.

### 5.22.1 Creating a PARMLIB data set

The main purpose of creating an XRC PARMLIB data set is to provide a single method to specify parameters for each system in a multisystem environment, without modifying the ANTXIN00 member of SYS1.PARMLIB. XRC uses a PARMLIB data set name in one of the following forms: `hlq.XCOPY.PARMLIB` or `SYS1.XCOPY.PARMLIB`.

### 5.22.2 PARMLIB members

When XRC first starts up using the XSTART command, it searches for member ANTXIN00 in SYS1.PARMLIB. When the value in the ANTXIN00 member differs from the value specified in the XSTART command, the value in the XSTART command always overrides the value found in ANTXIN00. After processing the ANTXIN00 member, XRC then looks for member ALL in the data set called *hlq*.XCOPY.PARMLIB, where hlq is the value found for the hlq parameter in the ANTXIN00 member, or the value entered in the XSTART command. The ALL member can contain values that you want to be common across all logical sessions on a system. hlq.XCOPY.PARMLIB must be allocated as RECFM fixed, with LRECL=80 and DSORG=PO. Finally, XRC reads *hlq*.XCOPY.PARMLIB for a member whose name matches the session ID of the session.

### 5.22.3 XSET PARMLIB

To invoke PARMLIB support at times other than XRC start up, you can issue the XSET PARMLIB command. You can use the XSET PARMLIB both before and after you issue an XSTART command. If XSET PARMLIB is invoked before an XSTART command, you can check the validity of the parameter syntax without applying any of the parameters.

### 5.22.4 PARMLIB parameters

PARMLIB parameters use syntax similar to that of TSO commands. Parameters consists of a category name, parameter names, and values. Refer to the IBM publication *z/OS DFSMS Advanced Copy Services,* SC35-0428 for the table describing possible PARMLIB parameters, and their associated XSET parameter, if applicable.

To determine the current settings for the PARMLIB parameter values, issue the XQUERY ENVIRONMENT(PARM) command. To find the default values for the parameters, issue the XQUERY ENVIRONMENT(PARM) command to an inactive session.

## 5.22.5 Description and use of flags

Flags are diagnostic tools that perform a specific action if the triggering event occurs. Users turn these flags on and off according to their preferred response to the triggering event. You can determine the current settings for each indicated flag by giving the MODIFY command without specifying ON or OFF. To determine the default values for all flag names at once, issue the XQUERY ENVIRONMENT(FLAG) command to an inactive session. This displays the values in the control address space (ANTAS000).

## 5.22.6 PARMLIB example

The following is an example of member ZOWIE, which was created in PARMLIB *hlq*.XCOPY.PARMLIB for use when starting XRC session ZOWIE.

```
/* */
/*ZOWIE */
/* */
/*This is an initialization parmlib member for XRC logical session */
/*ZOWIE.*/
/* */
/*Change Activity:*/
/**/
/*$L0=OW52938,HDZ11E0,010630,TUCRNC:XRC parmlib support */
/**/
VOLINIT -
MaxNumberInitializations(3)-
InitializationsPerPrimary(1)-
InitializationsPerSecondary(1)-
TracksPerRead(2)-
TracksPerWrite(2)
SHADOW -
StorageControlTimeout(00.04.05)-
MaxBytesTransferred(750001)-
MaxTracksUpdated(2)-
MaxTracksFormated(2)
BITMAP -
DelayTime(00.02.05)-
ChangedTracks(800)
FLAG -
Name(Abend_Lic)-
Action(On)
FLAG -
Name(SCTRAP)-
Action(On)
```

*Figure 5-28 PARMLIB example*

Refer to *z/OS DFSMS Advanced Copy Services,* SC35-0428 for a description of the parameters contained in this example.

### 5.22.7 PAGEFIX parameter in XSET command

This parameter specifies the number of megabytes that are assigned to the SDM as permanently page-fixed real storage. This is the maximum amount of storage that remains page-fixed for the duration of the XRC session. The default PAGEFIX value is 8 MB.

Each SC session requires up to 35 MB for its data buffers. If you have *n* SC sessions running concurrently in the SDM address space, up to *n* x 35 MB data buffers will be required. For example, with 6 primary LCUs and 2 SC sessions for each LCU, up to 420 MB data buffers will be required. If you can page-fix most or all of those data buffers, this will minimize the required MIPS and maximize the potential throughput. The SDM will fix and free any amount of additional real storage required.

Changes specified with the PAGEFIX parameter take place when the next SC buffers are processed.

### 5.22.8 TIMEOUT parameter in XSET command

This parameter specifies the maximum primary system impact that the XRC session will allow. In 5.14, "Unplanned outage support" on page 225 there is a detailed description of how this parameter is used. You should set this parameter to a value that is acceptable to your application production environment.

We recommend that you do not set this value lower than the MIH value for the primary volumes. The recommended MIH time for the ESS is 30 seconds.

If you are using channel extenders between the primary site and the SDM location, you should not set the TIMEOUT value lower than the channel extender value.

You can specify the subsystem id (SSID) to which the TIMEOUT value will apply. Each LCU in an ESS has a unique SSID, which is set during the installation and configuration process. If you do not specify any SSID, the TIMEOUT value applies to all LCUs in the XRC session.

You should not confuse this TIMEOUT parameter with the TIMEOUT parameter in the XSUSPEND command. These parameters are used for different purposes.

### 5.22.9 TIMEOUT parameter in XSUSPEND command

When you suspend a session with the XSUSPEND command, you must specify a TIMEOUT value (or you can specify that the LCUs default value set in the LCU is to be used, which is normally set to 5 minutes). The SDM communicates this value to all primary LCUs.

This value specifies the maximum time that a LCU can wait for the XRC session to be restarted with an XSTART command. How this parameter is used is discussed in 5.13, "Planned outage support" on page 224.

### 5.22.10 RFREQUENCY/RTRACKS parameters in XSET command

For ESSs with hardware bit map support, a hardware bit map in the ESS will continuously be updated to reflect changed tracks on primary volumes. This bit map will be used to add suspended volumes back to a session, and by this avoid a full volume synchronization.

SDM will also continuously maintain a software bit map as long as the ANTASnnn address space is active. This bit map is kept in the state data set and, with XRC Version 2, was used during resynchronization of a suspended volume. If the ANTASnnn address space is not active, or the data path between the SDM and the primary LCU is broken, SDM is unable to update this bit map. Unplanned outages were therefore not supported in XRC version 2.

You can use the RFREQUENCY and/or the RTRACKS parameters in the XSET command to tell the ESS (for hardware bit maps) and the SDM (for software bit maps) when those bitmaps should be reset. ESS and SDM will reset bit maps according to those parameters for all active volumes, bit maps for suspended volumes will not be reset.

In the RFREQUENCY parameter you can specify how often, in hours, minutes, and seconds, the bit maps should be reset. With the RTRACKS parameter, you specify the number of tracks that must change before the bitmaps are reset.

The default for RFREQUENCY is 30 minutes. If you specify a value of 0, XRC will not reset the bit maps as a result of elapsed time.

The default for RTRACKS is 7500 tracks. If you specify a value of 0 (or a value greater than the number of tracks on a volume), XRC will not reset the bit maps as a result of the number of tracks that have changed.

XRC uses whatever happens first of the REFREQUENCY or RTRACKS specifications.

Specifying high values in the RTRACKS and RFREQUENCY (for RFREQUENCY it depends on the update rate on primary volumes) means that the resynchronization time can take longer. For example, if you have specified RTRACKS 7500 (and let us assume that RFREQUENCY is 0) and a volume is suspended when 4000 tracks have been changed since the bit map was reset last time, 4000 tracks have to be resynchronized.

Specifying a low value in the RTRACKS and RFREQUENCY can put a greater demand on ESS and SDM processing resources.

We recommend that you keep the defaults unless you have specific needs.

### 5.22.11 DONOTBLOCK parameter in the XADDPAIR command

In 5.12, “Dynamic workload balancing” on page 224 we explained how XRC will try to balance the needs of the primary system and XRC when data starts to accumulate in the cache. Data paths to primary volumes with high write activity will progressively be blocked. That is, writes from the primary host will be inhibited until SDM can catch up. If there are volumes in your XRC configuration, which you do not want to have their data paths blocked, you can specify those volumes in the DONOTBLOCK parameter.

You should specify this option only for high performance, sensitive volumes that generate very short updates, like the IMS WADS volumes and spool data sets.

### 5.22.12 PRIORITY parameter in the XSET command

The PRIORITY parameter in the XSET command specifies the priority that the XADDPAIR command uses for selecting the next volume to synchronize or resynchronize.

The first priority is to resynchronize existing volume pairs, then to add new volume pairs to the XRC session.

If you specify PRIORITY(FIFO), XRC processes the volumes that were listed by an XADDPAIR command in the order that they are listed. First, XRC processes all volumes that are eligible for resynchronization. Next, XRC processes all volumes that are eligible for synchronization. XRC processes the volumes within the limits that you specified in the SYNCH and SCSYNCH parameters of the XSET command. (See 5.21.13, “ANTRQST application programming interface” on page 258.)

If you specify PRIORITY(LOAD), XRC processes the volumes that were listed by an XADDPAIR command based on the primary and secondary storage control load. First, XRC processes all volumes that are eligible for resynchronization. XRC can defer the initialization if the associated storage control has too much activity at the time that the volume is eligible.

The volume continues to be eligible for processing when the activity level has decreased. That is, it processes in the order you specify, but will skip volumes if they are busy, and come back and try again later. Next, XRC processes all volumes that are eligible for synchronization. Again, XRC can defer the initialization if the primary storage control activity is too high. XRC performs both resynchronization and synchronization processing within the limits that the SYNCH and SCSYNCH parameters of the XSET command specify.

Changes specified with the PRIORITY parameter take place when the next volume is processed. The default value is PRIORITY(LOAD).

### 5.22.13 SCSYNCH and SYNCH parameters in XSET command

The SDM monitors the update activity and the XADDPAIR activity by subsystem, and attempts to throttle the XADDPAIR activity such that the update activity is minimally impacted.

The SCSYNCH/SYNCH parameters can be set up to allow multiple concurrent **XADDPAIR**s across multiple primary LCUs.

**SCSYNCH:** This parameter specifies the maximum number of volume pairs that the SDM can synchronize or resynchronize concurrently per LCU. You can specify the SCSYNCH keyword as SCSYNCH(p,s) or as SCSYNCH(n), where:

- **p** is the maximum number per primary LCU
- **s** is the maximum number per secondary storage control
- **n** applies to both primary and secondary storage control

The parameter value applies to an LCU, regardless of the number of SC sessions that are active on the storage subsystem.

Set the SCSYNCH value in conjunction with the SYNCH parameter, which specifies the session limits for concurrent volume synchronization tasks.

Changes specified with the SCSYNCH parameter take place when the next volume initialization is processed. The default is SCSYNCH(2,2)

**SYNCH:** This parameter specifies the maximum number of volume synchronization or resynchronization tasks that XRC can concurrently start in the XRC session. The range is from zero to 45 tasks; the default value is four. Specify a value of zero to ensure that XRC does not start any new volume initialization.

Changes specified with the SYNCH parameter take place when XRC process the next volume.

## 5.23 Managing an XRC environment

In this section we provide some recommendations, tools, and procedures to help you manage the XRC environment.

## 5.23.1 Managing a coupled XRC environment

Coupled Extended Remote Copy (CXRC) support, also referred to as coupled system data movers (SDMs), allows multiple XRC sessions to be combined into a single logical session.

In defining a CXRC configuration, each SDM is set up as if it were a stand-alone independent SDM. The actual coupling can then be achieved by issuing the XCOUPLE command to each of the XRC sessions. In addition, a CXRC configuration requires a master data set, which is accessible by the SDMs.

### Identifying sessions in a coupled environment

Table 5-3 describes session types that are used in a coupled environment.

*Table 5-3 Coupled session descriptions*

| Session | Description |
|---|---|
| Active session | The system data mover is running an XRC session. Volumes can be added or deleted. You can issue the **XADVANCE** command and the **XCOUPLE** command with **ADD** and **DELETE** parameters to individual active XRC sessions. |
| Coupled session | An active individual XRC session has been added to a master session. |
| Inactive session | The system data mover is not active. You can issue the **XADVANCE**, **XSTART**, **XRECOVER**, and the **XCOUPLE** commands with the **PURGE** parameter to individual inactive XRC sessions. |
| Interlocked session | This is a coupled XRC session in a master session that has a consistency time before the master journal time and a journal time equal to or after the master recoverable time.<br>The master recoverable time is the minimum of the journal times reported by all XRC sessions. This is the latest time to which all sessions could recover their data in the event of a disaster, and still provide coupled consistency.<br>XRC will recover all interlocked sessions such that all secondary volumes across the coupled interlocked sessions will be consistent. Because coupled consistency requires all volumes to be at the same time, no session is allowed to apply data which is after the master recoverable time. |
| Master session | This is a logical entity that is used to coordinate session commands and data consistency across multiple XRC sessions. A master session exists as long as there is an XRC session coupled to the master session.<br>You can issue the **XQUERY**, **XEND**, **XSUSPEND**, and **XCOUPLE RELEASE** commands to a master session. |
| Suspended session | You can issue the **XADVANCE** command, **XSTART** command, **XRECOVER** command, and the **XCOUPLE** command with the **PURGE** parameter to individual suspended XRC sessions. |
| XRC session | You can issue the **XCOUPLE ADD** command to an XRC session to couple it to the master session. To uncouple the XRC session from the master session, issue the **XCOUPLE DELETE** command. |

### SDM boundaries

Ideally, it is best to split volumes so that the SDMs receive an equal workload distribution, possibly by the disk subsystem or by the subsystem identifiers (SSIDs).

### Using journal, state, and control data sets

There are special considerations for journal, state, and control data sets when you work with master data sets in CXRC. In a CXRC environment, each SDM uses its own journal, state, and control data sets. Place these data sets so a performance bottleneck is not created due to contention between the multiple SDMs for the same channel path, disk subsystem, or volume resources.

### Coupling XRC sessions

Use the XCOUPLE ADD command to couple an XRC session to a master session. The first time you issue this command for a session, XRC creates a new member in the session's state data set to indicate that the session is coupled to the specified master session. The member is also used during XSTART processing to determine if the session is to be recoupled to a master session. The XCOUPLE ADD command may only be invoked against XRC sessions that have been started with the SESSIONTYPE(XRC) parameter.

### Querying coupled status for XRC sessions

To determine the status of each coupled session that is associated with the specified master session, issue the XQUERY command with the MASTER parameter. The XQUERY output indicates the status of the master session, the status of each session, and the overall volume status for each session. In addition, the XQUERY command that is specified with the VOLUME or SUMMARY parameter provides the master session status.

### Abnormal XRC coupled session status

The IBM publication *z/OS DFSMS Advanced Copy Services,* SC35-0428, contains scenarios for managing the status of a CXRC environment in problem situations. Refer to this manual for a more detailed discussion.

## 5.23.2 Automation

An XRC environment may involve hundreds if not thousands of volumes. We recommend that you keep the configuration as simple and symmetric as possible, for example, a one-to-one correspondence between primary and secondary LCUs.

Nevertheless, you should implement some form of automation. Automation for XRC may range from simple JCL-/REXX generation code, to fully automated monitoring and recovery actions.

### JCL-/REXX generation

JCL-/REXX generation code is recommended to minimize errors when performing XRC operations. Code can be automatically generated to XADDPAIRs, to XSUSPEND, and so on. This will ensure that all eligible volumes are included in all lists. Even if major operations such as migration and recovery are going to be under manual control, errors can be minimized by automatic generation of jobs to be executed.

### Error recovery

Many systems management software packages allow error messages to be intercepted, filtered, and directed to an automation package. For example, if a volume drops out of duplex status, an operator could be notified, or an automatic XADDPAIR may be initiated.

Automation code may also be built to automate the full XRC recovery process in the event of a disaster. For Disaster Recovery it is important not to rely on personnel who may not be available to perform the recovery actions.

### 5.23.3 XRC and GDPS

GDPS, an industry leading e-business availability solution available through IBM Global Services, is a multi-site solution that is designed to provide the capability to manage the remote copy configuration and storage subsystems, automate Parallel Sysplex operational tasks, and perform failure recovery from a single point of control, thereby helping to improve application availability. GDPS supports both the Synchronous Peer-to-Peer Remote Copy (PPRC), as well as the asynchronous Extended Remote Copy (XRC) forms of remote copy.

Geographically Dispersed Parallel Sysplex (GDPS) is an integrated solution offering that manages all aspects of switching computer operation from one site to another, planned or unplanned. The SDM is placed outside the production sysplex. This support of XRC as a remote copy technology should accommodate two types of installations:

- Installations that cannot tolerate the performance impact of the synchronous remote copy technology
- Installations that need more distance between their primary and secondary sites than what can be handled with ESCON/FICON and dark fiber

One subset of the GDPS solution is the Remote Copy Management Facility (RCMF) offering. RCMF is an automated disk subsystem and remote copy management facility, with a high level user interface. This interface is implemented in the form of ISPF like displays, and virtually eliminates the tedious and time consuming work with TSO commands.

Managing XRC with RCMF could be the first step of a full GDPS implementation. GDPS is discussed in Appendix B, “Geographically Dispersed Parallel Sysplex (GDPS)” on page 359.

### 5.23.4 ICKDSF and XRC volumes

Recent developments in ESS, XRC, and ICKDSF have eliminated the need to XDELPAIR XRC volumes prior to relabeling DASD volumes. These developments allow for concurrent management and operations, as well as increasing XRC’s readiness as a Disaster Recovery solution.

**Note:** You must be running on recent ICKDSF and XRC maintenance.

When ICKDSF is used to initialize a volume, the updates will be captured by XRC and not written to the secondary DASD. XRC will update its control blocks so that a subsequent recovery will reflect the new volser of the primary volume. The V<volsername> member of the State Dataset is not updated until the next time the volume is suspended and XADDPAIR SUSPENDED.

The volser in the MVS UCB control block is not updated until the next VARY OFFLINE/ONLINE is performed. It is also recommended that an XRC CONFIG be performed to resync GDPS. We no longer have to do an XDELPAIR/XADDPAIR full volume copy, but we do still have some operations to *sync up* XRC, MVS, and GDPS.

Here is a procedure to update the VOLSER while XRC is running:

1. Vary the primary volume offline during ICKDSF operation.
2. ICKDSF to relabel the VOLSER (XRC control blocks are now correct).
3. Vary primary volume online after ICKDSF operation (UCB is now correct).
4. XRC SUSPEND VOLUME followed by an immediate XRC XADDPAIR SUSPENDED to update the state dataset's member name (can be done anytime).
5. *GDPS only*: GDPS CONFIG to update GDPS's policy (again, this can be done anytime).

## 5.23.5 SMF

For an active XRC session, SDM writes an SMF record type 42 subtype 11 whenever the SMF time interval expires. This record contains data about the LCUs participating in the XRC session and XRC statistics that can be used for further reporting to monitor XRC activity. See the *z/OS V1R3.0 MVS System Management Facilities (SMF),* SA22-7630 manual for a complete layout of SMF type 42 records.

## 5.23.6 Controlling access to XRC commands

There are two ways to protect access to XRC commands: You can put the commands in a RACF-protected library, or define resource profiles in the RACF FACILITY class and restrict access to those profiles.

### Placing TSO commands in a RACF protected library

Place the TSO commands in a RACF-protected library to restrict XRC TSO commands to authorized storage administrators.

To RACF-protect XRC commands, perform the following steps:

1. Issue the following RDEFINE command for each XRC command, and for each command abbreviation that you want defined to RACF:

   **RDEFINE PROGRAM** *cmd* **ADDMEM('SYS1.CMDLIB'/***vol***/NOPADCHK) -**

   **UACC(NONE)**

   The following terms apply to the above example:

   **cmd**: Defines the XRC TSO command name or an abbreviation of a command. Issue a separate RDEFINE command for each of the XRC commands, and any command abbreviations you plan to use. Examples of XRC command abbreviations are XADD, XDEL, and XRCV. RACF can only perform checking on commands and abbreviations that are defined to it.

   **vol**: Defines the name of the volume that contains the SYS1.CMDLIB data set.

2. Issue the PERMIT command for all commands and authorized XRC TSO command users as follows:

   **PERMIT** *cmd* **CLASS(PROGRAM) ID(***name***) ACCESS(READ)**

   The following terms apply to the above example:

   **cmd**: Defines the XRC TSO command name, or an abbreviation of a command.

   **name**: Defines the user ID receiving RACF access authority for that command name.

3. Issue the SETROPTS command from a user ID that has the appropriate authority:

   **SETROPTS CLASSACT(PROGRAM) WHEN(PROGRAM) REFRESH**

### Defining resource profiles in the RACF FACILITY class

You can limit the use of XRC commands by defining resource profiles in the RACF FACILITY class and restricting access to those profiles. To use a protected command, you need read access authority to the applicable profile.

All XRC commands can be restricted by using the FACILITY class profile **STGADMIN.ANT.XRC.COMMANDS**.

Use of the XQUERY command can also be authorized in the FACILITY class profile **STGADMIN.ANT.XRC.XQUERY**. XRC first checks **STGADMIN.ANT.XRC.COMMANDS** for authorization, and if authorization is not permitted with this profile, XRC will check **STGADMIN.ANT.XRC.XQUERY**.

The following examples activate the RACF FACILITY class, define the profile for the XRC commands, and give user STGADMIN authority to use this profile:

1. Activate the RACF FACILITY class:

   ```
   SETROPTS CLASSACT(FACILITY)
   ```

2. Define the profile for XRC commands, and authorize user STGADMIN to use this profile:

   ```
   RDEFINE FACILITY STGADMIN.ANT.XRC.COMMANDS UACC(NONE)
   PERMIT STGADMIN.ANT.XRC.COMMANDS CLASS(FACILITY) -
   ID(STGADMIN) ACCESS(READ)
   ```

## 5.23.7 Restricting access to secondary volumes

XRC requires that the primary and secondary volumes have different volume serial numbers because they both must be on-line to the SDM system.

Non-SDM applications can access secondary volumes by using a different catalog or by explicit unit and volser reference. However, we absolutely recommend that you prevent access to secondary volumes from all non-SDM applications because:

- Write operations to the secondary other than by the SDM produce data integrity problems.
- Read activity to secondary volumes may update last reference date for the data, and may experience various error conditions due to the way XRC handles secondary I/O functions.

One way to prevent access to the secondary volumes is to define the volumes to a storage group with the disable state (DISALL). SDM will not be affected by this status.

## 5.23.8 XRC volume format, track, and access mode restrictions

To ensure data integrity, all volumes that are part of XRC copy operations must conform to the following conditions:

- Volumes must have a standard format for record zero (R0). Volumes with R0 data lengths longer than 8 bytes can cause a track format error to remain undetected when the track is formatted in cache.
- Alternate tracks cannot be assigned in the user area. If the secondary address has a track assigned as an alternate track and the track is a user track on the primary address, user data may be overlaid.
- All storage control Define Extent commands must specify normal access authorization mode. Data written to a remote copy primary device while the storage control is in *diagnostic* or *device support* mode is not copied to the XRC secondary volume. It is therefore important to remove volumes from the session before running a utility program like ICKDSF.

## 5.23.9 HCD reconfiguration restriction

Do not reconfigure a device through the Hardware Configuration Definition (HCD) while that device is a member (primary or secondary) of an XRC pair.

## 5.23.10 XRC diagnostic aids

In this section we discuss the diagnostic tools that can be used in XRC problem situations.

### Restarting XRC address space

The XRC control address space, ANTAS000, is automatically started during system IPL. ANTAS000 handles the TSO commands that control XRC.

There may be conditions that prevent the system from starting this address space at IPL time, or to restart it automatically if it is cancelled by the operator. For such cases, you can restart the ANTAS000 address space by submitting the job shown in Figure 5-29.

```
//STARTANT JOB MSGLEVEL=(1,1),REGION=4096K
//STEP1 EXEC PGM=ANTSTRT
```

*Figure 5-29 Restart ANTAS000 address space*

### Diagnosing XRC with MVS MODIFY command

There are options to the MVS MODIFY command that can be used to diagnose XRC problems, and in some cases, to resolve them. For some of the options, there are no requirements that the MODIFY command be issued from the SDM system; it can be issued from any system with XRC support attached to the primary LCU.

There are 13 different operations that can be used in an XRC environment. The format of the MODIFY command is illustrated in Figure 5-30.

```
F ANTAS00x,operation optional_operands
```

*Figure 5-30 MODIFY command*

If **x=1**, the command must be issued from the SDM address space. If **x=0**, the command can be issued from any system attached to the primary ESS that supports XRC (the ANTAS000 address space must be active).

Next we show some of the available operations. All operations mentioned here can be issued from any system (with ANTAS000 address space active) attached to the primary LCU. You have to specify a device address in the primary LCU that you want to address with your command.

- **LISTSESS:** This operation returns the following information from the LCU:
  - Type of SC-sessions in the LCU
  - Status of the SC-session
  - SC-session number
  - SC-session owner
- **TERMSESS:** This operation is very powerful and can be used to end active, quiesced, suspended, or timed-out XRC or CC sessions for an LCU. TERMSESS provides four kinds of support:
  - To end a single SC session (CC or XRC) by specifying SC session number (available through the LISTSESS command).
  - To end all active or time-out XRC SC sessions that were last active on the system from which this TERMSESS operation is issued.

- To end all quiesced or suspended XRC SC sessions, regardless of which system the XRC operation was last active.
- To end all XRC or CC SC sessions that are in time-out status.

- **LISTDVCS:** This operation lists all devices that are part of a specific SC session. You have to specify the SC session number, which can be obtained by the LISTSESS operation.

For further description and additional options provided with the MODIFY command to diagnose and repair XRC, see the IBM publication *z/OS DFSMS Advanced Copy Services,* SC35-0428.

# 5.24 XRC testing

In this section we provide a summary of the recommended testing to go through before XRC is put into production.

## 5.24.1 Functional testing

This is to test that all XRC commands execute as expected and with correct return codes.

When performing these tests you should use XQUERY to produce various reports, both to get familiar with the content, but also to verify that the status of volumes, volume pairs and session is as expected.

## 5.24.2 Operational testing

This is to ensure that all required procedures are in place for operating the XRC environment. You should test scenarios that may be used in a production environment, like suspending sessions, adding utility devices, changing error recovery levels, suspending and resynchronizing volume pairs, and so on.

Some suggested operational scenarios are documented in the publication *z/OS DFSMS Advanced Copy Services,* SC35-0428.

## 5.24.3 Error injection testing

You may want to simulate various failure situations to verify that your operating procedures work.

When simulating a disaster, the most likely scenario is that one or more ESSs, or the links to these, are lost. These potential failure situations are best simulated, for example, through the ESCON director. Individual links may be removed, breaking the connection between the SDM and the primary ESSs. The recovery scenario can then be tested on the secondary site. A *rolling disaster* may be simulated by breaking the connections between the SDM and the primary ESSs at different times.

## 5.24.4 Automation testing

If you follow our automation recommendations (given in 5.23.2, “Automation” on page 264), you need to test the automation code to ensure that it operates correctly, generates complete and useful reports, intercepts error messages, and initiates the required actions.

### 5.24.5 Walk-through

If you are planning to use XRC for Disaster Recovery or workload migration, you may not have the facilities to test in a *production equivalent* environment. In these cases it can be a useful option to do a walk-through the critical procedures. This can identify problems such as ownership of responsibilities, and access to critical data. If you rely on manual intervention for go or "no-go" on Disaster Recovery, then the process must be clearly documented at both primary and recovery sites.

### 5.24.6 Dress rehearsal

The proof of a disaster recovery test is a full-scale dress rehearsal, preferably one that can be initiated without warning. Because XRC supports SUSPEND/RESUME processing, you can test recovery at a secondary site without disruption to your production environment. Once the test is complete, changes that have been accumulated at the primary site can be propagated to the secondary site, restoring disaster protection.

If it is essential to maintain disaster protection during a disaster recovery test, FlashCopy provides an ideal tool to take a quick backup of data at the secondary site. The disaster test can proceed against the FlashCopy of the secondary devices, while XRC between the primary and secondary devices is resumed.

Once a disaster recovery plan has been successfully validated by a full-scale test, it is important to make sure that documentation is maintained, and that the test is repeated at regular intervals.

## 5.25 Database recovery with XRC

The recovery of data is probably the most complex and vital component of a disaster recovery plan. The most crucial aspect of recovering databases is to ensure consistency of the recovered data. With XRC, data consistency can be guaranteed to an identified point-in-time across all volumes in the XRC session. This is done by the creation of a Consistency Group as we have described earlier in this chapter, and this is independent of applications and databases.

### 5.25.1 Today's backup recovery process

To achieve database consistency today, the backup and recovery steps may be similar to these:

1. Back up the database according to business needs. Different backup philosophies can be implemented, such as periodic backup from a time when all files are:
   - Closed with no forward recovery
   - Closed with forward recovery
   - Open with forward recovery
2. Ship to the secondary site:

   Periodically ship image copies to the recovery site.
3. Transfer log data sets:

   Use communication lines or manual transport to transfer the log after the log data set has been closed.

The recovery process includes the following activities:

1. Restore the latest image copies:

   If the latest image copies are not available, the recovery process will take longer because of additional forward recovery tasks.

2. Forward recovery with logs:

   Restore the databases to a known point-in-time. Apply the log data to make the database as current as possible.

3. Take image copies:

   Take image copies of the recovered database to have a consistent restart point.

4. Reprocess orphan data:

   Orphan data can consist of non-transferable log data and/or transferable log data sets at the primary site, and log data sets that have not yet arrived at the secondary site.

5. Check consistency of databases:

   Before you restart the applications, we highly recommend that you check the consistency of the databases.

### 5.25.2 Backup recovery process with XRC

Your normal operational requirements will already include procedures for restarting on-line applications in case of a short term outage (a power failure or CEC failure for instance).

For XRC Disaster Recovery, the same procedure should be used at the secondary site. The same checks for guaranteeing data consistency and restarting your applications must be done at your secondary site, just as they are done today at your application site.

XRC provides data currency across all volumes in the XRC session. In the event of a disaster at the primary site, updates in flight are lost. Nevertheless, recovery can be achieved to an identified consistent point-in-time. The recovery technique does not require database restore followed by forward recovery. Recovery can follow the procedure used at the primary site for a system outage, consistent to the point-in-time given. Any transaction that have been completed after this point-in-time might require investigation and be recreated against the secondary disks after recovery (using XRECOVER) or accounted for in your business process.

## 5.26 Using XRC with FlashCopy and Tertiary Volumes

The use of ESSs at the recovery site provides a unique ability to combine FlashCopy functions with XRC functions.

For example, you can use this combination to create data for disaster recovery testing or consistent data for point-in-time backup purposes. This technique also allows you to keep the existing data mover session active while you perform the recovery test.

**Note:** During a real Disaster Recovery, XRC will verify that the secondary VOLSERs and device numbers match those used in the last active XRC session, and will fail the recovery if they do not. This done to ensure that XRC is recovering the correct information. When FLASHCOPY is used to create another set of secondary volumes for DR testing, all data, including the VOLSER, is copied to a new device number. At this point the VOLSER and device number do not match. The ONLINE parameter indicates that recovery is using tertiary DASD and XRC is to verify only the VOLSER, allowing any device online with a VOLSER that matches a valid secondary VOLSER to be used for recovery.

The following scenario describes one method of creating data for disaster recovery testing by using FlashCopy and XRC commands. The following text is taken from the APAR OA06955, PTF UA10897, which applies to z/OS V1.1 and DFSMS V1.3 onward.

1. Suspend all volume pairs in the XRC session using the XSUSPEND VOLUME(ALL) command.
2. Issue FlashCopy commands to copy secondary volumes to tertiary volumes. Do not include volumes that contain the journal, state, control, and master data sets.
3. Using the DFSMDdss COPY DATASET command with the RENAMEUNCONDITIONAL option specifying a different high level qualifier, and the TOLERATE(ENQFAILURE) option, copy journal, state, control, and master data sets to tertiary volumes. If the session has been suspended, the TOLERATE(ENQFAILURE) option is not required. The high level qualifier used must not be the same as that used for the active session HLQ and MHLQ, and the high level qualifier used must be the same for all these data sets, including the master data set.
4. Issue the XADDPAIR command with the SUSPENDED parameter specified to add suspended volumes back into the session. Wait until all suspended pairs are in duplex state.
5. Vary the secondary volumes (FlashCopy source volumes) offline to the system on which the XRECOVER or XADVANCE commands will be issued.
6. Vary the tertiary volumes online to the system on which the XRECOVER or XADVANCE commands will be issued.
7. Issue the XRECOVER or XADVANCE command with the ONLINE parameter and the HLQ parameter with the new high level qualifier specified in step 3. Specifying the ONLINE parameter with the HLQ parameter will indicate to XRC that the HLQ will be used for all of the XRC data sets, including the master data set. Remember that the XRECOVER command will clip (match) the volsers to secondary volsers, but the XADVANCE command will not.

**Note:** The XRECOVER command must not be issued on the same system as an active session with the same XRC logical session name. Otherwise, the command will fail with a return code RC416. The command may be issued on a different system running in the same SYSPLEX.

6

# Concurrent Copy

This chapter describes the characteristics and operation of Concurrent Copy. We also discuss the considerations involved when planning to use Concurrent Copy.

The information presented in this chapter can be complemented with the following publications:

- *z/OS DFSMS Advanced Copy Services,* SC35-0428
- *z/OS DFSMSdss Storage Administration Reference,* SC35-0424
- *z/OS V1R3.0 DFSMS Installation Exits*, SC26-7396
- *DFSMS: Implementing System Managed Storage*, SC26-7407
- *DFSMShsm Storage Administration Guide*, SC35-0388
- *DFSMS/MVS Managing Catalogs*, SC26-4914

# 6.1 Introduction

Concurrent Copy is a copy function that helps you keep your high data availability objectives by allowing point-in-time copies of your data *concurrent* with normal application processing. Concurrent Copy works with the IBM TotalStorage Enterprise Storage Server and the DFSMS System Data Mover (SDM). Concurrent Copy is available for the z/OS and OS/390 operating systems, and requires software support provided in DFSMS/MVS®.

Concurrent Copy allows you to generate a *copy* or a *dump* of the data while the applications are updating that data. Concurrent Copy works not only on a full-volume basis, but also at a data set level. Also, the target is not restricted only to DASD volumes in the same ESS, but the target can also be a tape cartridge or a DASD volume on another ESS (see Figure 6-1).

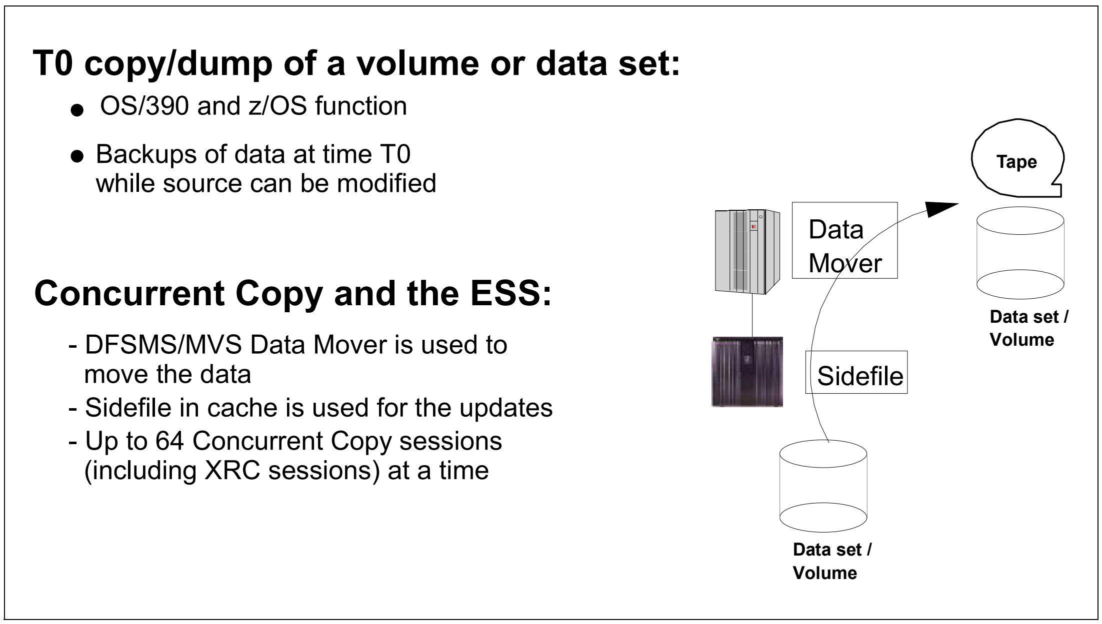



*Figure 6-1 Concurrent Copy basic characteristics*

# 6.2 Concurrent Copy terminology

This section lists the definitions of the terms used with Concurrent Copy. These terms will be discussed in detail in the following sections of this chapter.

### *Session*

A session is a logical concept representing a single invocation of Concurrent Copy (a single DFSMSdss DUMP/COPY command). A session can include one or more data sets or volumes, on the same ESS or across different ESSs. An individual ESS can support up to 16 simultaneous Concurrent Copy sessions per volume, with a maximum of 64 simultaneous sessions per LSS.

### *Session ID*

The system assigns a unique session ID to each Concurrent Copy session. The system uses the session ID to identify and coordinate all host and ESS resources associated with a particular Concurrent Copy session.

### *Concurrent Copy domain*

The set of devices and tracks identified during the initialization of a Concurrent Copy session is called the Concurrent Copy domain. It represents the set of data that Concurrent Copy copies.

### *Intercepted writes*

When an application tries to update information that is included in a Concurrent Copy domain, the ESS intercepts those writes, thus maintaining a copy of the data as it was at the time when the Concurrent Copy was requested.

### *Sidefile*

A sidefile is a temporary repository for Concurrent Copy domain tracks that still have not been copied by the SDM and are about to receive an update. During the processing of an intercepted write, the ESS copies a before-image of the track being updated into a sidefile for later processing. Together, the ESS and the SDM maintain two sidefiles, one in the ESS cache and another in processor storage.

### *Terminate*

The Concurrent Copy terminates when DFSMSdss has copied the Concurrent Copy domain and both the sidefiles are empty. In error situations, either the ESS or the SDM can terminate the Concurrent Copy session before the entire domain has been copied.

### *Fuzzy copy*

Without a tool like Concurrent Copy, if you make a copy of the data while the data is being updated then the copy will not reflect a point-in-time version of the original data. In this case, the copy will be *fuzzy*, because it does not represent any particular point-in-time status. A fuzzy copy is a set of data with no logical consistency from the application perspective.

### *Consistent copy*

A consistent copy is a set of data with logical consistency from the application perspective. The logical consistency of the source data is guaranteed by the application itself. The logical consistency of the copied data must be ensured by a means like Concurrent Copy. With the application taking care at all times that the original data is consistent, then with Concurrent Copy been able to produce a point-in-time copy of the complete set of data, then the copy will hold the same logical consistency status as the original data at that certain point-in-time.

## 6.3 Benefits of using Concurrent Copy

Concurrent Copy can dramatically reduce the amount of time that is required to back up your application data, hence increasing the application's availability time. When you use Concurrent Copy, application processing is interrupted for only a minimum time while the system initializes the Concurrent Copy *session*. Once Concurrent Copy is active, your applications continue to process the data while it is being backed up using Concurrent Copy.

Concurrent Copy provides point-in-time data consistency. The system serializes access to the data being dumped or copied just long enough for the Concurrent Copy session to be initialized. This serialization takes very short time to complete, and this process ensures the point-in-time copy will be consistent while the copy is being done with your applications running.

## 6.4 Concurrent Copy operation

For the Concurrent Copy process, we must distinguish between the *logical completion* of the copy and the *physical completion*. The copy process is logically complete when the System Data Mover has figured out what to copy. This is a very short process. After the logical completion, updates to the source are allowed while the System Data Mover, in cooperation with the ESS, ensures that the copy reflects at all times the state of the data at the time that Concurrent Copy was invoked.

The System Data Mover (SDM), a DFSMS/MVS component, reads the data from the source (volume or data set) and copies it to the target. When an update to the source is to be performed and this data has not yet been copied to the target, the original data is first copied to a sidefile in the ESS cache before the source is updated (see Figure 6-1). The Concurrent Copy process includes the following steps:

1. Initialization
2. Copying the data
3. Intercepting the writes
4. Session termination

### 6.4.1 Initialization

Initialization takes only a very short time to complete successfully. The session initialization process is a series of distinct steps, as follows:

1. An appropriate time is determined to start a backup or copy and for a brief time interrupt the update activity to the application.
2. DFSMSdss uses Concurrent Copy to perform the copy or dump of the data. You can use DFSMSdss directly from JCL, or from an application through the DFSMSdss application interface or User Interaction Module (UIM).
3. DFSMSdss locates the data sets to process and breaks them down into a hierarchy of volumes and extents.
4. DFSMSdss interfaces with the SDM to initialize a Concurrent Copy session.
5. The SDM determines which LSSs are attached to the affected volumes, and defines a Concurrent Copy session to each LSS. The SDM also makes each LSS aware of the volume and the extents on each volume that are included in the Concurrent Copy session. Once the SDM has completely initialized the session, it returns control to DFSMSdss.
6. DFSMSdss checks all SDM return codes. DFSMSdss releases serialization on those data sets or volumes, which were successfully included in the Concurrent Copy session. DFSMSdss keeps serialization on data sets or volumes that are not included in the session.
7. DFSMSdss issues messages that indicate the status of the session. If the application interface called DFSMSdss, DFSMSdss uses the *User Interaction Module* (UIM) to update the calling program with the status of the session.
8. The application can take appropriate action, which may include restarting applications. Automation traps may be set up to take action based on the issued messages.

### 6.4.2 Copying the data

Once the initialization is complete, DFSMSdss then starts to read the data from the source to perform the dump or copy operation. Applications can resume reading and writing the data while Concurrent Copy is making the copy. Meanwhile, the ESS monitors the I/Os for updates to any tracks that are included in the Concurrent Copy session.

DFSMSdss uses the SDM to read data from the devices within the Concurrent Copy session. For data that was not included in the Concurrent Copy session, DFSMSdss reads the data itself in the same way as for a non-Concurrent Copy operation. As DFSMSdss receives tracks from the SDM, or reads them itself, it writes the data to the output device *in the same way* as for any DFSMSdss operation.

### 6.4.3 Intercepting the writes

The purpose of Concurrent Copy is to take a copy of data without including any updates that are made to the data while the copy proceeds. To achieve this, the ESSs involved in a Concurrent Copy session monitor I/Os to the tracks included in the session. If an I/O is about to update a track that SDM has not yet copied, the ESS saves a copy of the original track image in the cache *sidefile* before it allows the I/O operation to continue. If DFSMSdss has already read the track or the ESS has saved it in the cache sidefile, the ESS does not save another copy. If a particular track is covered by more than one Concurrent Copy session, the ESS saves a copy of the track in the cache sidefile for each session. Within the next few channel programs, the SDM retrieves the track image from the *cache sidefile* and buffers it in a *host sidefile* stored in a data space until DFSMSdss requests the track.

In response to requests from DFSMSdss, the SDM merges tracks from the cache and host sidefiles with tracks read directly from DASD. The SDM delivers the resulting tracks to DFSMSdss in the same order in which they would have been read directly from DASD. This characteristic of the SDM and DFSMSdss ensures that the output from a Concurrent Copy operation is in the same format as a non-Concurrent Copy operation.

Each intercepted write on the Concurrent Copy process causes the ESS to stage the track concerned into the cache sidefile. When the SDM detects any tracks in the cache sidefile, it moves the tracks into the host sidefile, releasing the cache storage occupied by the cache sidefile. The application update operation is delayed only until the ESS stages the track into the cache sidefile.

### 6.4.4 Session termination

Once the tracks in the Concurrent Copy domain have been transferred to the output device, the session can be terminated. DFSMSdss interfaces with the SDM to terminate the Concurrent Copy session. The SDM cleans up the Concurrent Copy operation and releases all host sidefiles buffers associated with the Concurrent Copy session.

### 6.4.5 NOTIFYCONCURRENT keyword

For a *logical* data set `COPY` operation, the `NOTIFYCONCURRENT` keyword specifies that DFSMSdss issues a message for every data set that is successfully included in the Concurrent Copy operation. If not specified, messages are issued only for data sets that are not successfully included in the Concurrent Copy operation.

## 6.5 Serialization with Concurrent Copy

This section discusses the considerations related to the integrity of the data on data sets and volumes receiving updates while Concurrent Copy is copying or dumping the data.

## 6.5.1 Data set integrity

Suppose you took a Concurrent Copy of a data set that was in the process of being updated by a batch job, and you said to tolerate enqueue failure. The copy accurately represents the contents of the data set at the moment of logical completion (when the ESSs agreed to the session). Now you restore that copy and it really does have the same content as when the dump was logically completed. What is the usability of the data set? The status of any record in the data set is unknown. It may or may not have been changed by the batch job. The data set is really unusable from any reasonable point-of-view.

From this it should be apparent that the status of the data set at the time of the Concurrent Copy must be known if the copy is to be usable. If the data set is closed when the copy is taken, then the status is known (or can be inferred). If the data set is open for update then we need some kind of synchronization point (or *checkpoint*) identification that can be taken with the data set and put back when the data set is restored. From this synchronization point it then becomes the responsibility of the application to perform a forward recovery, or ensure the usability and status of the now recovered data set.

With this in mind, it should be clear that a Concurrent Copy for a data set that is open for update should only be made when the application is *aware* that a copy is to be made, and is therefore in a position to do *forward recovery* of the copy that might subsequently be restored. Typically, this function is available only in database managers.

If the updating application has no knowledge of the copy process nor a recovery capability, then a Concurrent Copy is no more usable than a standard copy. In consequence of these considerations, the serialization requirements for DFSMSdss do *not* change with DFSMSdss Concurrent Copy.

The VTOC is serialized as usual, for the length of the physical dump or until released after the VTOC is read, if so directed by the user's exit. The Concurrent Copy dump has no more content integrity than a standard-copy dump.

The initiation of a Concurrent Copy dump or copy of a data set does nothing to change data set serialization. If you did not say to tolerate enqueue failure, the data set will *not* be dumped/copied if it is open for output. If it is not open, the data set will be serialized for the dump/copy but released as soon as the function is logically complete.

If you do say to tolerate enqueue failure, DFSMSdss will still try first to serialize the data set, and only when that has failed do we dump or copy the data set anyway. In this case, there is no synchronization performed by DFSMSdss with respect to the updating program. The usability of the copy is dependent on the capabilities of the application.

## 6.5.2 Concurrent and BWO dumps

Concurrent Copy improves *backup-while-open (BWO)* processing by significantly reducing the invalidation of a backup-while-open dump because of updates to the data set. Here is a comparison of the various kinds of dumps you can ask for:

- **Normal dump:** Use of the data set must be quiesced so that serialization is obtained, the data set is dumped, and serialization is released. The data set cannot be used for the entire time.
- **Concurrent Copy dump:** Use of the data set must be quiesced so that serialization is obtained, Concurrent Copy initialization is completed within a very short time (compared to the actual time to dump the data set), serialization is released, and the data set is dumped. The data set can be used after Concurrent Copy initialization is complete.

- **Backup-while-open dump:** Serialization is attempted but is not required, and the data set is dumped. If eligible for backup-while-open, the data set is dumped without serialization and can remain in-use for the entire time, but the dump can be invalidated by update activity to the data set.
- **Backup-while-open dump using Concurrent Copy:** Serialization is attempted but is not required, Concurrent Copy initialization is completed, and the data set is dumped. If eligible for backup-while-open, the data set is dumped without serialization and can remain in-use for the entire time, and updates that occur do not cause the dump to be invalidated.

### 6.5.3 Full volume dump VTOC serialization

Concurrent Copy does not remove all data integrity exposures. For example, a DFSMSdss *full volume dump* serializes the VTOC of the source volume, but *does not* serialize the data sets on the volume. This ensures that the existing data sets are not deleted or extended, and new data sets are not allocated. However, there is an exposure in that the data in the existing data sets can be changed. Without Concurrent Copy, this exposure exists for the entire duration of the dump. With Concurrent Copy, the exposure exists only during initialization.

If you are using Concurrent Copy on VM-format volumes, DFSMSdss does not serialize VM data in any way.

If a dump requestor does not stop all updating of the data sets during the Concurrent Copy session initialization, the backup data integrity is compromised.

If a Concurrent Copy operation fails after signaling that the Concurrent Copy initialization was complete (and update activity on the data has resumed), it is not possible to recover the data at the point-in-time at which the Concurrent Copy operation was started. This is because the data may have been updated while the copy operation was progressing.

## 6.6 Scope of Concurrent Copy

You can think of Concurrent Copy from three different viewpoints:

**Host system:** From the perspective of the host, a Concurrent Copy session can include multiple data sets and span multiple volumes and storage controls. A data set can participate in multiple sessions. A session remains in effect until DFSMSdss transfers all tracks in the domain from the storage control to the host, and copies them to the backup media. There is no limit on the number of sessions that a host can initiate.

**Storage subsystem:** Within a given logical subsystem, a session begins when the SDM identifies entire volumes or range of tracks associated with the session, and ends when the SDM or ESS terminates the session. Each storage subsystem can participate in up to 64 simultaneous Concurrent Copy sessions, each of which may involve one or more volumes.

**Volume:** The SDM includes ranges of tracks for each volume in the domain. When the ESS has transferred all tracks in the range to the host system, the SDM removes those tracks from the Concurrent Copy session. A volume can participate in up to 16 sessions at a time. Ranges of tracks from multiple sessions may overlap.

## 6.7 Invoking Concurrent Copy

Concurrent Copy (CC) can be invoked by either the DFSMSdss or the DFSMShsm functions. IMS/ESA®, CICS, DFSMSrmm™, and DB2 can use CC for their backups.

DFSMSdss has the option of using Concurrent Copy when data sets are copied or dumped. To invoke Concurrent Copy, the **CONCURRENT** keyword must be specified on a DFSMSdss **COPY** or **DUMP** commands.

With DFSMShsm, system-managed data may be backed up automatically with Concurrent Copy by use of management class parameters. DFSMShsm also allows Concurrent Copy to be used when copying data using Aggregate Backup and Recovery Support (ABARS).

Using the **CONCURRENT** control statement of the DB2 COPY utility, you can invoke Concurrent Copy to make a full image copy. During recovery, DB2 can automatically use the most recent image copy and then apply records from the log.

Concurrent Copy requires the software support provided in DFSMS/MVS.

## 6.8 DFSMSdss invocation of other utilities

Only DFSMSdss uses Concurrent Copy, and Concurrent Copy can be used only when DFSMSdss moves the data. That is, Concurrent Copy cannot be used for those cases where DFSMSdss calls another utility function. The function itself, Concurrent Copy, is not aware of the data set organization.

DFSMSdss uses IDCAMS **REPRO** to make logical copies when the target device is different from the source, or if the target CISIZE, CASIZE, physical record size, physical block size, or imbed or span attributes are different from the source.

When copying partitioned load modules to an unlike device, DFSMSdss uses IEBCOPY with **COPYMOD** specified. This may result in a re-blocked data set. When copying to a like device, IEBCOPY with **COPY** specified is used.

But, if DFSMSdss invokes a utility such as IDCAMS (**REPRO**) or IEBCOPY for a data set copy operation, Concurrent Copy is not done for those data sets.

### In summary

Here are the main points from the foregoing discussion:

- If DFSMSdss invokes a utility such as IDCAMS or IEBCOPY for a data set copy operation, Concurrent Copy is not used for those data sets.
- The **CONCURRENT** keyword applies to all the data being dumped or copied by the function under which it is specified. It cannot be applied to a subset of the data being processed.
- To ensure data integrity, do not update the data during a Concurrent Copy initialization.
- If a Concurrent Copy operation fails after signaling that the Concurrent Copy initialization is complete (and update activity on the data has resumed), it is not possible to recover the data at the point-in-time at which the Concurrent Copy operation has started. This is because the data may have been updated while the copy operation was progressing.
- The **CONCURRENT** keyword cannot be used with the **DELETE, UNCATALOG,** or **FCWITHDRAW** keywords.
- The **CONCURRENT** and **FASTREPLICATION(REQUIRED)** keywords are mutually exclusive.

- The **RESET** keyword is ignored when the **CONCURRENT** keyword is also specified unless you use a patch to allow it.

Refer to the IBM publication *z/OS DFSMSdss Storage Administration Guide*, SC35-0423 for additional information on when DFSMSdss invokes a utility for a data set copy operation.

# 6.9 Concurrent Copy on the ESS

Concurrent Copy is initiated using the **CONCURRENT** keyword in DFSMSdss or in applications that internally call DFSMSdss as the copy program, for example, the DB2 COPY utility.

The System Data Mover establishes a Concurrent Copy *session* with the ESS. There can be up to 64 sessions active at a time (including sessions for Extended Remote Copy (XRC)) per ESS logical subsystem (LSS).

### Concurrent Copy and FlashCopy

Concurrent Copy and FlashCopy V1 or FlashCopy V2 can coexist in the same ESS.

If DFSMSdss is instructed to do a Concurrent Copy by specifying the CONCurrent (CC) keyword, and the copy is for a full volume (DFSMSdss COPY FULL command) the following will be honored based on whether FlashCopy V1 or FlashCopy V2 is installed:

- With the target within the same logical subsystem as the source and FlashCopy V1 installed, DFSMSdss will start a FlashCopy copy process instead of Concurrent Copy.
- With the target within the same ESS as the source and FlashCopy V2 installed, DFSMSdss will start a FlashCopy copy process instead of Concurrent Copy.

So, you get a FlashCopy invocation with a DFSMSdss COPY FULL command even if the CONCurrent (or CC) parameter is coded in the command. See also Figure 6-5 on page 290 for an example of the use of FlashCopy and Concurrent Copy together.

The FASTREPLICATION parameter of the COPY command does not affect Concurrent Copy, it only applies for FlashCopy and SnapShot invocation. FASTREPLICATION(REQuired) and CONCURRENT cannot be used together in the same COPY command. Refer to“FASTREPLICATION parameter” on page 50, for additional discussion.

# 6.10 Protecting DFSMSdss CC keyword with RACF

If you wish to protect access to DFSMSdss Concurrent Copy operations you can follow the procedures below. You will need to define FACILITY class resource profiles and restrict access to those profiles. For a given command or parameter, protection occurs when both of the following conditions are met:

- The RACF FACILITY class is active
- The indicated FACILITY class profile has been defined

In order to protect CONCURRENT with the DFSMSdss COPY function, you must define the profile name:

STGADMIN.ADR.COPY.CNCURRNT

In order to protect CONCURRENT with the DFSMSdss DUMP function you must define the profile name:

STGADMIN.ADR.DUMP.CNCURRNT

When the FACILITY class is active and the STGADMIN.ADR.COPY.CNCURRNT or STGADMIN.ADR.DUMP.CNCURRNT profiles are defined, you need READ access authority to them in order to use the indicated command or parameter. Failure to do this will allow anyone to use the CC parameter. If OS/390 Security Server, RACF element, Version 1 Release 7 or later is not installed, or if Facility Class checking is not set up for this keyword, any DFSMSdss user can use them.

## 6.11 Sizing and requirements

Concurrent Copy is supported with the IBM TotalStorage Enterprise Storage Server with the current supported releases of DFSMS, OS/390 and z/OS. Concurrent Copy requires the following components:

- DFSMSdfp for the SDM, management and storage class attributes, and ISMF support of those attributes
- DFSMSdss for the external interface to Concurrent Copy
- DFSMShsm to call Concurrent Copy during backup operations.

SMS is not necessary in order to use Concurrent Copy. You can use Concurrent Copy directly from DFSMSdss without SMS. However, DFSMShsm supports Concurrent Copy during automatic and aggregate backup (ABARS) only when SMS is active.

We recommend you review the following Web page for Concurrent Copy information and maintenance:

`http://www.storage.ibm.com/software/sms/sdm/index.html`

Additionally, running Concurrent Copy operation uses the following system resources.

- Central and expanded storage
- ESS cache
- Channel paths

### 6.11.1 Central and expanded storage

The SDM uses *central storage* for the *host sidefiles* that it maintains for Concurrent Copy sessions. The z/OS *real storage manager* may page sections of the sidefile into *expanded storage* in response to varying system loads.

The size of the host sidefile is a secondary consideration when using Concurrent Copy. The SDM uses the sidefile only to store temporary copies of tracks as part of the intercepting write process. The size and use of the sidefile relate directly to the number of intercepted writes. The size of the host sidefile is unlikely to be a significant factor unless you use Concurrent Copy during periods of very high update activity or in a very storage-constrained environment.

The SDM creates one host sidefile for each Concurrent Copy session. The host sidefiles for different Concurrent Copy sessions share a single data space, which can grow to a maximum size of 2GB. The actual size of the data space is, however, dynamic and changes in response to the SDM requirements. The size depends on the number of active Concurrent Copy sessions, the level of update activity, and the rate at which DFSMSdss reads data and writes it to the output media. In the unlikely event that the sidefiles will fill the data space, the SDM creates additional data spaces as necessary.

Another factor that affects the host sidefile size is the distribution of application writes across a device. I/O operations on a device typically concentrate on a set of tracks rather than uniformly distributing updates over the device. An application that updates a track causes an intercepted write. Subsequent updates to the same track do not result in an intercepted write. Similarly, as soon as DFSMSdss has processed a particular track, updates to that track do not result in an intercepted write. Concurrent Copy only uses the host sidefile when there are intercepted writes. Therefore, the grouping of I/Os by applications significantly reduces the potential maximum size of the host sidefile.

Running many Concurrent Copy jobs simultaneously can cause auxiliary (AUX) storage shortages. You can control the total amount of AUX storage that is used by Concurrent Copy jobs at any given time. The MVS System Resources Manager (SRM) uses two different percentages (a lower value and an upper value) to determine when AUX storage shortages have been reached. SDM provides the ability to modify the Concurrent Copy AUX values that are used in evaluating the AUX storage that is used by Concurrent Copy. During Concurrent Copy processing, SDM subtracts these values from the two MVS percentages to determine if the total current system AUX storage percentage being used by the system is above these newly computed SDM percentages

Example 6-1 shows how to modify the Concurrent Copy AUX delta to determine at what percentage value *new* Concurrent Copy data sets will not be copied using Concurrent Copy. SDM subtracts this delta value from the lower MVS percentage. The newly computed SDM percentage is then compared to the total current system AUX storage percentage being used. The default value is -1, indicating that SDM will not perform a percentage check for AUX storage usage by Concurrent Copy. The *nnnn* value may be any nonnegative value (including zero).

*Example 6-1 Modifying auxiliary storage delta for new Concurrent Copy data set*

```
F ANTMAIN,P .CMTUN+3A X'FFFF' X'nnnn'
```

Example 6-2 shows how to modify the Concurrent Copy AUX delta to determine at what percentage value *existing* Concurrent Copy data sets will be terminated. SDM subtracts this delta value from the upper MVS percentage. The newly computed SDM percentage is then compared to the total current system AUX storage percentage being used. The default value is -1, indicating that SDM will not perform a percentage check for AUX storage usage by Concurrent Copy. The *nnnn* value can be any nonnegative value (including zero).

*Example 6-2 Modifying auxiliary storage delta for existing Concurrent Copy data set*

```
F ANTMAIN,P .CMTUN+3C X'FFFF' X'nnnn'
```

### Guidelines for determining data space size

The Concurrent Copy support for the ESS uses data spaces to contain track image copies of the data being processed by the DFSMSdss.

z/OS data spaces are backed by *expanded storage* and *local paging* spaces. The amount of expanded storage and local paging space required for Concurrent Copy usage is dependent on a number of variables. Based on simulations and test scenarios, a typical data space size is about 10% of the amount of data being dumped or copied with Concurrent Copy.

If your data space size exceeds this nominal value, you may need to consider the following guidelines for determining how much expanded storage or local paging space may be required for the following Concurrent Copy functions:

- Full volume and tracks *copy*; as well as full volume, tracks, and physical data set *dump* operations:

  All volumes are processed on a track-by-track basis by DFSMSdss. The data space requirements can vary from 0% for a volume that has no updates during the DFSMSdss operation to 100% if the entire volume is updated before DFSMSdss can process it. For example, a 3390-3 that is 80% full (2671 cylinders) may require up to 2671 cylinders of data space storage if the volume is completely rewritten before DFSMSdss can process it. An example of this situation would be that a volume contains many VSAM data sets and a reorganization is done for all of the VSAM data sets on the volume while the Concurrent Copy job is being run for the volume.

- Logical data set *copy* and *dump* processing of non-VSAM data sets and non-indexed VSAM data sets (for example, VSAM ESDS), logical data set *copy* of indexed VSAM data sets (for example, VSAM KSDS), and logical data set *dump* of indexed VSAM data sets processed with **NOVALIDATE** are described as follows:

  These data sets are processed on a track-by-track basis by DFSMSdss. The data space is used to contain updates for tracks that have not yet been processed by DFSMSdss. The data space requirements can vary from 0% for a data set that has no updates during the DFSMSdss operation to 100% if the entire data set is updated before DFSMSdss can process it. For example, a 50-cylinder data set may require up to 50 cylinders of data space storage if the data set is completely rewritten before DFSMSdss can process it.

- Logical data set *dump* of indexed VSAM data set (for example, VSAM KSDS) processed with **VALIDATE** is described below:

  These data sets are processed with numerous accesses to sequence set information in the *index component* and track-by-track accesses to the *data component*. In all cases, update activity to either the data component or the index component maintains a copy of the updated track in the data space until the track is either processed by DFSMSdss or the dump operation is ended for all data sets.

  Index component tracks that do not contain sequence set information and data component tracks that are beyond the high used relative byte address are included in the Concurrent Copy operation but are never read by DFSMSdss. If those tracks are updated, they will remain in the data space for the duration of the dump operation for all data sets.

  If the data set has the sequence set information imbedded in the data component (using the **IMBED** attribute), no additional (non-updated) tracks are maintained in the data space. If the data set has the sequence set information in the index component, then all index component tracks containing sequence set information will be maintained in the data space (whether they were updated or not) for the duration of the dump processing for the data set. For example, if the index for a VSAM data set is 20 cylinders and the data is 2500 cylinders, you need to plan paging space of 20 cylinders for the index component.

  Based on the update activity during the dump operation, you should plan to use a paging space of between 0 and 2500 cylinders for the data. The most data space is used when doing a complete reorganization while dumping the VSAM data set. This requires 2520 cylinders of space. If only 10% of the data will change during the operation, you will need 20 cylinders for the index and 250 cylinders for the data or 270 cylinders of paging space.

In using Concurrent Copy against aggregate groups, determine the data space storage requirements based on the expected update rate to the data sets during the dump operations.

**Important:** Failure to allocate sufficient local paging space may result in system failures due to insufficient paging storage.

Note that all storage requirements will be in addition to the working set of storage required by all other applications active (including all other Concurrent Copy operations) during the execution of the DFSMSdss Concurrent Copy operation.

## 6.11.2 Cache

Concurrent Copy, by design, minimizes the amount of ESS *cache* that is used for sidefiles. Under normal circumstances, a Concurrent Copy session uses less than 0.5MB of cache. It is very unlikely that running a few simultaneous Concurrent Copy sessions will significantly affect the hit ratios that you normally achieve. It is, however, possible that the cache used by a Concurrent Copy session could have an effect on hit ratios. This is especially likely in cases where the amount of cache is only barely adequate for normal requirements. In those cases, you should increase the size of the cache regardless of whether or not you use Concurrent Copy.

Concurrent Copy uses several tactics to minimize the amount of cache that it uses:

- When reading data directly from DASD, the SDM and DFSMSdss inhibit the loading of tracks into the cache.
- When the cache sidefile holds no tracks, it occupies no cache space. A sidefile occupies cache storage only when it contains copies of tracks that are saved by the ESS during processing of *intercepted writes*.
- The SDM reads and deletes tracks from a cache sidefile as soon as it detects that they are present.

The ESS monitors the size of the cache sidefiles that an active Concurrent Copy session uses. In the highly unusual case of a system failure or software problem preventing the SDM from reading tracks, the ESS cancels active Concurrent Copy sessions when the sidefiles occupy more than half of the cache.

The ESS cancels sessions by starting with the session with the largest cache sidefile. This action preserves cache storage for use by sharing systems, but does not affect data integrity. Data is safe because the sidefiles contain only images of tracks before an update was applied. When Concurrent Copy is active, the subsystem maintains the same level of integrity for update operations that it maintains when Concurrent Copy is not active. If you cancel a Concurrent Copy session, you lose the copy. A Concurrent Copy session is not re-startable.

## 6.11.3 Channel paths

Because Concurrent Copy operations generate additional I/Os to the volumes involved, the use of Concurrent Copy always increases channel utilization, although this may not be a great concern when using FICON. The impact of using Concurrent Copy can vary, and depends on the number of simultaneous Concurrent Copy sessions, the DFSMSdss **`OPTIMIZE`** keyword, and the number of paths available.

### Pacing

You can tune the performance of a system by *pacing* the DFSMSdss read DASD I/O operations (**`READIOPACING`** parameter of the **`COPY`** and **`DUMP`** commands). Pacing reduces the channel utilization and lets other I/O (for example, from the database application) be processed in a more timely fashion. The pacing is done by waiting a specified amount of time before issuing each channel program that reads from DASD.

The System Data Mover dynamically controls pacing for Concurrent Copy I/O, so **`READIOPACING`** does *not* apply to Concurrent Copy I/O operations.

## 6.12 Production and performance considerations

When planning to use Concurrent Copy, you will be paying attention to system performance considerations such as:

- Application response time
- Storage subsystem utilization
- System throughput
- Concurrent Copy throughput

Your workload flow and your hardware configuration will determine how these factors affect your ability to use Concurrent Copy.

Also when planning for the production use of Concurrent Copy, some other things you will want to address are:

- When to schedule Concurrent Copy operations
- Where to use Concurrent Copy
- Number of simultaneous sessions (XRC and Concurrent Copy) that you can run

Let us look at these latter considerations in more detail.

### 6.12.1 When to schedule Concurrent Copy

Many factors influence when you can schedule Concurrent Copy operations. For example, the structure of your overnight batch processing determines at what stages in the processing you can use Concurrent Copy. Similarly, factors like availability of tape drives could restrict the intervals during which running a Concurrent Copy operation is feasible.

If other considerations are not a factor, you can use Concurrent Copy at times of lowest activity, especially lowest update activity. It may, however, be beneficial to use Concurrent Copy to back up data even during periods of higher I/O activity. In some cases, ensuring data availability may be more important than preserving levels of application performance.

### 6.12.2 Where to use Concurrent Copy

You can use Concurrent Copy to back up any data that can be backed up using DFSMSdss because DFSMSdss is the external interface to Concurrent Copy. The general-purpose design of Concurrent Copy simplifies the use of Concurrent Copy because it builds on existing experience with DFSMSdss.

For example, IMS databases can be backed up using DFSMSdss. During the recovery process, IMS database recovery control (DBRC) coordinates recovery of the DFSMSdss dump and the application of updates from the IMS log. The same is available also for DB2. Data sets that are consistently in use, such as DFSMShsm control data sets, databases, and libraries, require specialized facilities to ensure that data set backups are non-disruptive and preserve data set integrity.

Management class attributes let you choose how DFSMShsm and DFSMSdss should process data sets that are in use during availability management. Point-in-time capabilities using Concurrent Copy on the ESS allow you to:

- Use DFSMSdss to create a point of consistency backup of CICS/VSAM, IMS, or DB2 databases without needing to quiesce them during the entire backup process.
- Use DFSMSdss to create backups of data sets without requiring serialization during the entire backup process. DFSMSdss serializes the data during the Concurrent Copy *initialization* period (the time between the start of DFSMSdss and the issuing of the ADR734I message).

- Create and maintain multiple backup versions of DFSMShsm control data sets, while increasing the availability of DFSMShsm functions such as recall.
- Use the backup-while-open capability for CICS VSAM data sets with DFSMSdss in batch mode or with automated DFSMShsm, to provide backups with data integrity even when the data sets are being updated. Data integrity is assured for VSAM KSDSs even when CICS access results in control interval or control area splits or data set extends.

## 6.12.3 Concurrent Copy coexistence with XRC

Extended Remote Copy (XRC) is designed to work together with the ESS, thus efficiently managing the available resources. The SDM issues I/Os to efficiently drain from the cache records that are designated to be copied by XRC. Also with the XRC system data mover sessions, separate *storage control sessions* (SC sessions) can be defined to more effectively manage the remote copy environment.

Concurrent Copy and XRC both use ESS cache. If you run XRC and start up heavy Concurrent Copy activity, the cache may become rapidly consumed. In this case, Concurrent Copy activity may become canceled, and XRC may start to pace primary updates. You may instead want to run Concurrent Copy during low XRC update activity periods, or else run Concurrent Copy against the secondary volumes of the XRC pair.

Equip all the storage subsystems with adequate cache and channel connections to handle the work associated with the XRC sessions and the coexisting Concurrent Copy sessions. If adequate resources are not provided, the data mover may not be able to empty cache rapidly enough, and the cache could become overcommitted.

## 6.12.4 Simultaneous Concurrent Copy sessions

As said before, each Concurrent Copy session generates additional channel load and increases utilization of the storage paths within the ESS. If you are running multiple simultaneous Concurrent Copy sessions it is possible you may generate contention for these resources, resulting in greater than normal queueing and extended response times.

## 6.12.5 Number of data mover-allowed sessions

In addition to the server and storage resources needed for running multiple SDM sessions, you must consider the following design allowable maximums:

- 16 data mover sessions per device (combined total of Concurrent Copy and/or XRC)
- 64 data mover sessions per ESS logical subsystem (combined total of Concurrent Copy and/or XRC)
- 1024 data mover sessions per ESS (16 LSSs times 64)

If you attempt to use more than 16 data mover sessions per device, or more than 1024 data mover sessions per ESS *at the same time*, you will receive an error message, and the copy will continue using the traditional DFSMSdss copy.

### Sessions for aggregate group

If you relate Concurrent Copy processing to the **ABACKUP** command for an aggregate group that includes numerous data sets that are on multiple devices and LSS, a separate Concurrent Copy session is created for each LSS that has volumes containing data defined by the aggregate.

## 6.13 Diagnostic aids

The Concurrent Copy address space, ANTMAIN, is automatically started during system IPL. The data mover automatically re-initializes this address space if the operator cancels it, or if the address space ends abnormally.

There are conditions that prevent the system from starting this address space at IPL time, or after an operator cancels the address space. For such cases, you can restart the ANTMAIN address space by submitting the following program:

```
//STARTANT JOB MSGLEVEL=(1,1),REGION=4096K
//STEP1    EXEC PGM=ANTSTRT
```

For monitoring and diagnosing Concurrent Copy operation, you can use SMF records analysis and the z/OS MODIFY system command.

### 6.13.1 Diagnosing SDM with the MODIFY command

Use the z/OS MODIFY system command options that are described in this paragraph to diagnose and repair Concurrent Copy. Diagnostic procedures outlined in this section generate messages that begin with **ANTX89**. The publication *z/OS V1R4.0 MVS System Messages, Vol. 1 (ABA-AOM),* SA22-7631 describes the system data mover messages (which begin with **ANT**) and error codes.

The MODIFY command is abbreviated as F. The format of the MODIFY command function is as follows:

**F ANTAS00x,operation (*optional_operands*)**

The following is the list of valid operations for Concurrent Copy:

- **DUMP**
- **DVCDATA**
- **LISTSESS**
- **RCVRSESS**
- **SCDATA**
- **TERMSESS**

Refer to the publication *z/OS DFSMS Advanced Copy Services,* SC35-0428, for detailed discussion of the MODIFY command for diagnosing the *system data mover* functions.

### 6.13.2 SMF information

The SDM writes a system management facility (SMF) type 42 subtype 4 record that contains session statistics for each Concurrent Copy session when the session ends. Concurrent Copy records contain the identifier **CC**. Among other things, you can use the information in this record to determine the following statistics:

- Session initialization time
- Maximum size of host cached storage subsystem sidefiles
- Number of intercepted writes

The SMF type 42 subtype 4 record is described in detail in the publication *z/OS DFSMS Advanced Copy Services,* SC35-0428.

## 6.14 Examples of Concurrent Copy invocation

This section illustrates some examples of how Concurrent Copy can be invoked for either:

- Full volume dump
- Logical data set dump
- Logical data set copy
- Full volume copy invoking FlashCopy

Also discussed in this section is the invocation of Concurrent Copy from an application program.

### 6.14.1 DFSMSdss: Dump and copy examples

Figure 6-2 shows a DFSMSdss *full volume dump* using Concurrent Copy. No special action is required to perform a restore operation afterwards.

```
//DSSJOB JOB ...
//DUMPSTEP EXEC PGM=ADRDSSU
//SYSPRINT DD    SYSOUT=*
//DASD     DD    UNIT=SYSDA,VOL=SER=(SSDASD),DISP=OLD
//TAPE     DD    UNIT=TAPE,VOL=SER=(TAPE01,TAPE02,TAPE03),LABEL=(1,SL),
//  DISP=(NEW,KEEP),DSN=USER.BACKUP
//SYSIN    DD    *
       DUMP FULL INDDNAME(DASD) OUTDDNAME(TAPE) -
                 COMPRESS CONCURRENT
/*
```

*Figure 6-2 Full-volume dump operation with CONCURRENT*

Figure 6-3 shows a DFSMSdss *logical data set dump* of three fully qualified data sets using Concurrent Copy. No special action is required to perform a restore operation after a Concurrent Copy dump operation.

```
//JOB6     JOB ...
//DUMPSTEP EXEC PGM=ADRDSSU
//SYSPRINT DD    SYSOUT=*
//TAPE     DD    UNIT=TAPE,VOL=SER=(TAPE01,TAPE02,TAPE03),LABEL=(1,SL),
//  DISP=(NEW,KEEP),DSN=USER.BACKUP
//SYSIN    DD    *
       DUMP DATASET(INCLUDE(USER.LOG,USER.TABLE,USER.XREF)) -
            OUTDDNAME(TAPE) OPTIMIZE(4) CONCURRENT
/*
```

*Figure 6-3 Logical data set dump operation with CONCURRENT*

Figure 6-4 shows a DFSMSdss *logical data set copy* using Concurrent Copy.

```
//DSSJOB JOB ...
//COPYSTEP EXEC PGM=ADRDSSU
//SYSPRINT DD   SYSOUT=*
//SYSIN    DD   *
       COPY DATASET(INCLUDE(USER.LOG,USER.TABLE,USER.XREF)) -
            OUTDYNAME(OVOL01,OVOL02,OVOL03,OVOL08) -
            ALLDATA(*) ALLEXCP CONCURRENT -
            STORCLAS(BACKUP) RENAMEUNCONDITIONAL(USERX)
/*
```

*Figure 6-4 Data set Copy with CONCURRENT*

## 6.14.2 DFSMSdss: FlashCopy example

Figure 6-5 shows an example of the use of FlashCopy and Concurrent Copy together. It is a *full volume copy* between two volumes in the same LSS. As such, they were eligible for a FlashCopy Version 1 copy, and FlashCopy was used as shown by the message ADR806I. At the same time, the CC keyword is specified so you get both **ADR806I** and **ADR734I** Concurrent Copy initialization successful messages.

```
//STEPT40 EXEC PGM=ADRDSSU
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=V,OUTLIM=3000
//SYSIN   DD *
//*
COPY FULL -
   INDYNAM ((CP11S3)) -
     OUTDYNAM ((TP11S3)) -
   COPYVOLID -
     CC
ADR101I (R/I)-RI01 (01), TASKID 001 HAS BEEN ASSIGNED TO COMMAND 'COPY '
ADR109I (R/I)-RI01 (01), 2000.107 16:09:59 INITIAL SCAN OF USER CONTROL STATEMENTS
COMPLETED.
ADR016I (001)-RI01 (01), RACF LOGGING OPTION IN EFFECT FOR THIS TASK
ADR006I (001)-STEND(01), 2000.107 16:10:00 EXECUTION BEGINS
ADR241I (001)-DDTFP(01), TARGET VTOC BEGINNING AT 000550:0000 AND ENDING AT 000550:0014
IS OVERLAID
ADR806I (001)-TOMI (02), VOLUME COPIED USING A FAST REPLICATION FUNCTION.
ADR734I (001)-TOMI (03), 2000.107 16:10:12 CONCURRENT COPY INITIALIZATION SUCCESSFUL FOR
VOLUME CP11S3. SERIALIZATION FOR THIS DATA
                         IS RELEASED IF DFSMSDSS HELD IT. THE INTERMEDIATE RETURN CODE
IS 0000.
ADR320I (001)-SBRTN(01), VOLUME SERIAL TP11S3 ON UNIT D70C IS CHANGED TO CP11S3
ADR344I (001)-SBRTN(01), VOLSER ON UCB D70C IS A DUPLICATE. VOLUME MADE UNAVAILABLE.
ADR006I (001)-STEND(02), 2000.107 16:10:35 EXECUTION ENDS
ADR013I (001)-CLTSK(01), 2000.107 16:10:35 TASK COMPLETED WITH RETURN CODE 0000
ADR012I (SCH)-DSSU (01), 2000.107 16:10:35 DFSMSDSS PROCESSING COMPLETE. HIGHEST RETURN
CODE IS 0000
```

*Figure 6-5 Using FlashCopy and Concurrent Copy together*

### 6.14.3 Invocation from an application program

When DFSMSdss is invoked from an application program, you can use the *user interaction module* (UIM) to interact with DFSMSdss. For user interactions to take place, the application must invoke DFSMSdss and must provide a pointer to a user interaction module (UIM) list. DFSMSdss can be invoked by any of the following system macros:

```
ATTACH EP=ADRDSSU,PARAM=(OPTPTR,DDPTR,PAGEPTR,UIMPTR,UAPTR),VL=1
LINK EP=ADRDSSU,PARAM=(OPTPTR,DDPTR,PAGEPTR,UIMPTR,UAPTR),VL=1
CALL (15),(OPTPTR,DDPTR,PAGEPTR,UIMPTR,UAPTR),VL
```

When a UIM exit routine is specified, DFSMSdss processes normally, then at each point in the process (DFSMSdss exit points), the UIM exit routine is called conditionally to allow some types of user operations. The exit identification block is passed to the UIM every time DFSMSdss gives control to it.

Among the exit points available with DFSMSdss (Eioptions 1 to 26), DFSMSdss calls the UIM with option code 24 (Eioption 24) to inform it that the initialization of the Concurrent Copy session for a given data set or volume has completed. For full volume or tracks operation, there is only one call (because there is only one input volume). For a physical data set operation, there is one call for each input volume. For a logical data set operation, there is one call for every data set.

DFSMSdss does not call the UIM with this option code if the **CONCURRENT** keyword is *not* specified. DFSMSdss provides the UIM with information through the EIREC24 structure within the *exit identification block*, ADREIB (in the ADREID0 macro).

For a detailed description of the DFSMSdss interaction with the user interface module (UIM) when invoking from an application program, refer to the publication *z/OS V1R3 DFSMSdss Storage Administration Reference,* SC35-0424.

### 6.14.4 Installation options exit

Usage of the Concurrent Copy function can also be controlled through the installation options exit, a product-sensitive programming interface intended for users. Refer to Options Installation Exit Routine (ADRUIXIT) described in *z/OS V1R3.0 DFSMS Installation Exits*, SC26-7396, for more information.

## 6.15 Concurrent Copy with DFSMShsm

This section gives an introduction on how Concurrent Copy and DFSMShsm work together. For more detailed information you can refer to the following IBM publications:

- *z/OS DFSMSdss Storage Administration Guide*, SC35-0423
- *DFSMShsm Storage Administration Guide*, SC35-0388

Provided that the requirements are met, DFSMShsm automatically uses Concurrent Copy. There are no special DFSMShsm commands to control whether or not Concurrent Copy is used.

DFSMShsm uses Concurrent Copy in the following situations:

1. Backup of a specific DFSMShsm data set:
   - The BACKDS command creates a backup version of a specific data set. DFSMShsm supports a CC keyword which can be used to specify if you want to use Concurrent Copy during BACKDS processing to provide a point-in-time backup copy.
   - The Inline Backup program also supports Concurrent Copy.
2. During automatic, volume, or command data set backup processing:
   - DFSMS provides a BACKUP COPY TECHNIQUE attribute for the SMS management class that informs DFSMShsm whether or not to use the Concurrent Copy facility when backing up data sets during automatic, volume, or command data set backup processing.
     - `CONCURRENT REQUIRED` specifies that Concurrent Copy is required. If a Concurrent Copy session cannot be established, the data sets will not be backed up.
     - `CONCURRENT PREFERRED` specifies that Concurrent Copy is preferred. If a Concurrent Copy session can be established, the data sets are backed up using Concurrent Copy. Otherwise, DFSMSdss logical dump (without Concurrent Copy) is used.
     - `STANDARD` specifies that Concurrent Copy is not to be used. DFSMSdss logical dump without Concurrent Copy processing is used.
   - The DFSMS storage class ACCESSIBILITY attribute allows you to direct allocation of new data sets to DASD volumes that belong to an ESS, using either FlashCopy or Concurrent Copy. In order to request Concurrent Copy you assign system-managed data sets a storage class with the ACCESSIBILITY attribute set to CONTINUOUS or CONTINUOUS PREFERRED.
3. Backup of DFSMShsm CDS:
   - You can use Concurrent Copy to back up the control data sets. The use of Concurrent Copy significantly reduces the duration of complex-wide serialization of the DFSMShsm control data set resources when performing CDS version backup. This reduction increases the availability of DFSMShsm functions.
   - You can back up the CDSs and the journal using Concurrent Copy by allocating them on SMS managed ESS devices (does not apply to the journal) and specifying DATAMOVER(DSS). Ensure that you associate any CDS backup volumes with the management class BACKUP COPY TECHNIQUE attribute of CONCURRENT REQUIRED or CONCURRENT PREFERRED.

If Concurrent Copy was used to back up the data set and a system failure or cancel of DFSMShsm occurred after the Concurrent Copy session was established, then the data-set-changed indicator can be off in the VTOC even though the data set was not successfully backed up. This can result in the data set prematurely meeting the expiration criteria or migration will not identify the data set as needing a backup copy.

For more information on using Concurrent Copy with DFSMShsm, refer to publication *z/OS DFSMShsm Storage Administration Guide*, SC35-0388.

## 6.16 Concurrent Copy with DFSMSrmm

You can use Concurrent Copy with DFSMSrmm CDS backup to enable DFSMSrmm to continue to process requests while the backup is taken. If you plan to use DFSMSdss to back up the control data set, place the control data set on a Concurrent Copy enabled volume.

To create a backup copy using EDGHSKP, specify the BACKUP parameter. You can submit a job to back up the control data set to tape with JCL as shown in Example 6-3.

*Example 6-3 JCL for backing up the control data set to tape*

```
//BACKUP   JOB ...
//*
//STEP1    EXEC PGM=EDGHSKP,PARM='BACKUP(DSS)'
//SYSPRINT  DD  SYSOUT=*
//MESSAGE   DD  DISP=SHR,DSN=RMM.MESSAGE
//BACKUP    DD  DISP=(,CATLG),UNIT=TAPE,DSN=BACKUP.CDS(+1),
//          LABEL=(,SL)
//JRNLBKUP  DD  DISP=(,CATLG),UNIT=TAPE,DSN=BACKUP.JRNL(+1),
//          DCB=(RECFM=VB,BLKSIZE=0,LRECL=9000),LABEL=(2,SL),
//          VOL=REF=*.BACKUP
//DSSOPT    DD  *
  CONCURRENT OPTIMIZE(1) VALIDATE
/*
```

In the example above:

- Specify PARM='BACKUP(DSS)' in the EXEC statement if you want to use DFSMSdss to perform the backup. DFSMSdss will invoke Concurrent Copy for the backup if the data sets to be backed up are on a Concurrent Copy enabled volume.
- JRNLBKUP DD statement is specified only if you want to backup the journal.
- DSSOPT DD statement is optional and allows you to customize the DFSMSdss DUMP and RESTORE commands. The names of the control data set and journal are obtained from the running DFSMSrmm subsystem and should not be specified in the JCL.

For more information on using Concurrent Copy with DFSMSrmm, refer to publication *z/OS DFSMSrmm Implementation and Customization Guide*, SC26-7405.

# 6.17 Concurrent Copy with DB2

This section gives an overview of how Concurrent Copy and DB2 work together. For detailed information on using Concurrent Copy with DB2, refer to publication *DB2 UDB for OS/390 and z/OS V7 Utility Guide and Reference*, SC26-9945.

## 6.17.1 Implementing Concurrent Copy with DB2 V7

The DB2 **COPY** online utility creates up to four image copies of any of the following objects:

- Tablespace
- Tablespace partition
- Data set of a linear tablespace
- Index space
- Index space partition

The image copies created can be a full image copy of all pages in a tablespace, partition, data set, or index space, or an incremental image copy of pages that have been modified since the last use of the COPY utility. The copies are used by the RECOVER utility when recovering a tablespace or index space to the most recent time or to a previous time. Copies can also be used by MERGECOPY, RECOVER, COPYTOCOPY and UNLOAD.

The CONCURRENT option of the COPY utility invokes Concurrent Copy to make the full image copy. When the CONCURRENT option is used:

- You must specify either a COPYDDN DD statement, a RECOVERYDDN DD statement, or both.
- The resulting image copy is recorded in the catalog table SYSIBM.SYSCOPY with ICTYPE=F and STYPE=C.
- If the SYSPRINT DD statement points to a data set, you must use a DSSPRINT DD statement.
- The SHRLEVEL CHANGE option is not allowed when you use Concurrent Copy for tablespaces having a page size greater than 4 KB.

To obtain a consistent offline backup:

1. To ensure that no updates to data occur during backup, start the DB2 objects being backed up for read-only access by issuing the following command:

   **START DATABASE***(database-name)* **SPACENAM(***tablespace-name)* **ACCESS(RO)**

2. Run QUIESCE with the WRITE(YES) option to quiesce all DB2 objects being backed up.
3. Back up the DB2 data sets if the QUIESCE utility completes successfully.
4. Issue the following command to allow transactions to access the data:

   **START DATABASE(***database-name***) SPACENAM(***tablespace-name***)**

Restrictions on use of DFSMS Concurrent Copy:

- You cannot use a copy made with Concurrent Copy with the RECOVER utility PAGE or ERRORRANGE options. If you specify PAGE or ERRORRANGE, RECOVER bypasses any Concurrent Copy records when searching the SYSCOPY table for a recoverable point.
- You cannot use the CONCURRENT option with SHRLEVEL CHANGE on a tablespace with 8KB, 16KB or 32KB page size.
- You cannot run the following DB2 stand-alone utilities on copies made by Concurrent Copy:
  - DSN1COMP
  - DSN1COPY
  - DSN1PRNT
- You cannot invoke the CONCURRENT option from the DB2I Utilities panel or from the DSNU TSO list.
- If you specify COPY SHRLEVEL REFERENCE with the CONCURRENT option, and if you want to copy all of the data sets for a list of tablespaces to the same output device, specify FILTERDDN in your COPY command to improve tablespace availability. In this scenario, specifying COPY without the FILTERDDN option forces DFSMS to process the list of tablespaces sequentially, which might limit the availability of some of the tablespaces being copied.

### 6.17.2 Examples

In this section we show some examples of how DFSMS Concurrent Copy may be invoked in a DB2 environment.

Example 6-4 shows how to copy a list of tablespaces, using the CONCURRENT option to invoke Concurrent Copy. Update activity is allowed during the COPY operation.

*Example 6-4 Invoking Concurrent Copy with COPY utility*

```
//STEP1    EXEC DSNUPROC,UID='IUJMU111.COPYLST',UTPROC='',
//         SYSTEM='V61A',DB2LEV=DB2A
//COPY1    DD DSN=IUJMU111.COPYLST.STEP1.TS1,
//         DISP=(MOD,CATLG,CATLG),UNIT=SYSDA,
//         SPACE=(4000,(20,20),,,ROUND)
//COPY2    DD DSN=IUJMU111.COPYLST.STEP1.TS2,
//         DISP=(MOD,CATLG,CATLG),UNIT=SYSDA,
//         SPACE=(2000,(20,20),,,ROUND)
//COPY3    DD DSN=IUJMU111.COPYLST.STEP1.TS3,
//         DISP=(MOD,CATLG,CATLG),UNIT=SYSDA,
//         SPACE=(2000,(20,20),,,ROUND)
//SYSIN    DD *
   COPY TABLESPACE DBAU2901.TPAU2901
   COPYDDN(COPY1)
   TABLESPACE DBAU2901.TLAU2902
   COPYDDN(COPY2)
   TABLESPACE DBAU2901.TSAU2903
   COPYDDN(COPY3)
   CONCURRENT SHRLEVEL CHANGE
```

Example 6-5 shows how to copy a list of tablespaces, using the CONCURRENT and FILTERDDN options to create a single DUMP command for DFSMS Concurrent Copy, allowing maximum availability.

*Example 6-5 Invoking Concurrent Copy with COPY utility using filter*

```
//COPY     EXEC DSNUPROC,SYSTEM=V61A
//SYSCOPY  DD DSN=CONCOPY.WFILT,DISP=(MOD,CATLG,DELETE),
//         UNIT=SYSDA,SPACE=(CYL,(42,5),RLSE)
//FILT     DD DSN=FILT.TEST1,DISP=(MOD,CATLG,DELETE),
//         UNIT=SYSDA,SPACE=(CYL,(1,1),RLSE)
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
   COPY TABLESPACE TS1
   TABLESPACE TS2
   TABLESPACE TS3
   FILTERDDN(FILT)
   COPYDDN(SYSCOPY)
   CONCURRENT
   SHRLEVEL REFERENCE
```

# 6.18 Concurrent Copy with IMS

This section gives an overview of how Concurrent Copy and IMS work together. For detailed information on using Concurrent Copy with IMS, refer to the publication *IMS V7 Utilities Reference: Database and Transaction Manager*, SC26-9440.

## 6.18.1 Database Image Copy 2 utility (DFSUDMT0)

IMS provides a utility, the Database Image Copy 2 utility (DFSUDMT0), which performs the tasks required when DFSMS Concurrent Copy is used to take image copies of IMS databases. DFSUDMT0 invokes DFSMSdss DUMP to copy an IMS database data set, enabling the user to obtain an image copy created with Concurrent Copy in a manner similar to running a batch image copy.

An image copy created by the Database Image Copy 2 utility is in DFSMSdss dump format, rather than standard batch image copy format. The copy is registered with DBRC as an SMSNOCIC or SMSCIC image copy, depending on the parameters specified when the image copy was taken. If more than two output copies are created, only the first two are registered with DBRC. The output from the DFSUDMT0 utility is used as input to the Database Recovery utility.

By using the DFSMS Concurrent Copy function, the Database Image Copy 2 utility increases database availability. The utility can copy a database that is either stopped or active. If the database is stopped, it can be restarted after the logical copy is complete, and database updating can continue. The database is available to IMS without waiting for the physical copy to be completed. Note that nonrecoverable databases must be stopped before the utility is run. You have the option to wait until the physical copy is complete before releasing the database for update. This is useful in cases where an image copy is required for a specific purpose (such as end-of-month processing). In this case, it is safer to wait for the physical copy before restarting the database.

This utility is only available for databases that are registered with DBRC.

Example 6-6 shows how you can use invoke the Database Image Copy 2 utility.

*Example 6-6 Invoking Database Image Copy 2 utility*

```
//TOCOPY   JOB ...
//STEP1    EXEC   PGM=DFSRRC00,PARM='ULU,DFSUDMT0'
//STEPLIB  DD     DSN=IMS.RESLIB,DISP=SHR
//DFSRESLB DD     DSN=IMS.RESLIB,DISP=SHR
//IMS      DD     DSN=IMS.DBDLIB,DISP=SHR
//RECON1   DD     DSN=RECON1,DISP=SHR
//RECON2   DD     DSN=RECON2,DISP=SHR
//RECON3   DD     DSN=RECON3,DISP=SHR
//SYSPRINT DD     SYSOUT=A
//DBIN     DD     DSN=IMS.DBIN,DISP=SHR
//OUTPUTD1 DD     DSN=IMS.DBAOUT1,DISP=(NEW,KEEP),
//                UNIT=TAPE,VOL=SER=DMPNO1,LABEL=(,SL)
//OUTPUTD2 DD     DSN=IMS.DBAOUT2,DISP=(NEW,KEEP),
//                UNIT=TAPE,VOL=SER=DMPNO1,LABEL=(,SL)
//OUTPUTD3 DD     DSN=IMS.DBAOUT3,DISP=(NEW,KEEP),
//                UNIT=TAPE,VOL=SER=DMPNO1,LABEL=(,SL)
//OUTPUTD4 DD     DSN=IMS.DBAOUT4,DISP=(NEW,KEEP),
//                UNIT=TAPE,VOL=SER=DMPNO1,LABEL=(,SL)
//SYSIN    DD     *
   4 DBDNAMEX DBIN     OUTPUTD1 OUTPUTD2 OUTPUTD3 OUTPUTD4 XL
```

## 6.18.2 IMS Fast Path DEDB

Although the commands used for fast path are different from those used with the Database Image Copy 2 utility, the concept remains the same:

- You need to stop the area, initialize the Concurrent Copy session, and restart the area at logical completion.
- You will get a point-in-time copy, while DEDBs would be offline from IMS for a minimum time.
- You could then proceed with updates while the copy is being made, and the physical completion will indicate when the copy is done.

### 6.18.3 Backup-while-open of IMS data sets

DFSMSdss supports backup-while-open processing of IMS data sets. Backup-while-open serialization is applicable for HISAM, SHISAM, and index (primary and secondary) databases.

Backup as an open data set for IMS is triggered through a UIM request. Note that VALIDATE must be specified to ensure that an IMS backup-while-open data set, that is dumped while updates are being made, can be successfully restored. VALIDATE allows DFSMSdss to validate and correct the data set during the dump process, or to end the dump (with an `ADR943E` message).

DFSMSdss COPY with DELETE, DUMP with DELETE, and RESTORE with REPLACE must be used with care. These commands can cause irreparable damage to an IMS data set for the following reasons:

- The enqueue serialization obtained by COPY and DUMP are insufficient to ensure that the data set is not also being used by an IMS application in an environment where GRS (or equivalent) is not being used.
- COPY and DUMP do not provide any special handling for any data set defined as BWO(TYPEIMS).
- RESTORE does not provide any protection against reallocating or overwriting any IMS data set for which RESTORE is able to obtain an enqueue serialization on SYSDSN and SYSVSAM.

## 6.19 Concurrent Copy with CICS

This section gives an overview of how Concurrent Copy and CICS work together. For detailed information refer to the IBM publication *CICS Recovery and Restart Guide*, SC33-1698.

### 6.19.1 Using CICS DFSMS Concurrent Copy support

The Concurrent Copy function works along with CICS backup-while-open (BWO) support.

Concurrent Copy can be used to backup VSAM files being used by CICSTS, but only if the files are closed to CICS while the backup is being made. With BWO, the backup can be made while the files are still being accessed by CICS transactions. Backups taken with BWO are accepted as input by the CICSVR forward recovery program.

In order to use Concurrent Copy, specify the CONCURRENT keyword when you use DFSMShsm to dump BWO data sets.

### 6.19.2 Backup-while-open

The backup-while-open (BWO) function allows backups to be taken by DFSMSdss when applications are running in continuous operation while the data set is open for update with full data integrity of copied data. This is feasible only for CICS VSAM file control data sets for which CICS creates forward-recovery logs.

The Concurrent Copy function used along with BWO by DFSMSdss allows backups to be taken with integrity even when control-area and control-interval splits and data set additions (new extents or add-to-end) are occurring for VSAM key sequenced data sets.

BWO is available only for user files accessed by CICS File Control and for the CICS system definition (CSD) file. Considerations for BWO:

- VSAM data sets that are to use this facility must reside on SMS-managed DASD, and must have an ICF catalog structure.
- Only VSAM ESDS, RRDS, and KSDS data sets are supported. ESDS and KSDS are supported both with and without alternate indexes.
- BWO is supported at the VSAM sphere level. You cannot take BWO copies of some sphere component and not others. The first data set opened for update against a VSAM base cluster determines the BWO eligibility for the sphere.
- CICS defines a data set as eligible for BWO when the FCT is defined using RDO. If BACKUPTYPE=DYNAMIC is specified for a VSAM file, the file is defined as eligible for BWO when the data set is opened. BACKUPTYPE=STATIC, the default, defines a file as not eligible for BWO.
- If DFSMShsm and DFSMSdss is to backup a data set while it is open, and the FCT specifies BACKUPTYPE=STATIC, all CICS files currently open for update against that data set must be closed before the backup can start.

Backup-while-open is a better method than using SHARE or TOLERATE(ENQFAILURE) for backing up CICS VSAM file-control data sets that are in use and open for update. When you dump data sets that are designated by CICS as eligible for BWO processing, data integrity is maintained through serialization interactions between CICS, CICSVR, VSAM record management, DFSMSdfp, and DFSMSdss. BWO processing ensures that any update activity that may invalidate the dump is detected. Simultaneous recovery or delete of the data set while it is being dumped is also prevented.

There are certain restrictions on using BWO with VSAM KSDS data sets:

- If you use neither Concurrent Copy nor BWO, you must close your files, lose the current copy of any CICS maintained data tables, and keep them closed for however long it takes until the copy is complete.
- If you use Concurrent Copy but not BWO, you must close your files and therefore lose the current copy of a CICS maintained data table. The file only needs to remain closed while the Concurrent Copy session is set up.
- If you use BWO but not Concurrent Copy, you do not need to close your files. Therefore, you do not lose the current copy of any data table. However, your copy will be invalidated if a CI or CA split occurs while the copy is being taken.
- If you use both BWO and Concurrent Copy, you do not need to close your files so you do not lose the current version of your data table. Your copy is only invalidated if a CI or CA split occurs during the short time it takes to set up the Concurrent Copy session.

At the end of the backup, DFSMShsm and DFSMSdss check the ICF catalog, and if a split has occurred or is still in progress, will discard the backup. Thus, certain heavily-updated VSAM KSDS data sets may not be suitable candidates for BWO, or might be eligible only during periods of reduced activity.

# 7

# ESS Copy Services Web User Interface

This chapter reviews the ESS Copy Services Web User Interface (WUI) panels and ESS Copy Services server (CSS) characteristics. The descriptions and figures presented in this chapter are based on an ESS with LIC 2.3.0.

Detailed information on how to use the ESS Copy Service WUI can be found in the publication: *IBM TotalStorage Enterprise Storage Server Web Interface User's Guide,* SC26-7448.

Examples of common procedures are given as well as how to display status information. It is assumed that most zSeries users will prefer to use the tools discussed in Chapter 4, "Managing Copy Services" on page 145, so only a brief overview of the Web User Interface tasks are given here. For in-depth detailed descriptions of using the Web User Interface to configure and manage Copy Services on ESS, refer to the companion redbook, *IBM TotalStorage Enterprise Storage Server Implementing ESS Copy Services in Open Environments,* SG24-5757.

## 7.1 ESS Copy Services

The IBM TotalStorage Enterprise Storage Server provides an ESS Copy Services Web User Interface (WUI) to manage the PPRC and FlashCopy functions. ESS Copy Services run on each ESS cluster of an ESS Copy Services server group (CSS). For an overall understanding of the ESS Copy Services, the following concepts should be understood:

### *Prior to LIC 2.2.0*

Prior to LIC 2.2.0, the following features were available:

**Copy Services server group:** The Copy Services server group (also referred to as a Copy Services Domain) is a collection of ESS clusters participating in the Copy Services functions managed by an active Copy Services server (CSS).

**Active Copy Services server:** The active Copy Services server is the ESS cluster that is currently running the Copy Services server functions. This would normally be the primary Copy Services server, unless it fails, in which case you would invoke the backup Copy Services server as the active Copy Services server.

**Backup Copy Services server:** The backup Copy Services server is the ESS cluster in the Copy Services server group that is designated to run the Copy Services server functions when the primary Copy Services server fails. An operator must manually restart it in the event of a failure.

**Primary Copy Services server:** The primary Copy Services server is an ESS cluster in the Copy Services server group that performs the role of the active Copy Services server.

**Copy Services client**: A Copy Services client is the function that runs on each cluster in your Copy Services server group.

### *LIC 2.2.0 or higher*

With LIC 2.2.0 and higher, the Active / Backup server relationship was replaced with Dual Active Server. Both servers are active and can action requests. Configuration changes are shared between the servers.

**ServerA:** The Copy Services ServerA. This is the ESS cluster that is running ServerA.
**ServerB:** The Copy Services ServerB. This is the ESS cluster that is running ServerB.

With LIC 2.2.0 and above, the following enhancements were made available:

- Improved availability with the option to configure dual active copy services servers, thereby allowing either server to initiate tasks or monitor domain status.
- User capability to reconfigure the domain without CE assistance.
- Domain-wide Reset capability.

With LIC 2.3.0 and above, the following enhancements were added:

- Copy Services Domain support for up to eight ESSs, allowing additional flexibility.
- Copy Services Web enhancements offering server status notification. An alert is generated from the Copy Services server to the ESS to notify you that the Copy Services server is down or not responding.

Dual active Copy Services servers (ServerA and ServerB) can be configured in a Copy Services server group. On each ESS cluster that is part of a Copy Services Domain, there is a Copy Services *client* running that communicates to the active server. Therefore, an Ethernet connection between all of the related ESSs is necessary so that the server and clients can communicate.

You can configure up to eight ESSs within a Copy Services Domain. Each Copy Services Domain manages a maximum of 2048 relationship-pairs (FlashCopy and PPRC pairs).

### 7.1.1 ESS Master Console

Access to the ESS Copy Services can be done through an Internet browser. Using a Web browser gives you the possibility to easily control the ESS Copy Services functionality from the network.

The user Web browser can be running on any workstation connected to the ESS or by using using the ESSNet. Early machines were shipped with an ESSNet to provide browser access to the ESS; the IBM representative sets it up when installing your ESS.

The ESS Master Console replaces the ESSNet Console and as well as providing a browser; it also includes service connection tools and remote support facility connectivity. For additional information on the ESS Master Console and the ESSNet, refer to the following publications: *IBM TotalStorage Enterprise Storage Server User's Guide,* SC26-7445, and *IBM TotalStorage Enterprise Storage Server Introduction and Planning Guide,* GC26-7444.

### 7.1.2 Web browser user interface

You can use the Web browser that comes with the ESS Master Console to connect to the ESS Copy Services server, or you can also run your Web browser from your own workstation. To do the last one, you either connect your workstation directly to the ESSNet hub, or you connect your workstation to your intranet and connect your intranet to the ESSNet hub. If you use your own workstation, IBM recommends that it have at least 128 MB of memory.

The ESS Copy Services Web User Interface requires one of the following Internet browsers:

- Netscape Navigator
- Microsoft Internet Explorer (MSIE)

For supported versions of Netscape Navigator and Internet Explorer, refer to the following publication: *IBM TotalStorage Enterprise Storage Server Web Interface User's Guide,* SC26-7448.

### 7.1.3 Starting the Web browser

As already mentioned, to manage the ESS Copy Services you can either use the ESS Master Console, or a Web browser installed on a workstation connected through the ESSNet.

There is a difference when accessing the ESS, whether your are using the ESS Master Console, or some other workstation. The ESS Master Console has a Netscape icon with the label ESS Specialist Launcher. This icon sends you to a panel that provides a list of IP addresses of the ESS clusters. Click the address of the cluster that is an active Copy Services server to start using the ESS Copy Services Web User Interface.

If you use a workstation other than the ESS Master Console, you must enter the desired cluster IP address in the address field of your Web browser window. The address that you click in the ESS Specialist Launcher of your ESS Master Console, or that you type in the Address field of your Web browser window, is the host name alias or the dotted decimal IP address of one of the ESS clusters. IBM configures those addresses in your ESS at installation time, based on the addresses you enter in the Communications Resources worksheet. The publication *IBM TotalStorage Enterprise Storage Server Introduction and Planning Guide,* GC26-7444, gives detailed information on this configuration procedure.

Figure 7-1 shows the ESS Welcome panel. On the left navigation bar, you can see the **Copy Services** button that is used to start the interface with the ESS Copy Services functions.

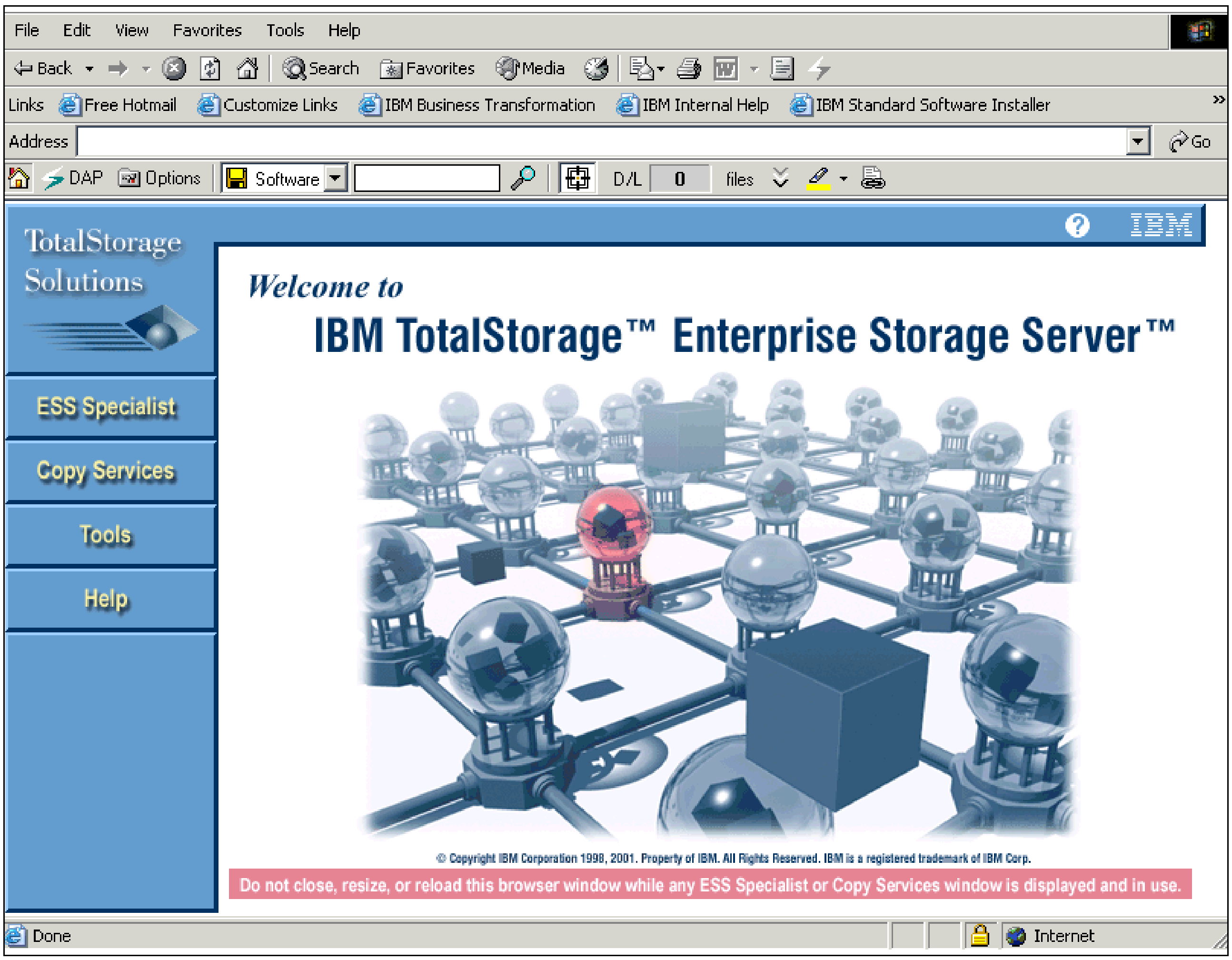



*Figure 7-1 ESS Welcome panel*

## 7.1.4 ESS Copy Services Command Line Interface (CLI)

With the ESS Copy Services server you have the possibility to save the setup of any data copy action as a *task*. These tasks can then be invoked from any of the supported open system servers through a command (rsExecuteTask).

For detailed information on the ESS Copy Services Command Line Interface (CLI), and how to use it, you can refer to the publication: *IBM TotalStorage Enterprise Storage Server Copy Services Command-Line Interface Reference,* SC26-7449.

## 7.2 Setting up the ESS Copy Services Domain

The ESS Copy Services Domain can be configured and maintained by selecting the **Tools** button from the ESS Welcome Panel (see Figure 7-1). After clicking the **Tools** button, the Copy Services Main Service Page will be displayed. Figure 7-2 shows the selections that can be made.


*Figure 7-2 Copy Services Main Service Page*

From this page, you can choose an action that will display a page with the Active Server Configuration and the Available Actions as shown in Figure 7-3 and Figure 7-4.

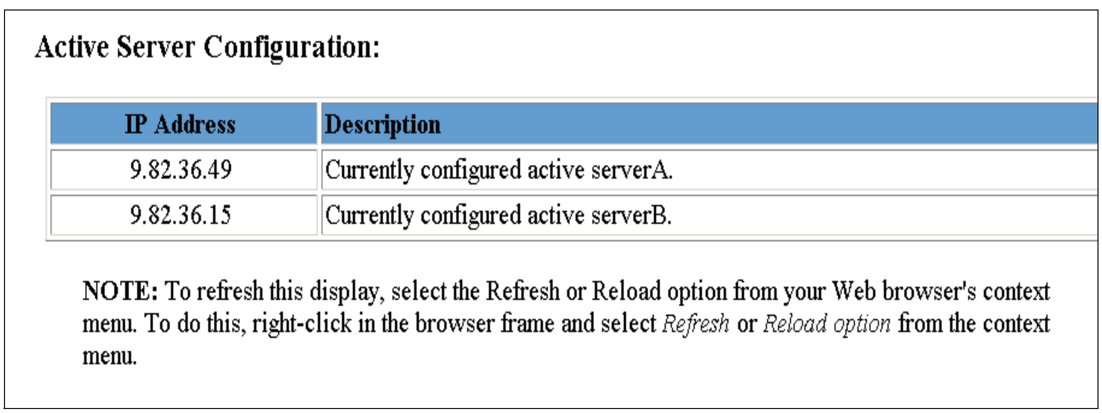



*Figure 7-3 Active Server Configuration*

**Available Actions:**

**NOTE:** To ensure proper processing of these actions, your browser should be set to avoid caching these pages:
For **Netscape**, select Edit, Preferences, Advanced, Cache, and select *Document in cache is compared to document on network **every time***.
For **Internet Explorer**, select Tools, Internet Options, General, Temporary Internet File Settings, and *Check for newer versions of stored pages **on every visit to the page***.

| Action | Description |
|---|---|
| Restart | Restart Copy Services on this cluster. |
| Disable | Disable Copy Services on this cluster. |
| Domain-wide Reset | Perform a Domain-wide Reset. |
| Define Servers | Define Copy Services active servers. |
| Define Clients | Define Copy Services clients. |
| Cancel | Return to the Tools main page, without performing any action. |

*Figure 7-4 Copy Services Available Actions*

## 7.2.1 Restart

This function stops and restarts Copy Services in this cluster only. It does not affect the other clusters within the Copy Services Domain.

## 7.2.2 Disable

This function stops Copy Services in this cluster only. It does not affect the other clusters within the Copy Services Domain.

## 7.2.3 Domain-wide Reset

Domain-wide Reset stops and restarts Copy Services on each cluster within the Copy Services Domain. Domain-wide Reset needs to have a common Web Administrator level user ID and password defined on all ESS clients in the ESS Copy Services Domain.

Copy Services clients need to be configured on ESS clusters that are defined as ServerA and ServerB to support Domain-wide Reset. Domain-wide Reset must be activated from the cluster where the Copy Services primary server is running. The Domain-wide Reset function reports success or failure results of resetting Copy Services in each cluster in the domain.

**Attention:** We recommend that you do not perform a Domain-wide Reset, unless you are certain that there are no other recovery options.

When you perform the Domain-wide Reset function, please be aware that:

- You will lose any PPRC or FlashCopy tasks for which you have not received a successful completion message.
- Established PPRC and FlashCopy relationships are maintained.
- You cannot submit any additional command line interface (CLI) tasks until ESS Copy Services has initialized.

- This function will not be successful on the cluster, which does not have the Domain-wide Reset code installed.
- For the clusters that do not have the Domain-wide Reset code, reset has to be performed on the local cluster separately.

## 7.2.4 Define servers

The active Copy Services server is the ESS cluster that is currently running the Copy Services server code and communicating to the Web browsers. Two Copy Services servers (ServerA and ServerB) can be active simultaneously in a Copy Services Domain.

ServerA must always be defined, in other words, you cannot define ServerB without defining ServerA. Dual active Copy Services servers removes the single point of failure. Figure 7-5 shows IP address input boxes to define active Copy Services ServerA and ServerB.

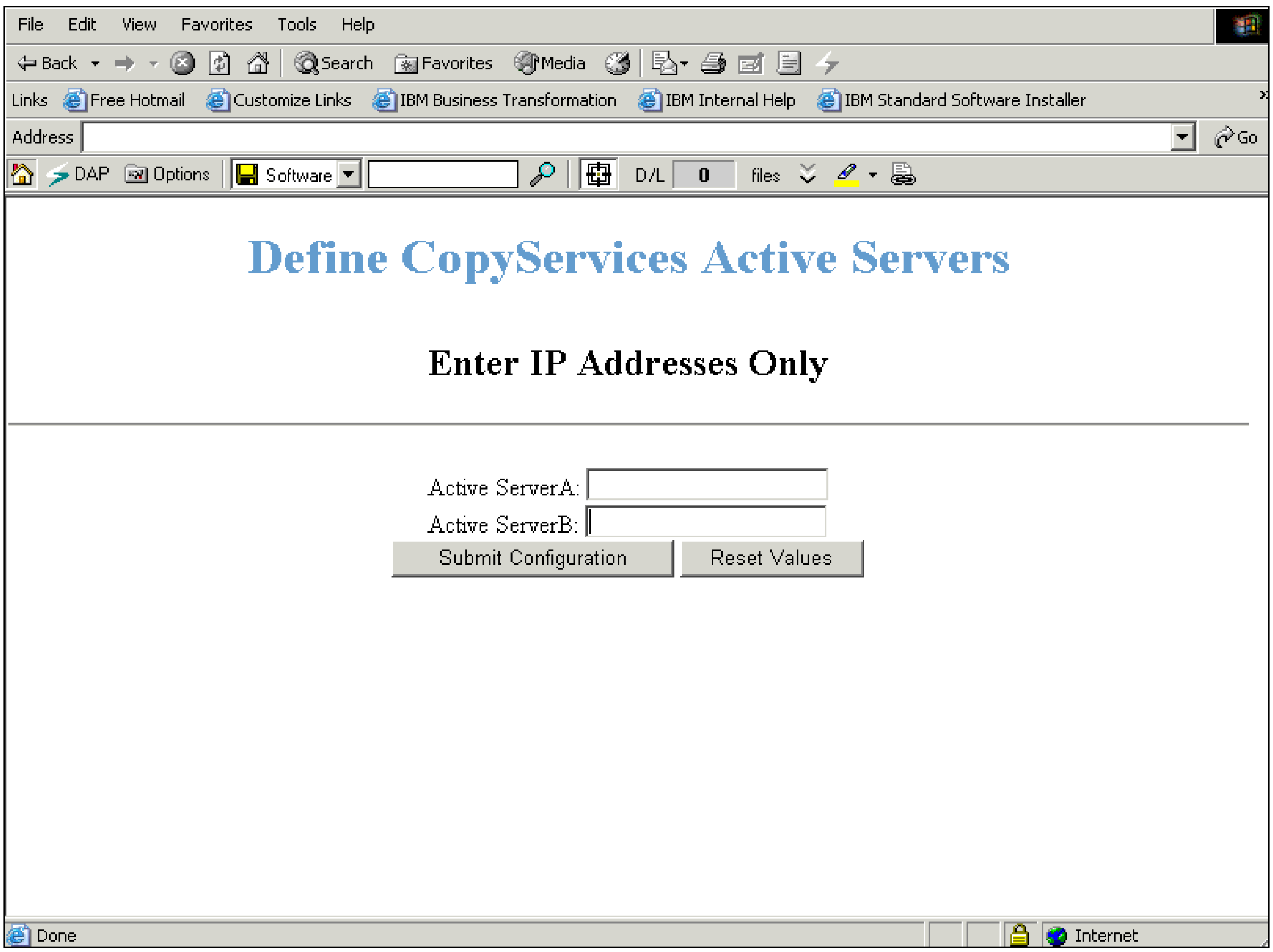



*Figure 7-5 Defining Copy Services active servers*

### Dual active Copy Services servers

At initialization, Copy Services servers register with each other. The task repository in ServerA will overwrite the task repository in ServerB for the tasks that are supported by both servers.

In a dual active Copy Services server configuration, the following operations are performed:

- Configuration and state changes will be reflected in both servers.
- Each server can create and modify tasks; tasks will be reflected in the other server provided the other server supports the options in the task.
- Each server can execute tasks; task reports will only been sent to the server that initiated the task.
- ServerA will overwrite tasks supported by both servers on ServerB.
- Tasks not supported by both servers will be appended with `~xx` (where `xx` is 00-99).

Also, when both servers are active, two agents per cluster will be executing, one connecting to each Copy Services server. Reporting is logged in *rsCSclientA.log* and *rsCSclientB.log* respectively.

Modifying a task or group task in a dual Copy Services server configuration will be honored as follows:

- Tasks that are supported by both servers will be saved and maintained in both servers.
- Tasks that are supported by only the initiating server will be saved with names appended `~xx` in the initiating server only.
- If one or more tasks in a group task are not supported by the other server or their name has `~xx` appended, all tasks in that group that are supported by both servers will be cloned. All the tasks within that group will have names with `~xx` appended. The group task name also has `~xx` appended.

After a code upgrade on one Copy Services server, tasks with `~xx` appended must be manually renamed to remove the append and cause it to be replicated to the other server.

Table 7-1 shows supported Copy Services servers configurations. The table is read as follows. Start with the row and compare to the column for that row:

- ServerA cannot exist with another ServerA in the same configuration (row 2 - column 2).
- ServerA and ServerB configuration is valid (row 2 - column 3).
- ServerA cannot exist with Primary Server in the same configuration (row 2 - column 4).
- ServerA can exist with Backup Server (row 2 - column 5).
- ServerA can exist with no alternate server defined (row 2 - column 6).

The configuration layout for ServerB is as follows:

- ServerB can exist with ServerA in the same configuration (row 3 - column 2).
- ServerB cannot exist with another ServerB in the same configuration (row 3 - column 3).
- ServerB and Primary Server is allowed but not recommended (row 3 - column 4).
- ServerB and Backup Server configuration is not valid (row 3 - column 5)
- ServerB cannot exist without an alternate server (row 3 - column 6).

*Table 7-1 Copy Services server configurations*

| | **ServerA** | **ServerB** | **Primary** | **Backup** | **None** |
|---|---|---|---|---|---|
| **ServerA** | No | Yes | No | Yes | Yes |
| **ServerB** | Yes | No | Yes (not recommended) | No | No |

When configuring invalid server configuration, servers will be allowed to start up, but the servers will not communicate with each other. They will run as two separate servers, therefore, no synchronization of tasks will take place.

> **Tip:** We recommend that you configure ServerA at the recovery site and ServerB at the production site.

### 7.2.5 Define clients

A Copy Services client is software that runs on each cluster in the Copy Services server group and performs the following functions:

- It communicates configuration, status, and connectivity information to the Copy Services server.
- It executes data-copy tasks on behalf of the Copy Services server.

Copy Services clients can be defined on clusters that have no Copy Services server defined. The Copy Services client is defined for one cluster and configuration changes are not sent to the alternate cluster. Figure 7-6 shows the define Copy Services clients Web panel.

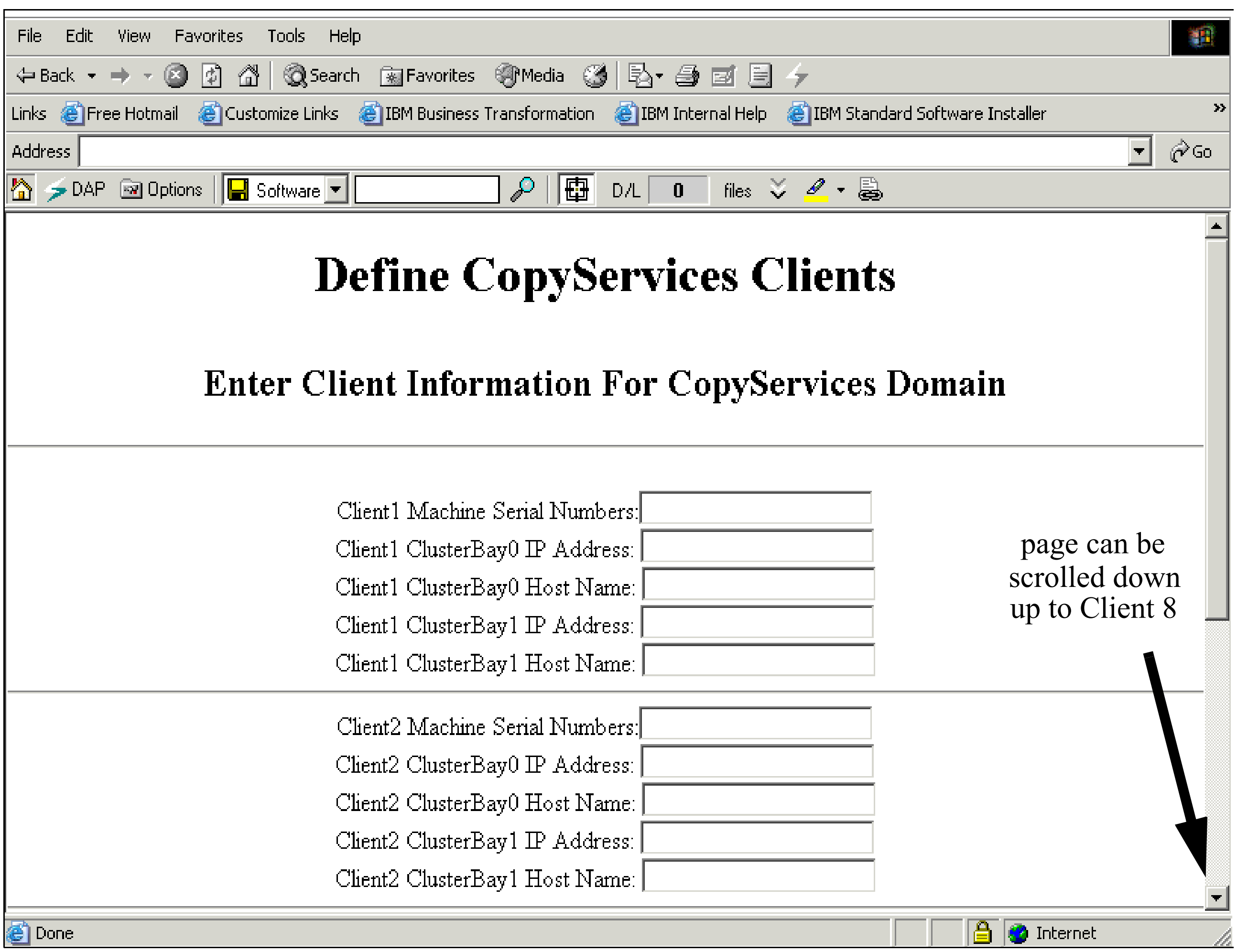



*Figure 7-6 Defining Copy Service clients*

When defining a client, you specify the serial number of the ESS, IP addresses for Bay0, and Bay1 of a cluster, which can be cluster 0 or cluster 1, or both. Specify a host name to be associated with cluster bays in the TCP/IP network.

## 7.3 ESS Copy Services Web User Interface

The first panel to open, the Welcome to IBM TotalStorage Enterprise Storage Server panel, has a **Copy Services** button that you click to access ESS Copy Services (refer back to Figure 7-1 on page 302).

After clicking the **Copy Services** button, the start copy services menu is presented (see Figure 7-7). This panel shows two areas: Server Configuration and Available Actions.

### Server configuration

Currently configured active Copy Services servers are displayed here. In Figure 7-7 both Copy Services servers are configured.

### Available actions

This connects the browser to the ESS cluster defined as the Copy Services server. In Figure 7-7 you can either choose to connect to ServerA or ServerB by clicking the **Start ServerA** or **Start ServerB** button.

**Note:** It is assumed that most zSeries users will prefer to use the tools discussed in Chapter 4, “Managing Copy Services” on page 145 so only a brief overview of the Web User Interface tasks are given here. For in-depth detailed descriptions of using the Web User Interface to configure and manage Copy Services on ESS, refer to the companion Redbook, *IBM TotalStorage Enterprise Storage Server Implementing ESS Copy Services in Open Environments,* SG24-5757.

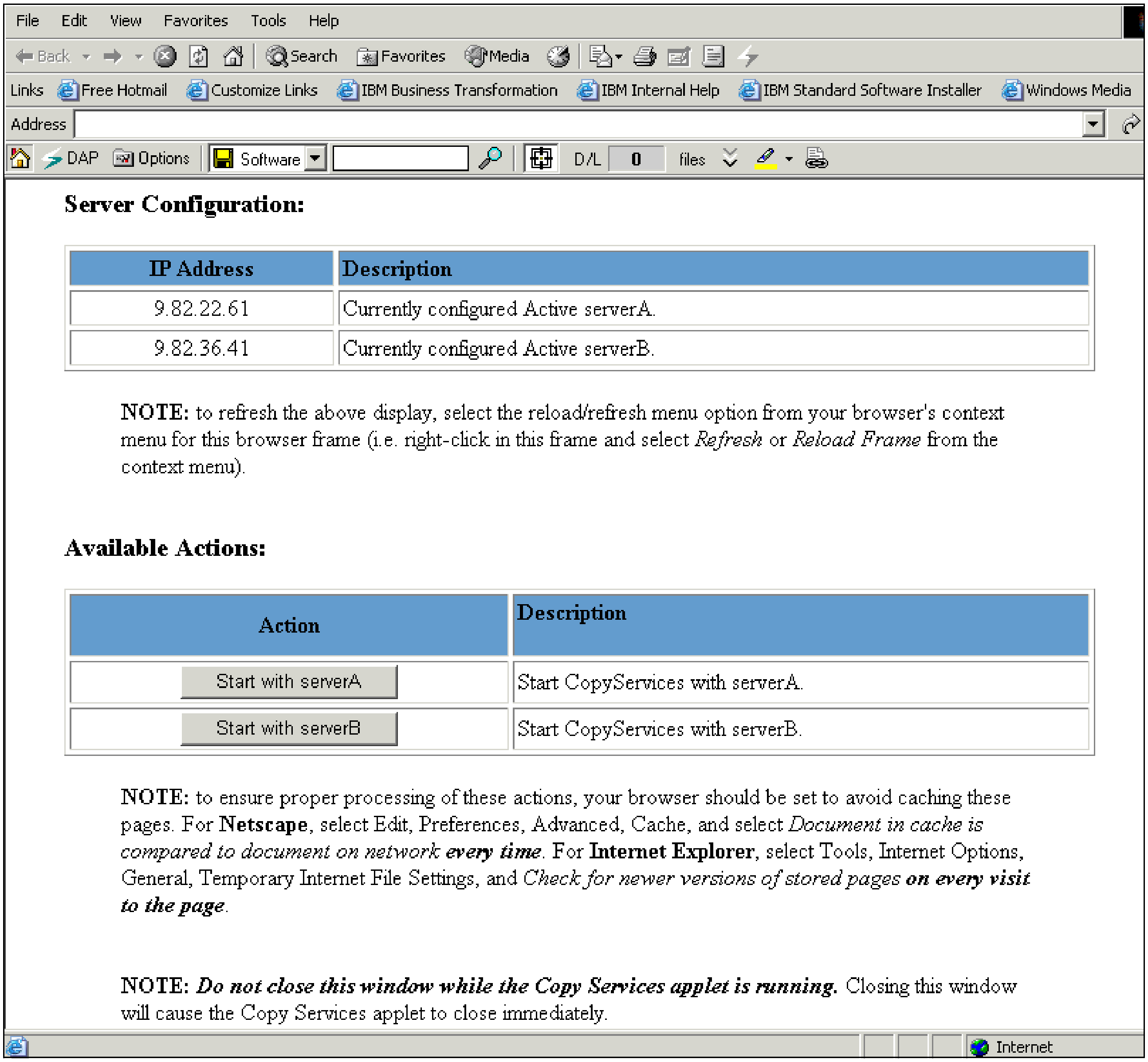



*Figure 7-7 ESS start Copy Services*

> **Important:** *Do not close this window while the Copy Services applet is running*. Closing this window will cause the Copy Services applet to close immediately. The window can be minimized for convenience.

Once you click a server, the message window shown in Figure 7-8 will be displayed while connecting to the selected Copy Services server.


*Figure 7-8 Connecting to Copy Services Server message*

**Note:** Depending on the network configuration, the process of downloading the Java applets to your workstation and presenting the Copy Services panels can take several minutes. Be patient.

When the connection to the Copy Services server is completed, the main menu of the ESS Copy Services WUI will be displayed (see Figure 7-9). From here you can access the ESS Copy Services menus by clicking the buttons on the left side.


*Figure 7-9 ESS Copy Services Welcome panel*

## 7.4 The Volumes panel

With the Volumes panel you select individual volumes from the source and target LSSs, and then you establish a relationship between them. The following can be done:

- Establish and withdraw FlashCopy pairs
- Establish, suspend, re-synchronize, and terminate Synchronous PPRC copy pairs
- Failover/Failback PPRC pairs
- Establish, suspend, and terminate PPRC Extended Distance copy pair
- Convert a PPRC Extended Distance copy pair to Synchronous PPRC
- Establish multiple volume-pair relationships for FlashCopy and PPRC
- Find a volume
- Display volumes based on a filter
- View information about a volume

Figure 7-10 shows the Volumes panel.

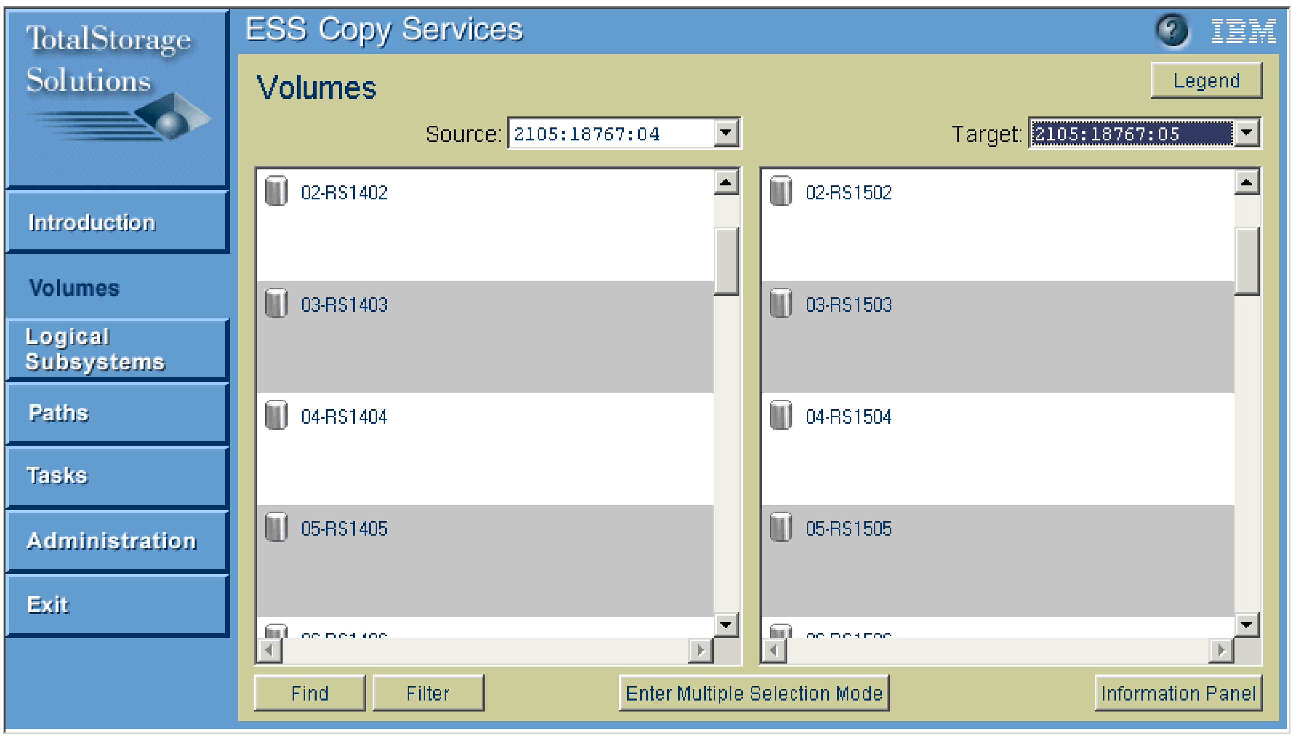


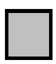

*Figure 7-10 Volumes panel*

The source and target *logical subsystems* (LSSs) are displayed in the following way: *device type* (4 digits): ESS *serial number* (5 digits): LSS *number* (2 digits). Examples are logical subsystem addressed as `2105:18767:04` or `2105:18767:05`.

## 7.4.1 Icons visible in the Volumes panel

The Volumes panel shows all volumes in the LSS. Next to each volume you find its unique number. Status icons are shown next to the volume and indicates if the volume is or is not in a copy relationship and its status in the relationship. Clicking the **Legend** button in the top right hand corner will show all the possible volume icons and their descriptions as shown in Table 7-2.

*Table 7-2 Meaning of the icons and colors in the Volume panel*

| Icon | Description |
|---|---|
|  | Volume icon. This icon designates a volume created in the ESS. This icon is always displayed whether this volume is or is not in a copy relationship. |
|  | Designated source volume. When you right-click a volume to select it as a source, then the volume icon and its serial number are surrounded by a rectangle filled with blue color. |
|  | Designated target volume. When you click a volume to select it as a target, the volume icon and its serial number are surrounded by a rectangle filled with red color. |
|  | Designated source and target volume. When you click, followed by a right-click, on the same volume, then the volume icon and its serial number are surrounded by a rectangle filled with grey color. |
|  | PPRC source volume. The volume is a source volume in an established PPRC relationship. And the volume is in duplex mode for this relationship. |
|  | PPRC target volume. The volume is a target volume in an established PPRC relationship. The volume is in duplex mode for this relationship. |
|  | PPRC source volume copy in progress. The volume is a source volume in an established PPRC relationship. The volume is in duplex pending mode for this relationship. |
|  | PPRC target volume copy in progress. The volume is a target volume in an established PPRC relationship. The volume is in duplex pending mode for this relationship. |
|  | PPRC source suspended. The volume is a source volume in an established PPRC relationship. The volume is in suspended mode for this relationship. |
|  | PPRC target suspended. The volume is a target volume in an established PPRC relationship. The volume is in suspended mode for this relationship. |
|  | PPRC Extended Distance source. The volume is a source volume in an established PPRC-XD relationship. |

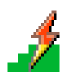
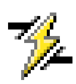
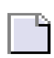
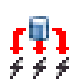
| Icon | Description |
|---|---|
|  | PPRC Extended Distance target. The volume is a target volume in an established PPRC-XD relationship. |
|  | PPRC Trusted Primed for Resynchronization |
|  | FlashCopy source. The volume is a source volume in an established FlashCopy relationship. |
|  | FlashCopy target. The volume is a target volume in an established FlashCopy relationship. |
|  | Change recording enabled. This icon displays on the source and target volume when you establish a FlashCopy with the Change recording enabled option. The pair can be used to perform Incremental FlashCopy or Reverse Restore FlashCopy. |
|  | Dataset FlashCopy (for S/390 only). |
|  | Volume Copy. The volume is the target of a FlashCopy relationship. This icon remains even if the FlashCopy relationship is withdrawn. To remove this icon manually, you need to withdraw the relationship with a *withdraw from target* command. |
|  | Volume Session Active. The volume is active in a session. |
|  | Volume Session Join Pending. The volume is joining the session, but is not yet active. |
|  | Volume Session Remove Pending. This volume is in the process of being removed from the session, when complete, this icon will be removed. |
|  | Multiple FlashCopy relationship. The volume is a source volume for more than one FlashCopy relationship. |
|  | The state of this volume cannot be determined. |

*Figure 7-11 Volume icons and description*

## 7.4.2 Information panel

You can get more detailed information about a single volume by selecting the volume and then clicking the **Information Panel** button. The Information Panel window will pop-up, displaying detailed information about the selected volume (see Figure 7-12).

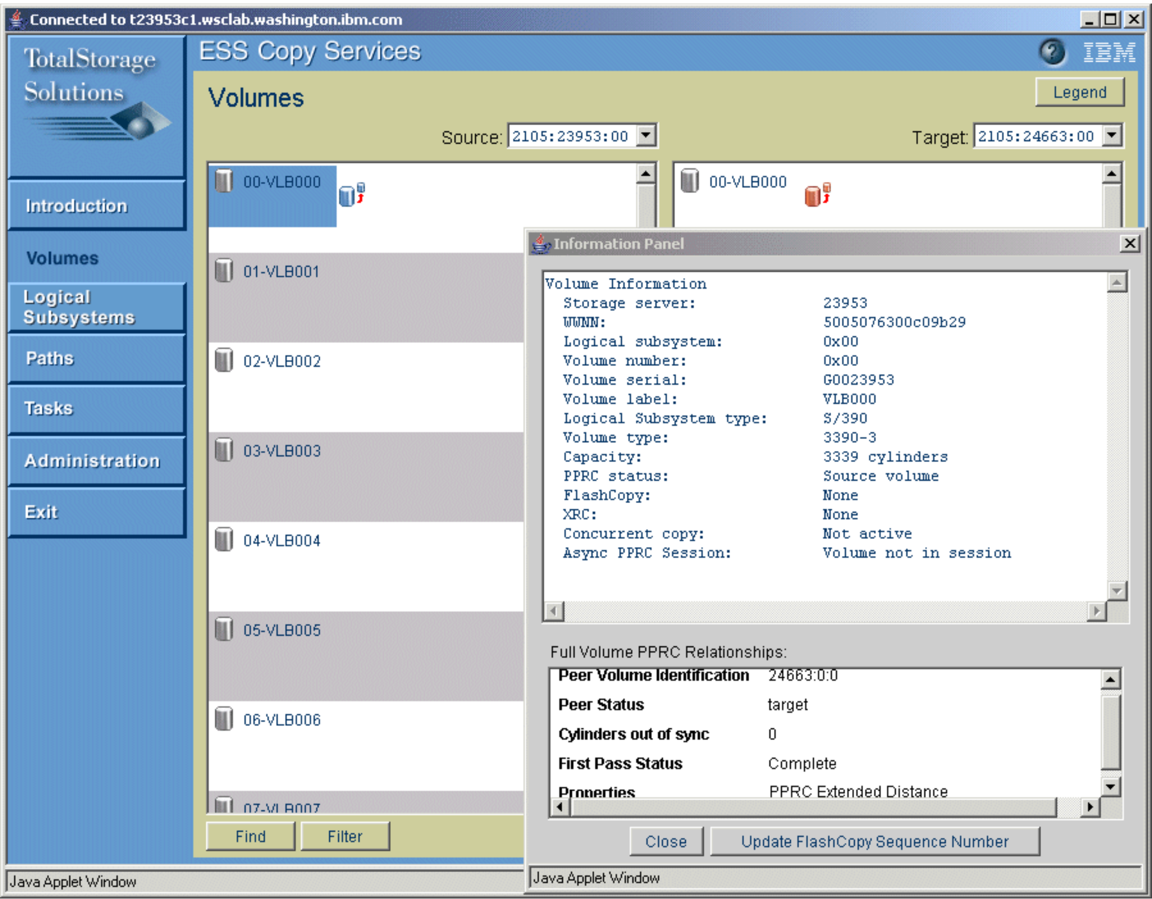



*Figure 7-12 Volume information window*

The following information is displayed for a volume:

- The ESS and logical subsystem (LSS) designations
- The volume number
- The volume serial number
- The LSS type
- The volume type (for example, 3390,9337, or the like)
- The volume capacity
- PPRC status (whether the volume is primary or secondary, and the state of the volume)
- Any active FlashCopy, Concurrent Copy, XRC operations, or Asynchronous PPRC
- Information about a companion volume if a copy pair exists

Additional information about the buttons available in the Volumes panel can be found in the IBM publication: *IBM TotalStorage Enterprise Storage Server Web Interface User's Guide,* SC26-7448.

## 7.5 The Logical Subsystems panel

The Logical Subsystems panel displays the logical subsystems (LSS) of the connected ESSs in the Copy Services Domain. Each of the logical subsystems is indicated by the serial number of the ESS it belongs to and its 2-digit LSS number within the ESS.

From the Logical Subsystems panel, you are able to:

- Suspend or terminate PPRC copy pairs for the LSS
- Convert PPRC-XD pairs to synchronous pairs for all the volumes in the LSS
- Filter a list of LSSs
- Find an LSS
- Freeze a PPRC Consistency Group
- Modify the PPRC Consistency Group time-out value
- Remove orphaned paths
- Re-synchronize PPRC copy pairs for the LSS
- Run a Consistency-Group-Created operation
- Withdraw FlashCopy pairs for the LSS
- View information about an LSS

Figure 7-13 shows the Logical Subsystems panel. The color indicates the state of the LSS, whether it contains volumes that are currently in a copy relationship (primary, secondary, or mixed), or not.

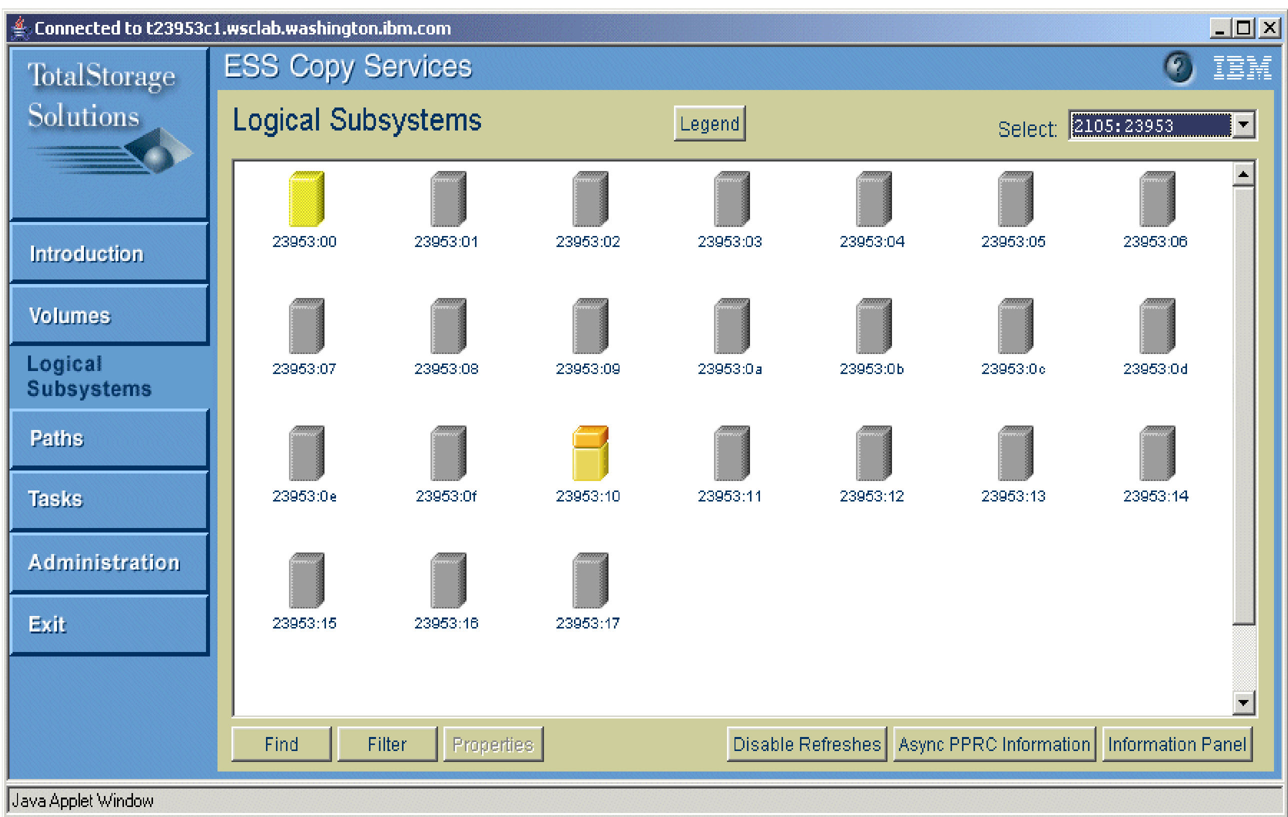
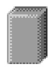
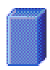
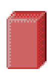
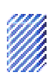
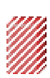


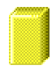

*Figure 7-13 Logical subsystem panel*

## 7.5.1 Logical Subsystems panel icons

Table 7-3 explains the icons that can appear in the Logical Subsystems panel.

*Table 7-3 Logical Subsystems panel Icons*

| LSS Icon | Appearance | Meaning |
|---|---|---|
|  | Gray solid ("Simplex state") | All the volumes in the LSS are in simplex state, meaning that they are not in a copy relationship. |
|  | As above with yellow top indicates that this LSS is also a Master | |
|  | Blue solid (PPRC source) | All the volumes in the LSS are in PPRC pairs (Synchronous or Extended Distance) in a non-suspended state. |
|  | As above with yellow top indicates that this LSS is also a Master | |
|  | Red solid (PPRC target) | |
|  | As above with yellow top indicates that this LSS is also a Master | |
|  | Blue and white stripes (PPRC suspended source) | All the volumes in the LSS are in PPRC pairs (Synchronous or Extended Distance) in a suspended state. |
|  | As above with yellow top indicates that this LSS is also a Master | |
|  | Red and white stripes (PPRC suspended target) | |
|  | As above with yellow top indicates that this LSS is also a Master | |

| LSS Icon | Appearance | Meaning |
|---|---|---|
|  | Yellow solid | Not all volumes in the LSS are in the same type of Copy Services relationship (this applies for PPRC and FlashCopy relationships), and none of the volumes are in a suspended state. |
|  | As above with yellow top indicates that this LSS is also a Master | |
|  | Yellow and white stripes (mixed types and states) | The volumes in the LSS are either the same type, but some are in a non-suspended state, or not all of the volumes are the same type, but some are in a suspended state. |
|  | As above with yellow top indicates that this LSS is also a Master | |

## 7.5.2 Information panel

You can get more detailed information about a single logical subsystem by selecting the **LSS** from the Logical Subsystems panel, and then clicking the **Information Panel** button. Figure 7-14 illustrates the information that is displayed for a selected LSS.

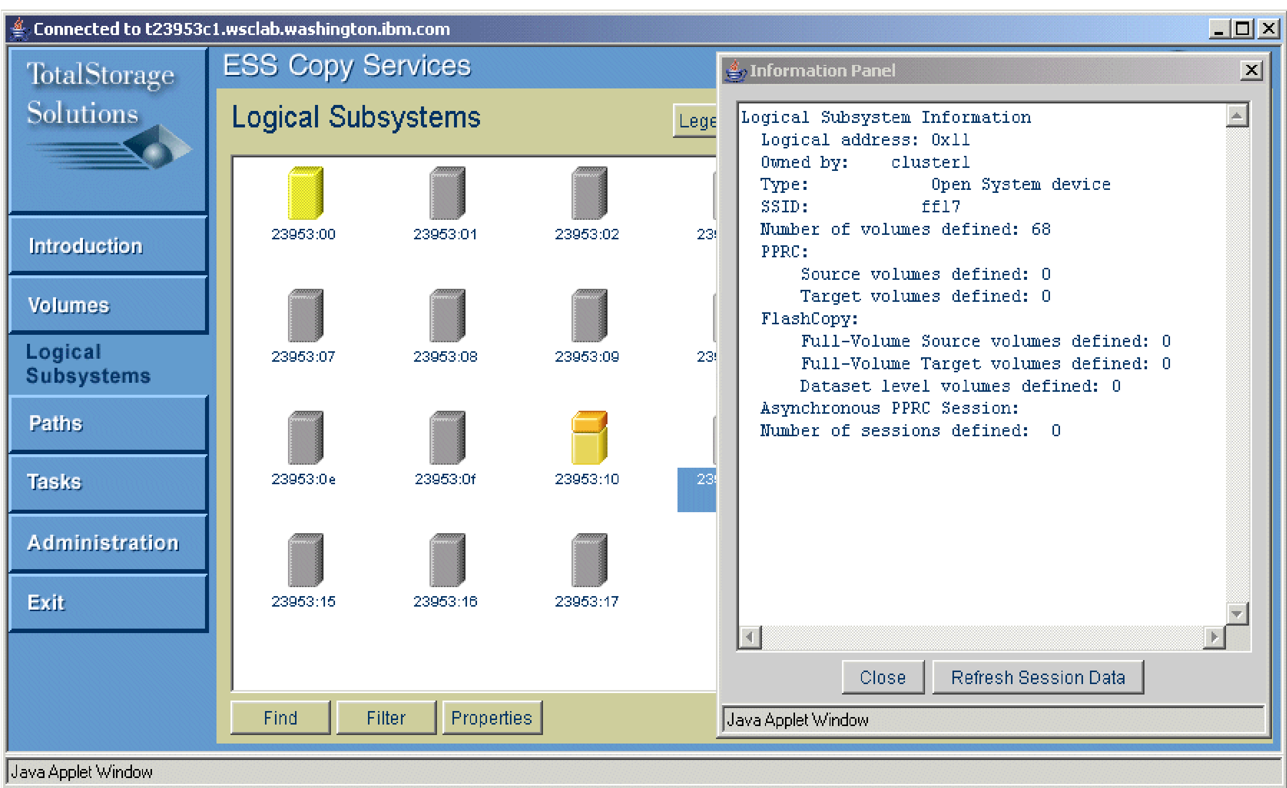



*Figure 7-14 LSS information panel*

From the paths panel, the Asynchronous PPRC status can be display by clicking the **Async PPRC Information** button at the bottom of the screen as shown in Figure 7-15.

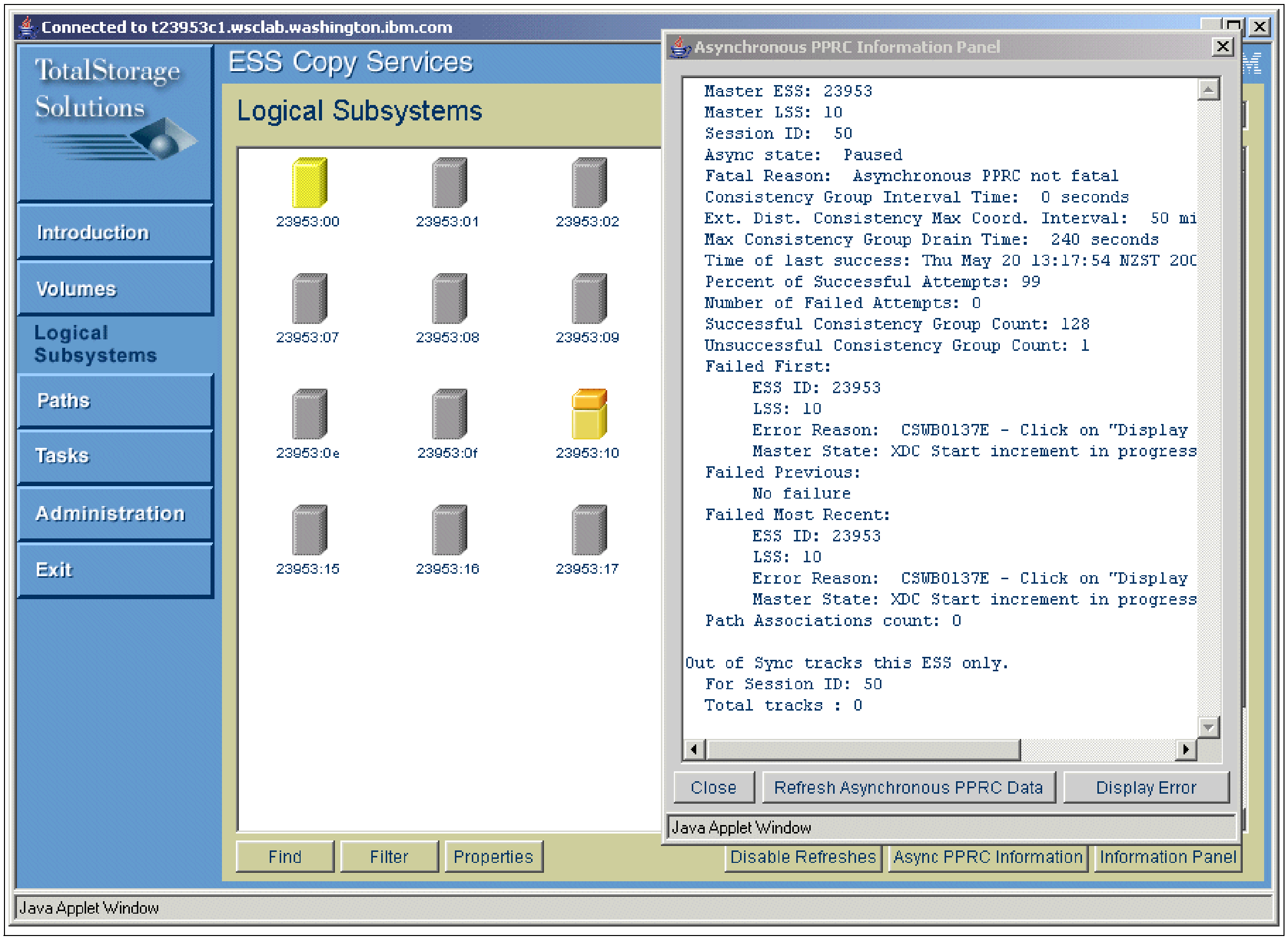



*Figure 7-15 Asynchronous PPRC information panel*

## 7.6 The Paths panel

A *path* logically connects the primary and secondary LSSs. The physical *link* is the ESCON or FCP connection between the two ESSs. One physical PPRC link can be used for more than one logical path definition; for ESCON, all must have the same direction.

Before you can create PPRC volume pair relationships, you must define the logical paths between the primary and secondary LSSs. After you establish the paths, the Paths panel displays the current status of the paths for the selected LSSs.

From the Paths panel you can choose one of the **View Path Status** or **Work with Paths** radio buttons. You can then select the LSS and type of link you want to work with, ESCON or Fibre Channel, to:

- View information about paths
- Establish paths
- Add paths
- Remove a group of established paths
- Remove one or more paths from a group of established paths
- Enable PPRC Consistency Group
- Configure Asynchronous PPRC sessions
- Manage Asynchronous PPRC

Figure 7-16 shows the Paths panel with the **View Path Status** radio button and ESCON links selected. This displays the ESCON paths for the selected LSS.

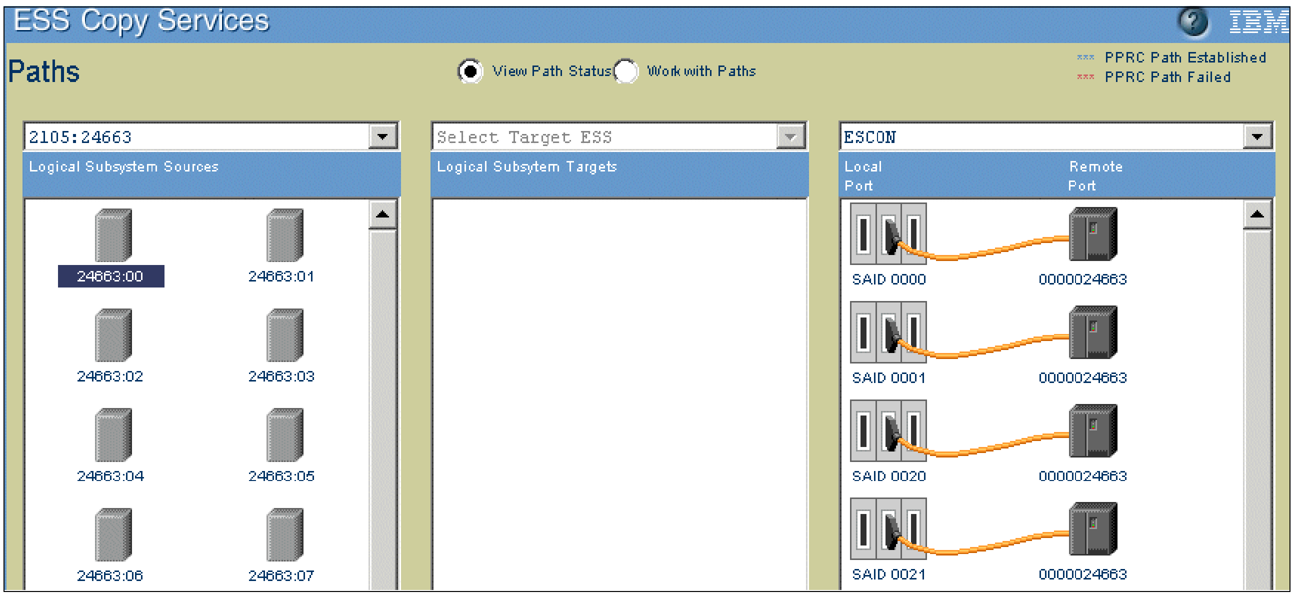



*Figure 7-16 View Path Status: ESCON links*

Figure 7-17 shows the Paths panel with the **Work with Paths** radio button and FCP paths selected. When working with paths, both source and target LSSs must be selected.

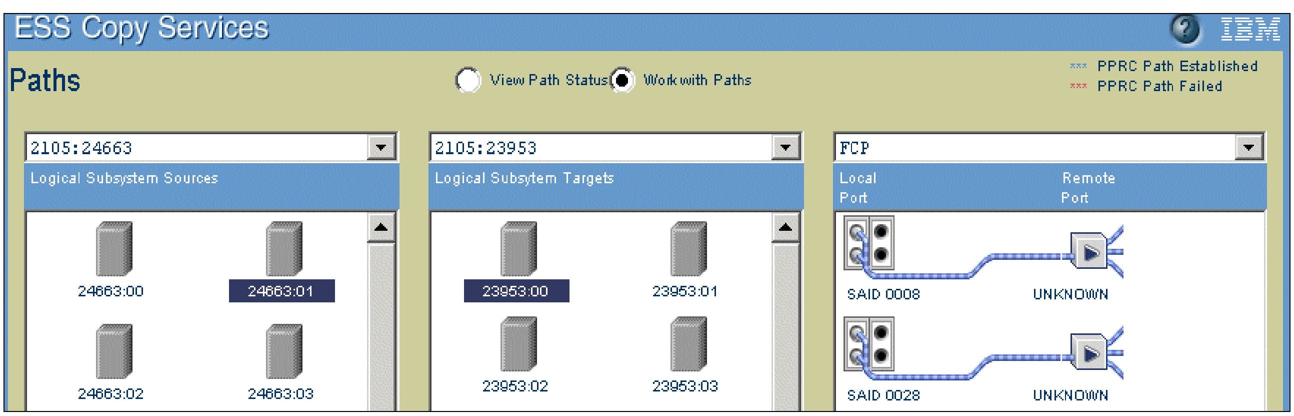
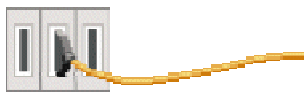
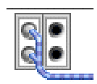
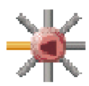
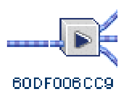
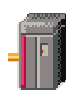
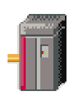



*Figure 7-17 Work with Paths panels: FCP links*

## 7.6.1 Paths panel icons and symbols

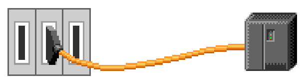
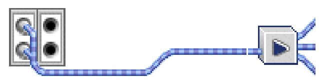
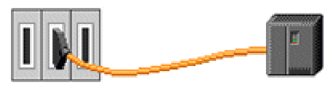
Table 7-4 shows the meaning of the symbols displayed in the Path Origin window area.

*Table 7-4 Path Origin window symbol*

| Connection icon | Description |
|---|---|
|  | ESCON host adapter port. See Appendix E, “System Adapter ID (SAID)” on page 391. |
|  | FCP host adapter port |
|  | ESCON director |
|  | FCP Switch |
|  | Host server |
|  | ESS storage server |

If there are *logical paths* defined on an ESCON or FCP adapter, you will find three blue asterisks directly below the path-connection symbol in the Path Origin window area. Three red asterisks below the path-connection symbol in the Path Origin area mean that the last attempt to establish the path failed. No asterisks mean that there are no logical paths defined on this ESCON or FCP connection. Table 7-5 shows these path status information symbols.

*Table 7-5 Paths status*

| ESCON without defined path | FCP with defined path | ESCON failed to establish path |
|---|---|---|
|   |   |   |

## 7.6.2 Display Direct Connect Paths button

If you have two ESSs that are directly connected (that is, no switch between them), you can click the **Display Direct Connect Paths** button to display those paths.

## 7.6.3 Information panel

Once a SAID adapter is selected, you can get more information about the paths by clicking the **Information Panel** button at the bottom of the Paths panel screen.

Figure 7-18 shows a path from primary LSS 00 in ESS serial number 23953, using FCP port SAID 000c.

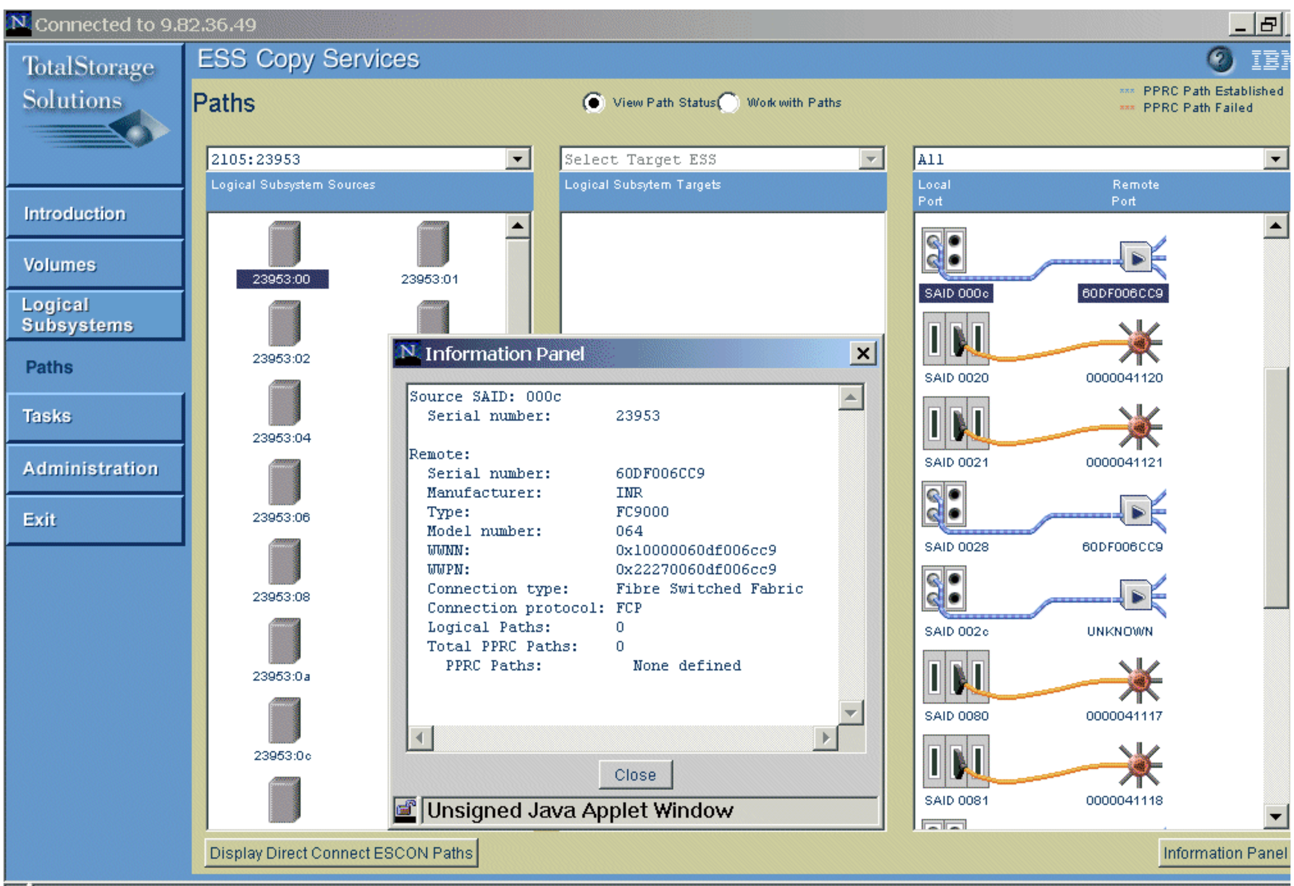


*Figure 7-18 Paths information panel*

# 7.7 The Tasks panel

When using the ESS Copy Services WUI, you have the possibility to save any data copy request as a *task*. This could be any kind of FlashCopy or PPRC request.

In addition, multiple tasks can be grouped together into a single *task group*. This could be the case if multiple FlashCopy pairs from different LSSs have to be established at the same time in order to do a backup. All tasks within a task group will be processed in parallel.

From the Tasks panel you will be able to:

- Group tasks
- Modify a task
- Remove a task group
- Remove a saved task
- Run a saved task
- Ungroup tasks
- View error information about a failed task
- View information about a saved task

Figure 7-19 shows the Tasks panel presented by the ESS Copy Services Web User Interface. For each previously saved task, the name, a description, and the status of the last execution is displayed.

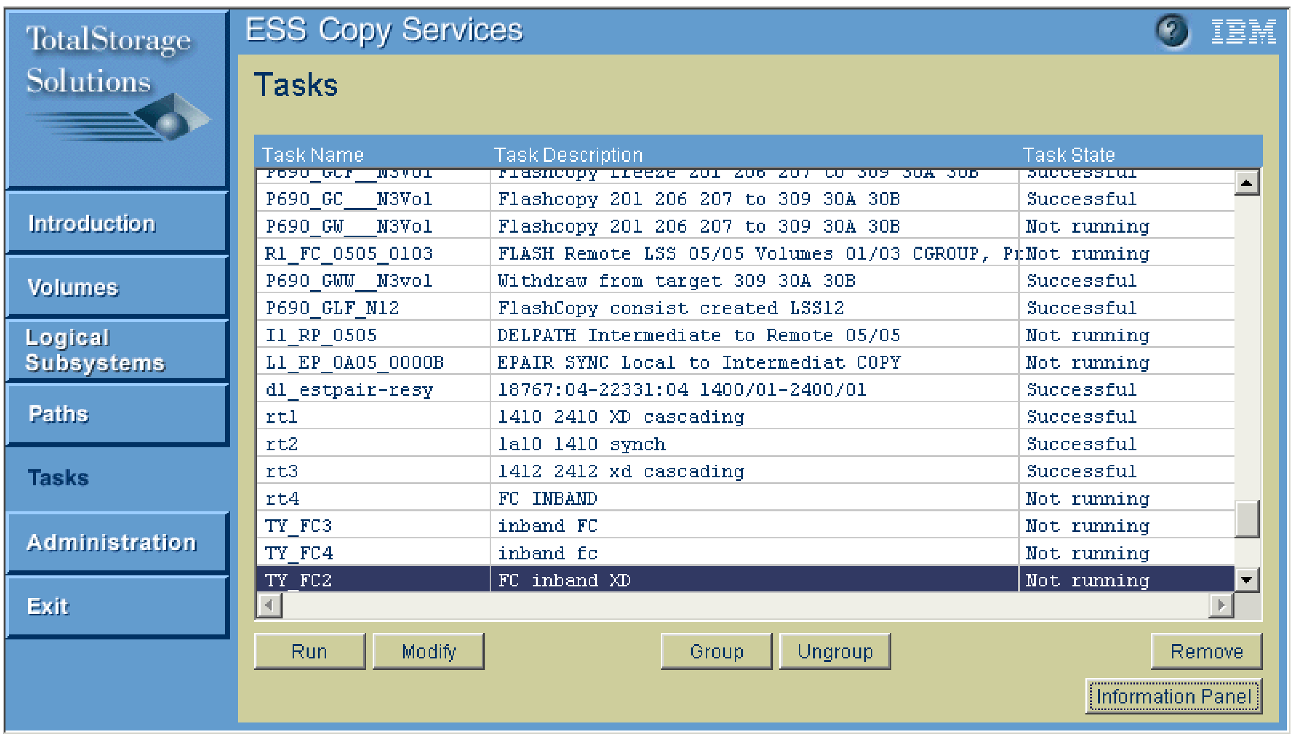



*Figure 7-19 Tasks panel*

## 7.7.1 Grouping and un-grouping tasks

To group tasks (refer to Figure 7-19), you click the desired tasks while holding either the Shift key (to select a range, from the listed tasks) or the Ctrl key (to pick up single tasks from the listed tasks). Once you are finished, click the **Group** button and specify the group name. It is not possible to include a group into another task group.

To ungroup a task (refer to Figure 7-19), you select the task group, and click the **Ungroup** button at the bottom of the Tasks panel.

## 7.7.2 Information panel

You can also get detailed information about a task by selecting the task, and then clicking the **Information Panel** button. In our example in Figure 7-20, the task is named `TY_FC2`. This task establishes a FlashCopy pair with the inband option.

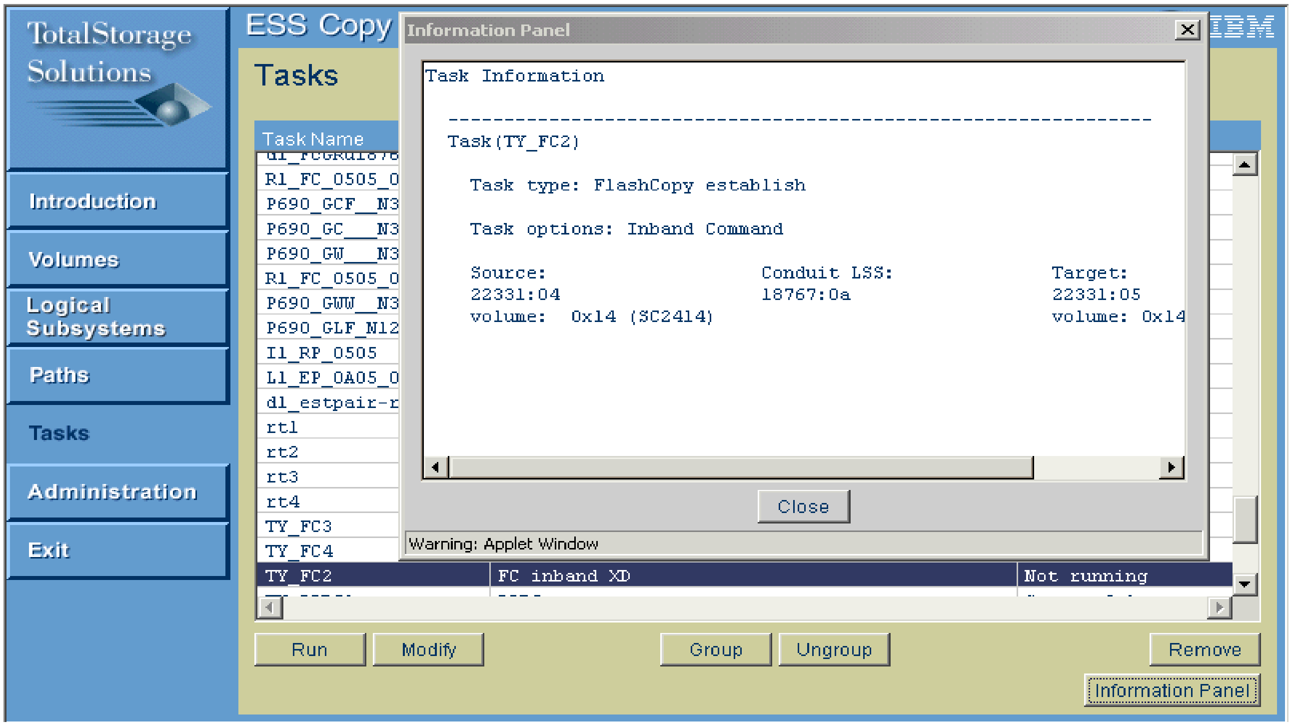



*Figure 7-20 Task information panel*

## 7.7.3 Removing a task

To remove a task (Figure 7-19 on page 322), select the task, and click the **Remove** button.

## 7.7.4 Running a task

To run a saved task (Figure 7-19 on page 322), select the task, and click the **Run** button. The task will be executed immediately.

If the task fails, a failure report is available through the Information panel. The failure report contains the error messages and the sense data. The messages are documented in the publication: *IBM TotalStorage Enterprise Storage Server Web Interface User's Guide,* SC26-7448.

**Tip:** We recommend that you save all tasks, because only the saved tasks show you the ESS Copy Services messages from the Information panel. Otherwise, you have to go to the Administration panel log/report for the related *undefined* task.

## 7.7.5 Modifying a task

Use the Tasks panel (Figure 7-19 on page 322) to modify a previously saved task. You can change a task from one that establishes a PPRC copy pair, to one that suspends, re-synchronizes, or terminates a copy pair. For FlashCopy, you can change a task that establishes a FlashCopy pair to one that withdraws a FlashCopy pair.

**Note:** Grouped tasks cannot be modified. The tasks must be un-grouped and then tasks can be modified.

## 7.8 The Administration panel

The Administration panel (shown in Figure 7-21) is used to manage the server logs, reports, and ESS Copy Services CLI user IDs, and passwords. You can also use this panel to refresh the volume and LSS information for an ESS.

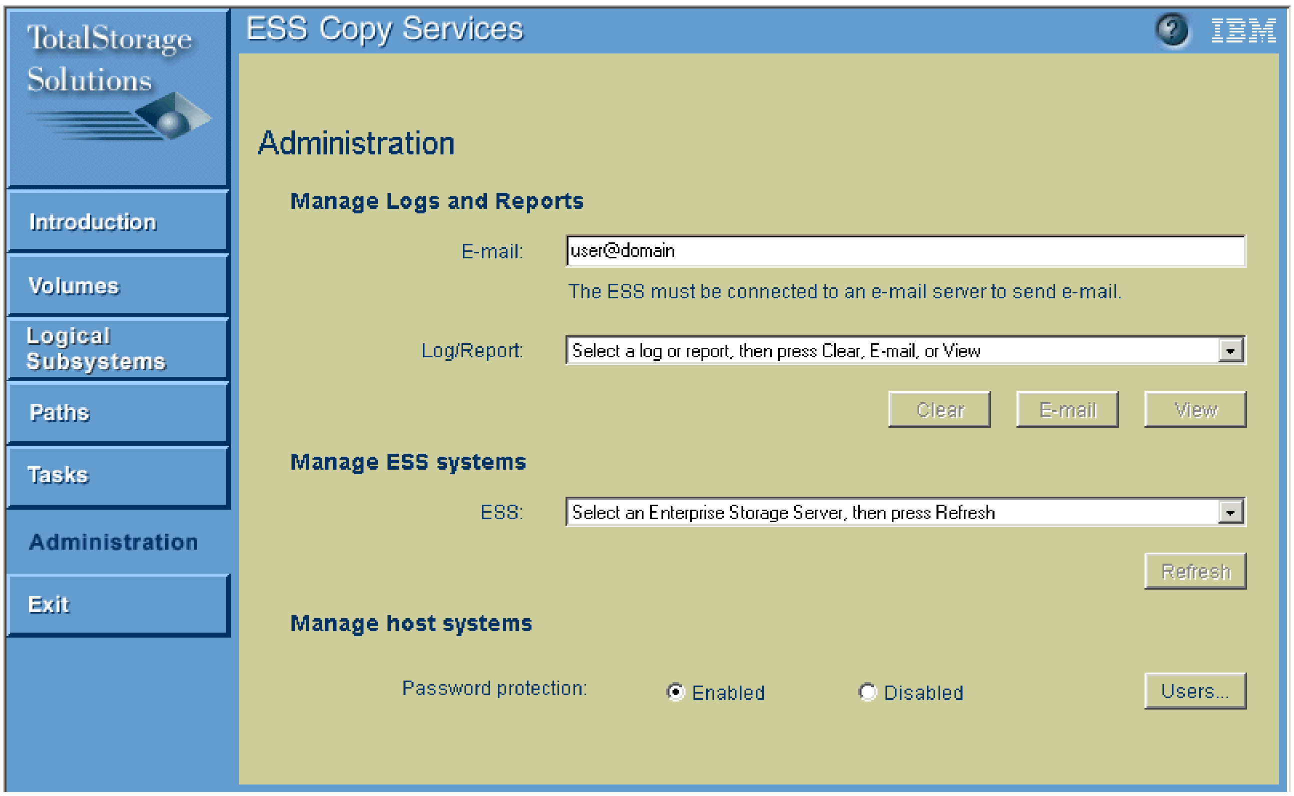



*Figure 7-21 Administration panel*

You can perform the following actions from the Administration panel:

- Clear the server logs
- Manage the CLI user ID and password for an open-systems host
- Define the user ID and password
- Remove the user ID and password
- Refresh the path and volume information for an ESS
- Send the ESS Network Configuration report to an e-mail address
- Send the ESS Resource Configuration report to an e-mail address
- Send the server logs to an e-mail address
- Specify the e-mail address that receives server logs, ESS Network Configuration reports, and ESS Resource Configuration reports
- View the ESS Network Configuration report
- View the ESS Resource Configuration report
- View the server logs

## ESS Copy Services logs and reports

The ESS Copy Services server maintains the following logs:

- Copy Services status log (copyservices.log):

  This log contains messages that are associated with user actions issued through ESS Copy Services. This log includes messages associated with such actions as saving tasks and executing tasks. The log also includes information about the success or failure of the tasks. You can click the *error message* to get a description of the problem and its possible solutions.

  **Important:** The status log is useful for problem determination. Therefore, you should only clear it when no problems exist, or when you have completed data collection for all failures.

- Copy Services Timing log (rsCStiming.log):

  This log contains information about the time it took for data-copy functions (PPRC or FlashCopy) to complete on a specific volume.

- You can view and e-mail the following reports:
  - ESS Network Configuration report:

    The ESS Network Configuration report contains important information about the network-defined ESS resources:

    - IP addresses for primary and backup Copy Services servers
    - IP addresses and host names of ESS clusters
    - IP addresses of ESS clients
  - ESS Resource Configuration report:

    The ESS Resource Configuration report contains the following important information about the ESS resources:

    - 2105 connection information (SAIDs, port types, and values)
    - PPRC path information (SAIDs, LSSs, remote SAIDs, ports, status)
    - S/390 or zSeries volumes (LSSs, labels, Concurrent Copy, FlashCopy, PPRC, or XRC status)
    - Open-systems volumes (LSSs, volume serial numbers, FlashCopy, or PPRC status)

To clear a log, select the log in the log/report list and click the **Clear** button. Please note that you cannot clear a report. You can view the log or report by selecting the appropriate log/report and clicking the **View** button.

To send a report or a log to an e-mail address, select the appropriate log/report in the log/report list, and click the **e-mail** button. Please note that you have to specify the e-mail address in the E-mail field.

For additional information about the Administration panel and how to manage the CLI user ID and password, refer to the IBM publication: *IBM TotalStorage Enterprise Storage Server Web Interface User's Guide,* SC26-7448.

# 7.9 Exiting the ESS Copy Services interface

There are several legitimate ways to quit the ESS Copy Services Web User Interface:

- You can click the **X** at the top right of the ESS Copy Services window.
- You can click **Exit** in the navigation frame of any of the ESS Copy Services panels.
- You can shut down the whole browser from another browser window.

In addition, if you simply want to return to the ESS Web browser interface initial page without closing ESS Copy Services (for example to launch the ESS Specialist), you can keep the ESS Copy Services window open, and use standard navigation methods such as Alt-Tab to switch to the ESS Welcome window.

The difference between the first two options above is nothing more than a warning message. If you click **Exit** in the navigation frame of any of the ESS Copy Services panels, the warning message opens on top of the browser window and tells you that the tasks that you have submitted will continue running after you close the window. Click **OK**. The ESS Copy Services browser window will be closed. In either case, the next time you click the **Copy Services** button on the ESS Web interface Welcome panel, the ESS Copy Services browser window opens without requiring you to log in. The browser loads a cached version of the applet.

Loading the cached applet can be a problem in two situations:

1. If you load a new version of the ESS microcode during that time while you have not quit your browser, the cached applet might not conform with the new code on the ESS.
2. If unauthorized users have access to your workstation when you leave it, they can access the ESS through your cached user ID.

To prevent these situations, you should close all browser windows.

# 7.10 Examples of common procedures

In the following section we give some examples of common uses of Copy Services.

## 7.10.1 Establishing PPRC paths

To establish a PPRC path, go to the ESS Copy Services Path panel (see Figure 7-22) and follow these steps:

1. Select the primary LSS in the Select Source Subsystem pull-down menu.
2. In the Local Port window, select the appropriate SAID.
3. Select the target ESS in the Common Storage Server Targets column.
4. In the Logical Subsystem Targets column, select the target LSS.

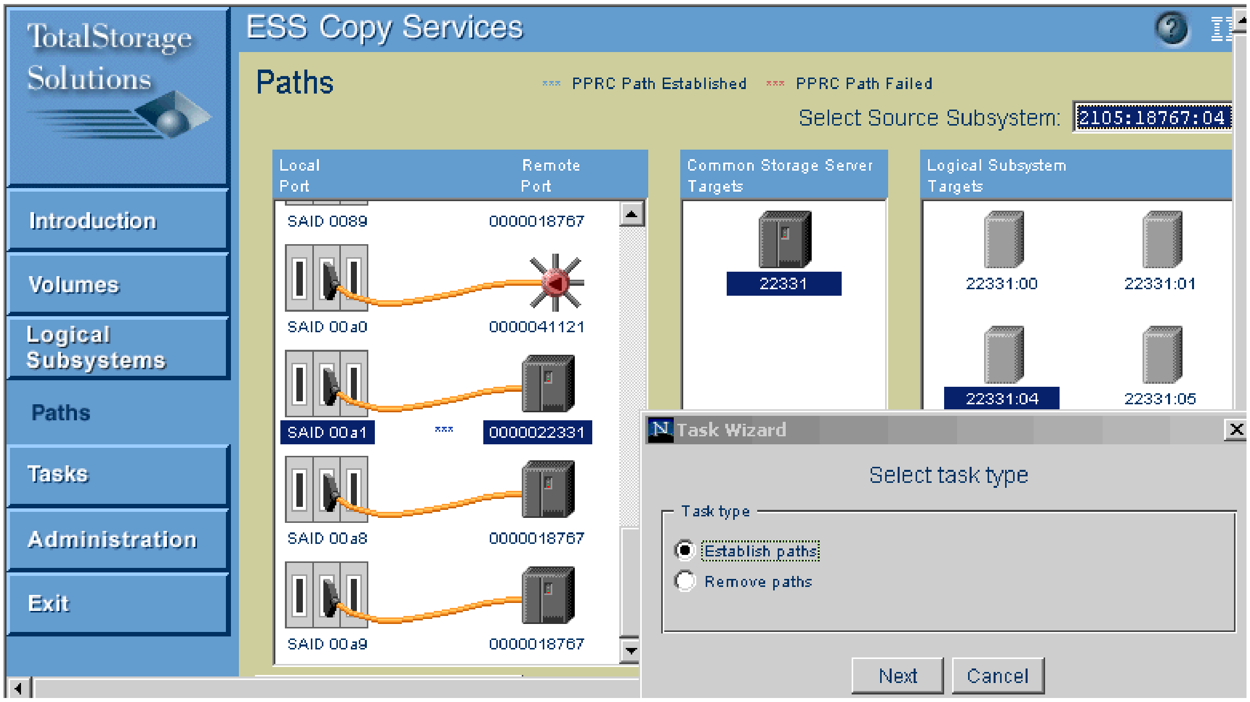



Figure 7-22   Path panel: PPRC establish path

The source subsystem is displayed in the following way: *device type* (4 digits): ESS *serial number* (5 digits): LSS *number* (2 digits). As illustrated in Figure 7-22, we have selected the source subsystem **2105:18767:04**, where 18767 is the primary ESS serial number, and 04 is the primary LSS number.

The local port selected is the ESCON adapter port with the SAID 00a1 (see Appendix E, “System Adapter ID (SAID)” on page 391).

Then the selected target logical subsystem is **22331:04**, where 22331 is the secondary ESS serial number, and 04 is the secondary LSS number.

5. In the Select Task Type panel select **Establish paths** and click **Next**.
6. When the Select Path Options panel pops-up, select **Do not establish paths if they already exist**, as Figure 7-23 illustrates.

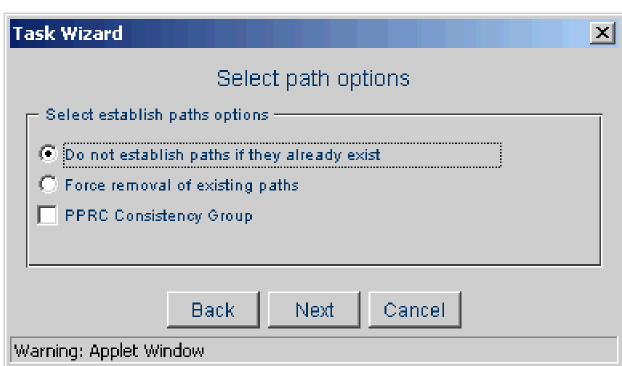



Figure 7-23   PPRC establish path - Select path option panel

7. Click **Next** and then name and save the task for later execution.

## 7.10.2 Establishing Synchronous PPRC pairs

Once the PPRC paths have been defined it is possible to establish PPRC pairs. To establish a PPRC pair, go to the ESS Copy Services Volume panel (see Figure 7-24) and proceed as follows:

1. Select the Source and Target LSSs.

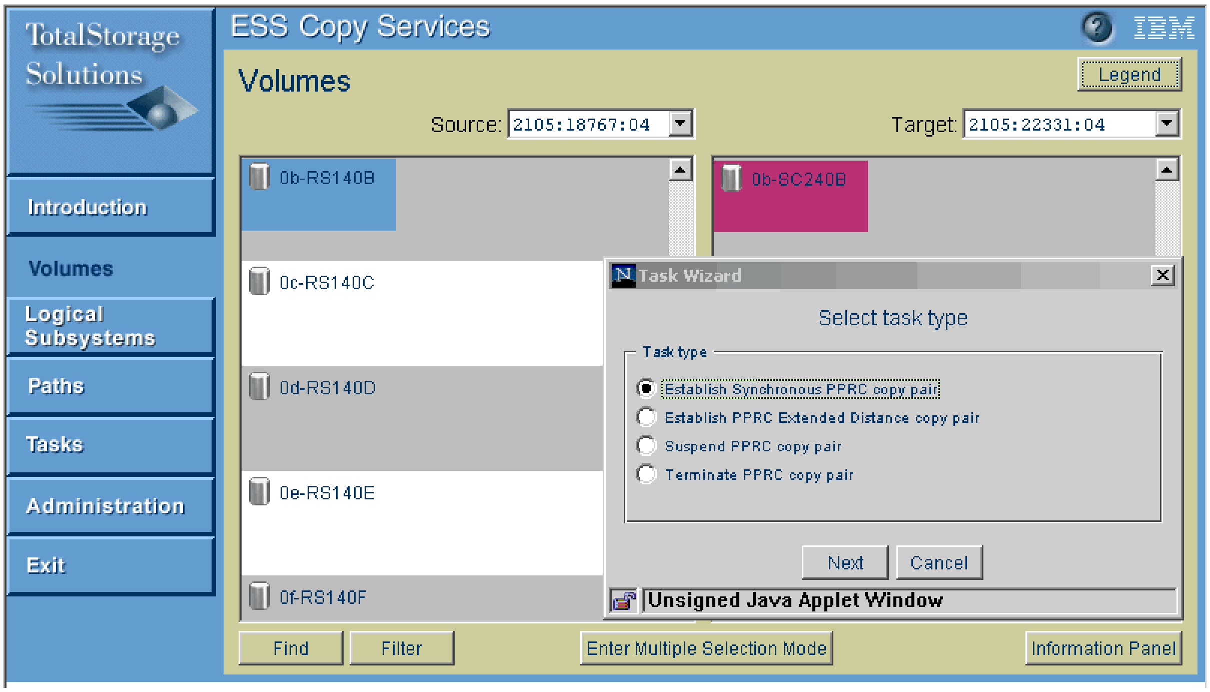



*Figure 7-24 Volumes panel: PPRC establish pairs*

2. In the Source window, select the primary PPRC volume.
3. In the Target window, select the secondary PPRC volume (right-click). Right-click again so the Select Task Type panel pops-up.
4. In the Select Task Type panel (see Figure 7-24), select **Establish Synchronous PPRC copy pair and** click **Next.**
5. When the Select Copy Option panel pops-up (see Figure 7-25), check the option you need for the task you are creating. In our example we selected **Copy entire volume**.

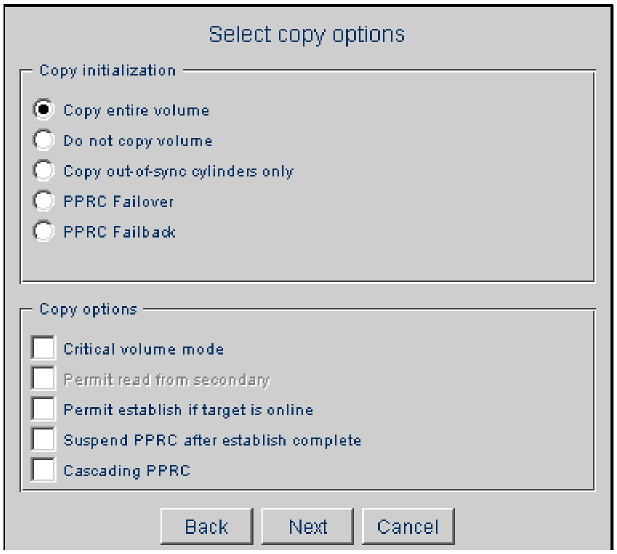



*Figure 7-25 PPRC establish pairs - Select Copy Option*

6. Click **Next** and then name and save the task for later execution.

## 7.10.3 Establishing PPRC-XD pairs

When using the Web User Interface, you begin as usual by selecting the primary and secondary volumes from the Volumes panel in order to launch the Select Task Type wizard panel. You select the **Establish Extended Distance PPRC copy pair** option (see Figure 7-26) to either initially establish a pair in a XD mode, or to re-establish a suspended pair.

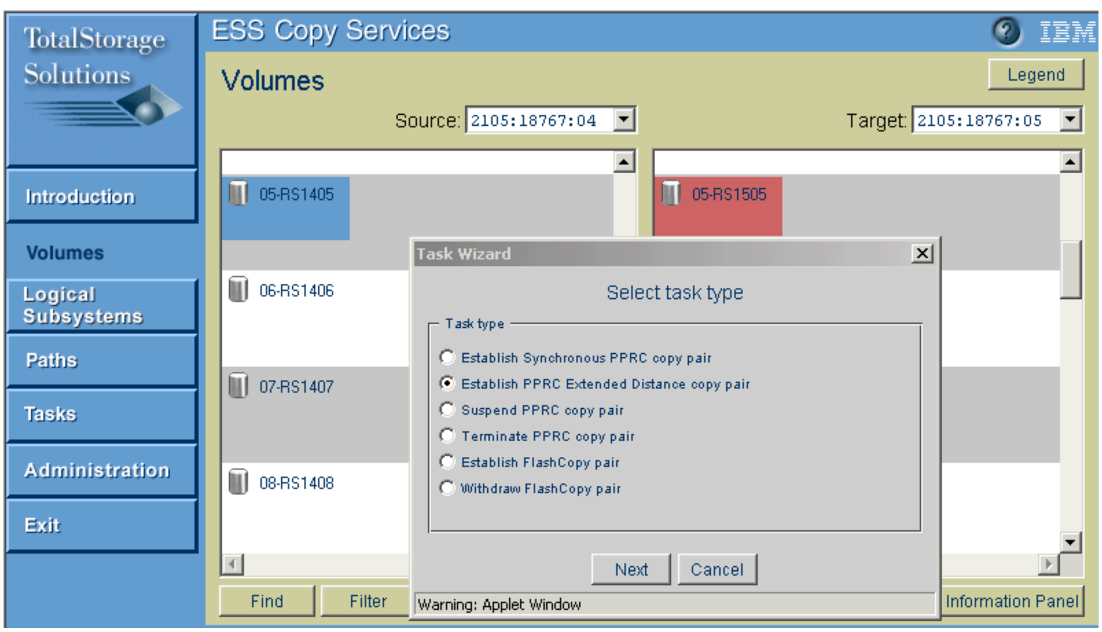



*Figure 7-26 Selecting the volume pair and launching the Task Wizard*

After selecting the **Establish Extended Distance PPRC copy pair** option and clicking the **Next** button, you will be able to select the copy options from the Select copy options window that pops up (see Figure 7-27). As you can see in that panel, you can either select the option **Copy entire volume**, or the option **Copy out-of-sync cylinders only**. You use the first option to do an initial copy of a newly established pair — going from simplex to *duplex-pending XD* state. You use the second option to re-synchronize a suspended pair — going from suspended to *duplex-pending XD* state.

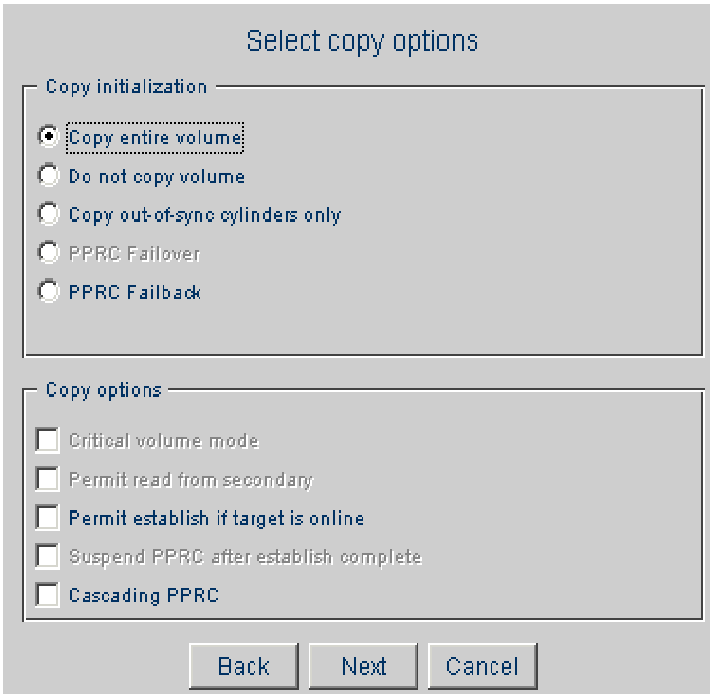



*Figure 7-27 Selecting PPRC-XD copy options*

Table 7-6 summarizes the copy options that will be used in the ESS Copy Services WUI panels when working with XD pairs. Refer also to Table 7-6 for the copy options when doing a go-to-SYNC transition.

*Table 7-6 ESS Copy Services Web User Interface options for the PPRC-XD transitions*

| **Operation** | **Volume pair state** | | **WUI options** | |
|---|---|---|---|---|
| | **from** | **to** | **Select task type** | **Select copy options** |
| Establish initial copy pair | *simplex* | *duplex pending* XD | Establish Extended Distance PPRC | Copy entire volume |
| Reestablish suspended pair | *suspended* | *duplex pending* XD | Establish Extended Distance PPRC | Copy out-of-sync cylinder only |

## 7.10.4 Establishing Asynchronous Cascading PPRC pairs

In this section we present an example — using ESCON paths — of how an Asynchronous Cascading PPRC configuration can be established using the Web User Interface. The configuration in this example is the recommended configuration, where the local PPRC pair is in synchronous (SYNC) mode, and the remote PPRC pair is in non-synchronous (XD) mode. The configuration is illustrated in Figure 3-22 on page 112, and the setup procedure is done as follows:

1. Create the PPRC logical paths between the local and intermediate sites. In Figure 7-28, the selected local LSS is `2105:18767:0a`, where 18767 is the ESS serial number and 0a is the LSS number.

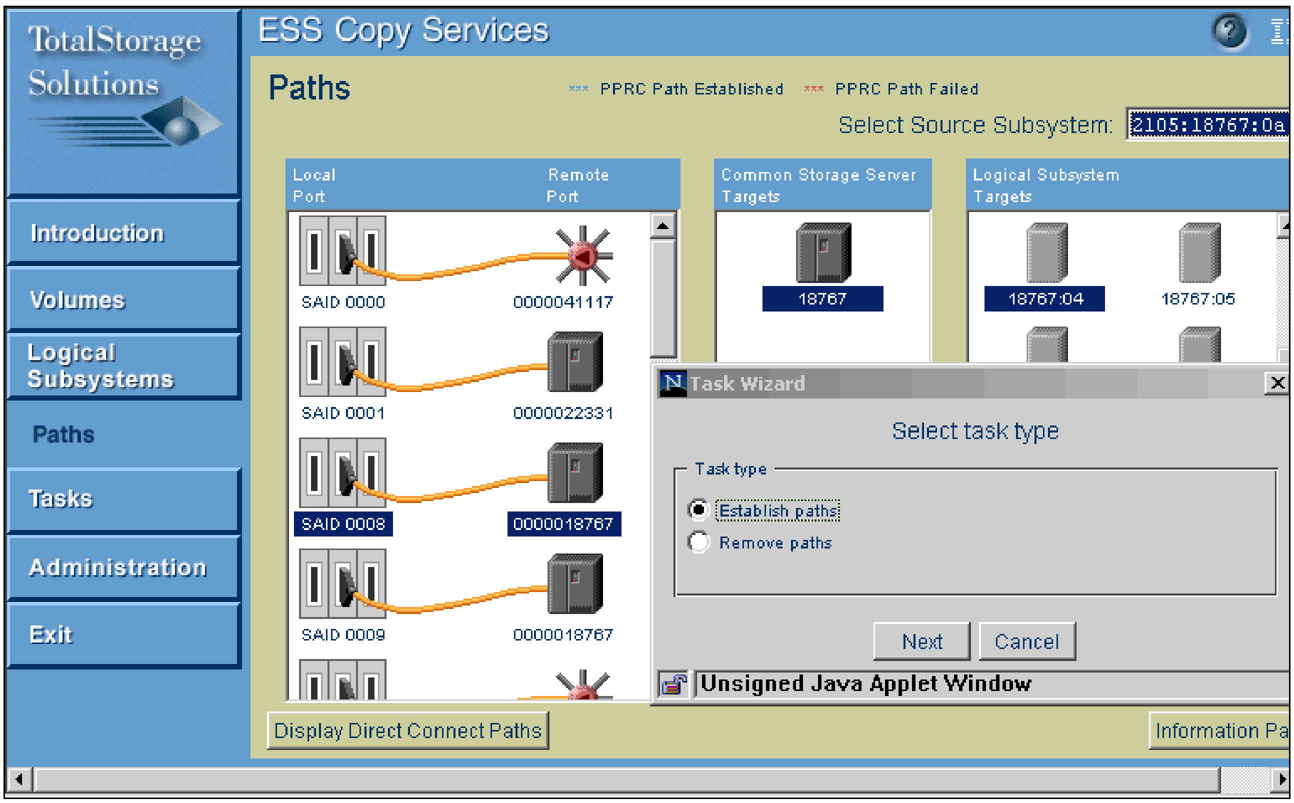



*Figure 7-28 PPRC Establish path: Local to intermediate site*

In our example, the local site and the intermediate site volumes are in the same ESS but on different LSSs. The intermediate site LSS number is 04. The path origin selected corresponds to the ESCON host adapter port with the SAID 0008 (see Appendix E, “System Adapter ID (SAID)” on page 391).

2. Establish the PPRC paths between the intermediate and the remote site as shown in Figure 7-29.

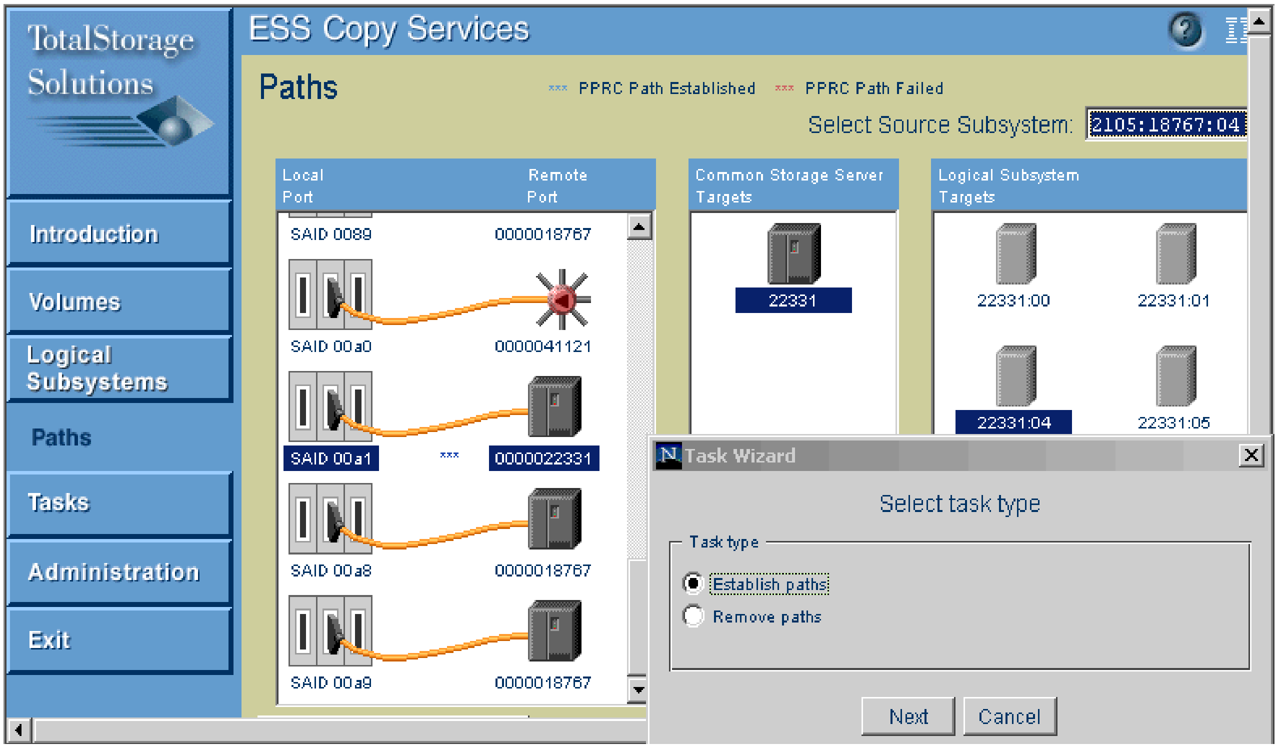



*Figure 7-29 PPRC Establish path - Intermediate to remote site*

As you can see in the Logical Subsystem Targets window in Figure 7-29, the remote site logical subsystem `04` is in the ESS with serial number 22331.

3. Once the logical paths are defined, establish the PPRC-XD non-synchronous relationships for the remote PPRC pairs (intermediate site to remote site) as shown in Figure 7-30.

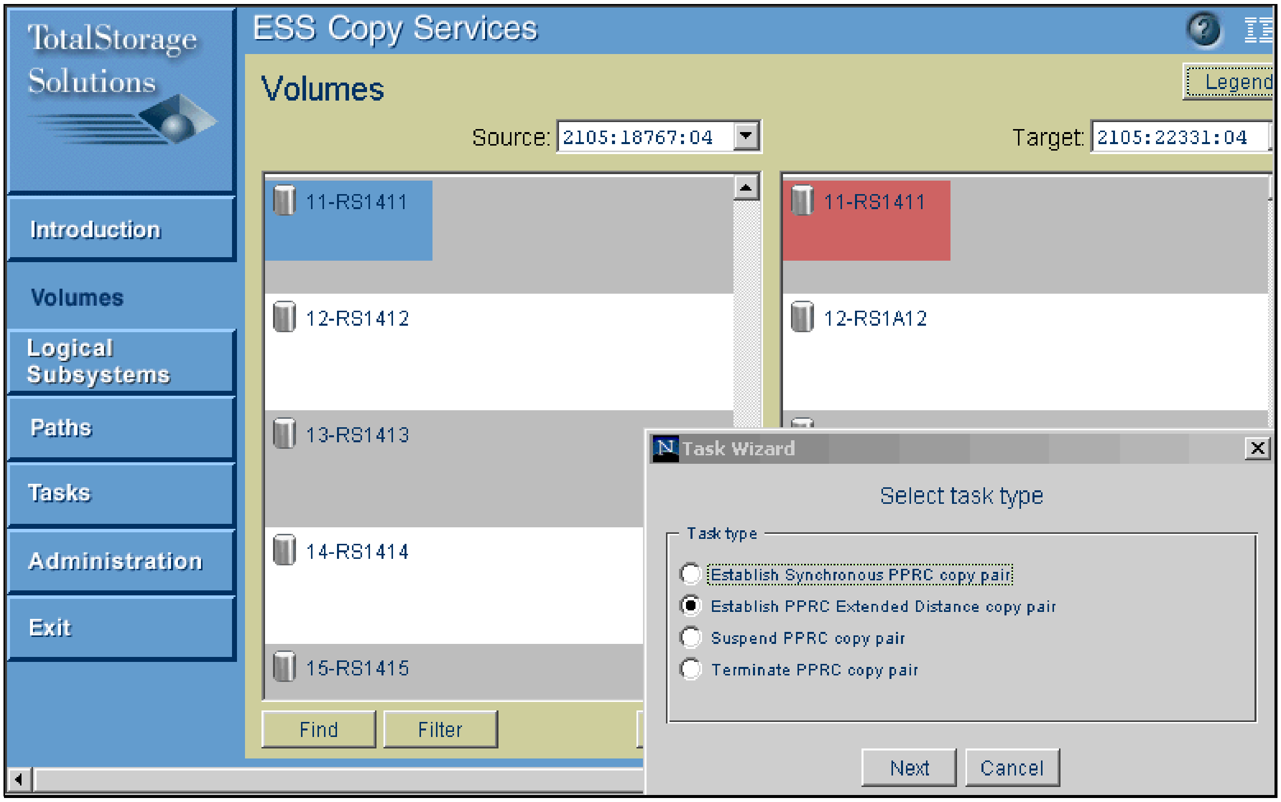



*Figure 7-30 Establish PPRC XD volume pair relationships- Intermediate to remote site*

So far in our example, the remote PPRC pairs consist of volumes in LSS 04 of the intermediate site ESS (serial number 18767) and volumes in LSS 04 of the remote site ESS (serial number 22331).

When the Select Copy Options panel pops up, select the **Do not copy volume** and the **Cascading PPRC** options (see Figure 7-31).

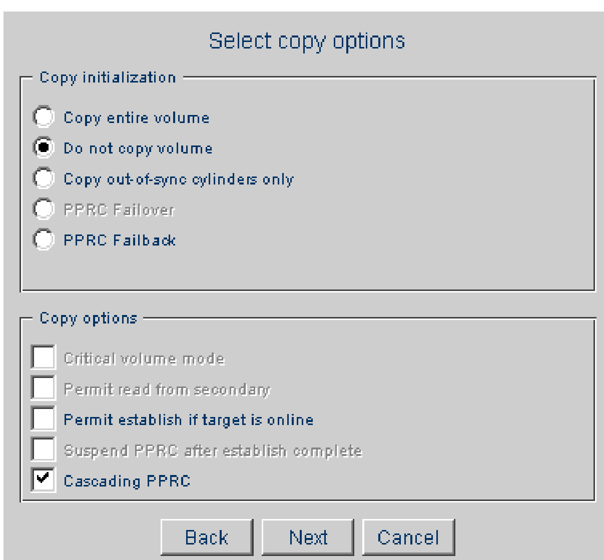



*Figure 7-31 Select Copy Options panel - Cascading PPRC*

4. We can check if the volumes at the intermediate site are eligible for cascading from the information in the Information Panel (PPRC Status in Figure 7-32).

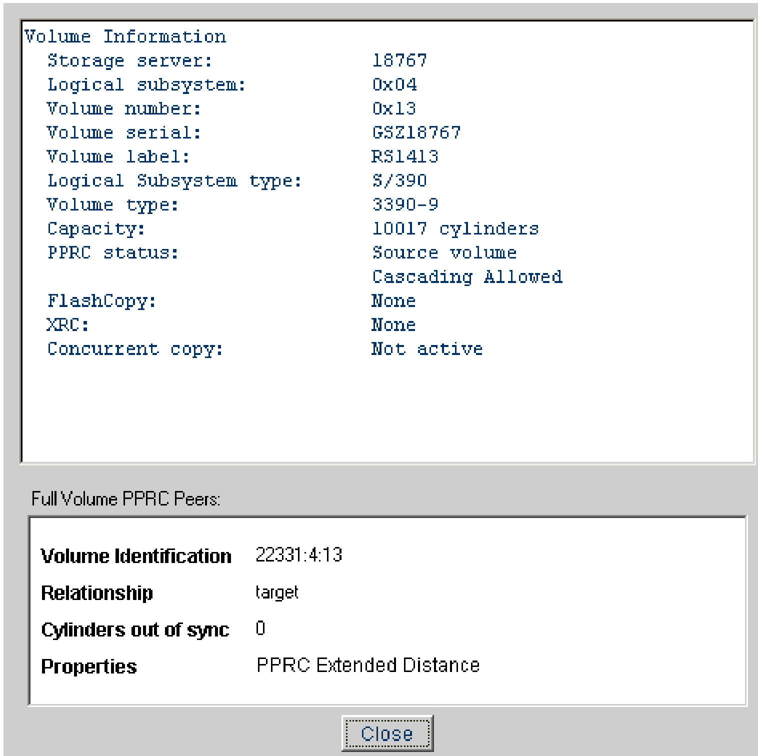



*Figure 7-32 Volume information panel - Cascading allowed*

5. Once the remote PPRC pairs are established in extended distance mode, you can proceed to establish the Synchronous PPRC relationships for the local PPRC pairs (local site to intermediate site), as illustrated in Figure 7-33.

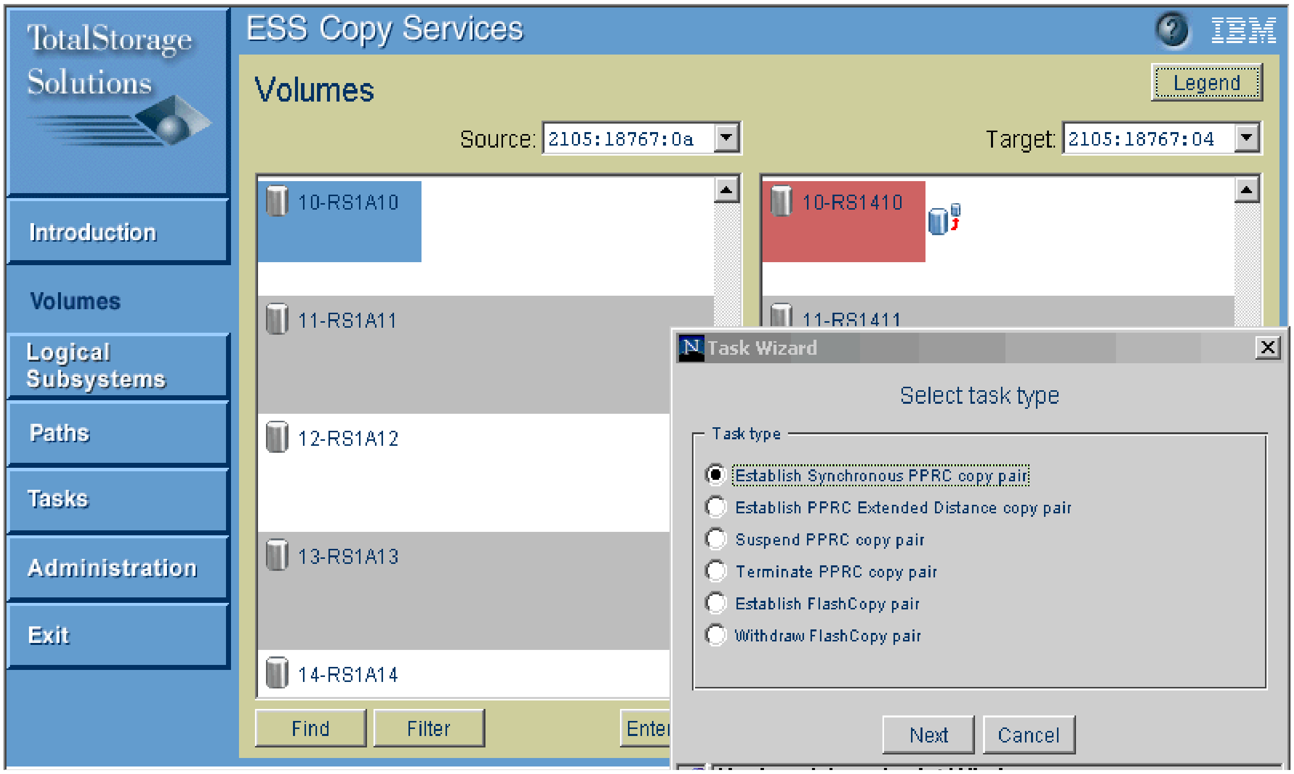



*Figure 7-33 Establishing PPRC-SYNC volume pairs - Local to Intermediate sites*

In our example, the local and intermediate volumes are in the same ESS (serial number 18767) but on different LSSs. The primary volumes are in LSS 0a and the secondary volumes are in LSS 04.

As shown in Figure 7-34, when the Select Copy Options panel pops up, the **Copy entire volume** option is selected.

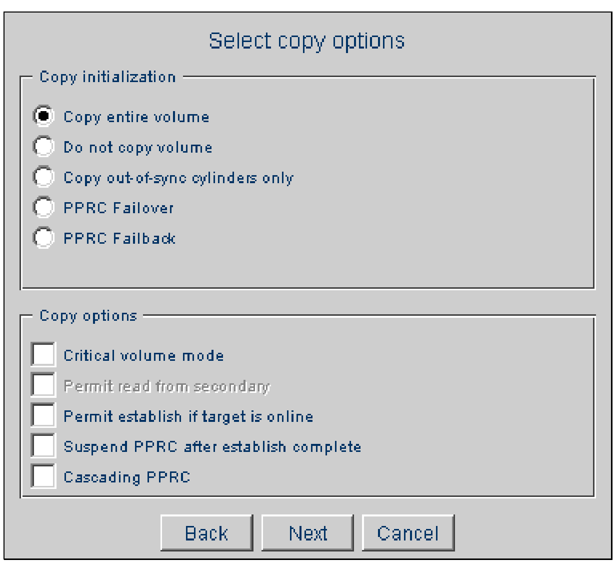



*Figure 7-34 Select Copy Options panel - Copy entire volume*

6. Once the local PPRC pair volumes reach the duplex state you will see in the Web interface Volumes panel that the intermediate volume pairs will present both the *primary full duplex* icon and the *secondary duplex pending-XD* icon. This is how volume 10-RS1410 appears in Figure 7-35.

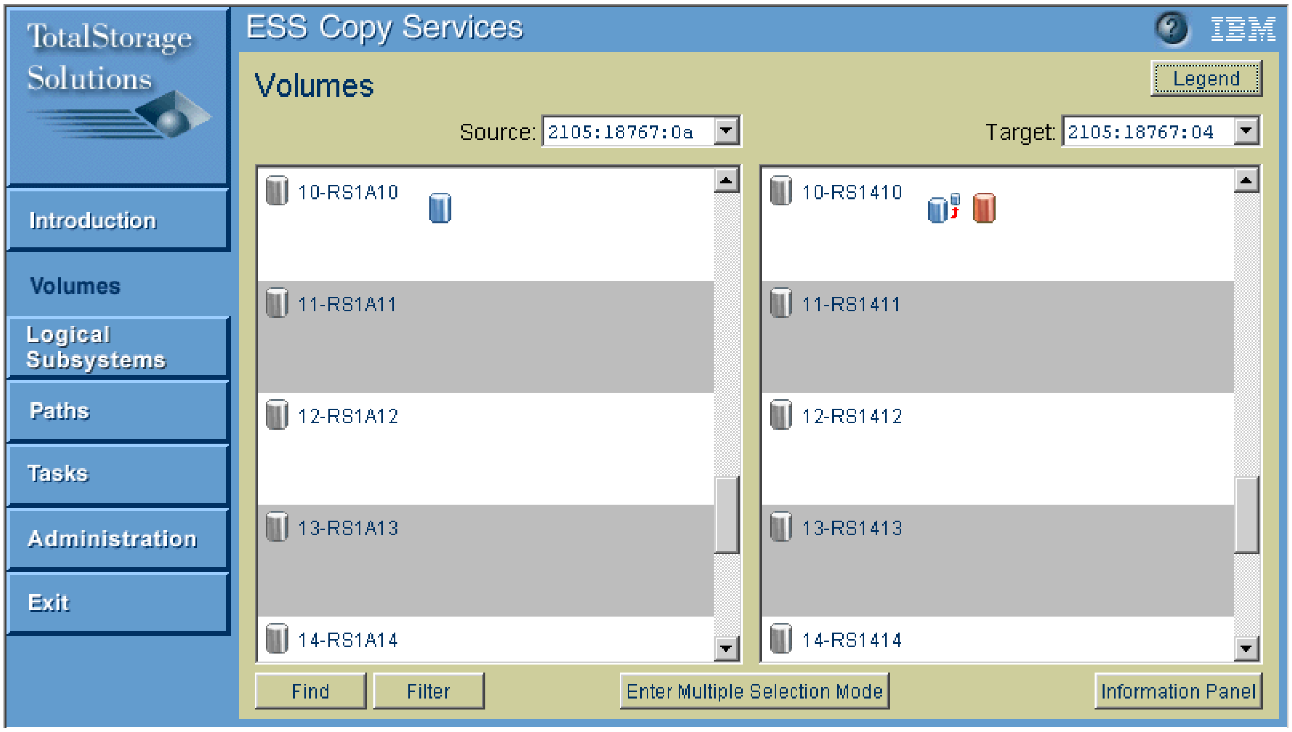



*Figure 7-35 Volumes panel - Intermediate volume enabled for cascading*

This combination of icons indicates that volume 10-RS1410 is in a PPRC-XD relationship as primary volume in the remote PPRC pair (blue icon), and at the same time it is in a PPRC-SYNC relationship as the secondary volume in the local PPRC pair (red icon).

## 7.10.5 Establishing Asynchronous PPRC

Asynchronous PPRC provides a long distance copy solution based on non-synchronous PPRC-XD technology, that also gives recoverable point-in-time consistency at the remote site with no I/O performance degradation. Repeated consistent copies of all the volumes in the Asynchronous PPRC session are taken with only a short delay between them. This delay can be set by the user, but generally will be less than 10 seconds. The age of the data at the remote site is no more than the time since the last copy was taken, but more importantly, because the volumes are all at the same point-in-time, consistency is maintained and recovery is faster and with less data loss.

This is discussed in detail in Chapter 3, “Peer-to-Peer Remote Copy” on page 69. In this section we will describe the steps to use and manage Asynchronous PPRC using the Web User Interface provided with the ESS.

The following section is broken up into two parts:

- The first part explains how to set up Asynchronous PPRC. Step-by-step procedures are given, showing how to create the environment, set up the volume relationships, and build the session. See 7.10.6, “Setting up Asynchronous PPRC” on page 337.

- The next part shows you how to start, monitor, and manage the Asynchronous PPRC session. See 7.10.7, “Starting Asynchronous PPRC” on page 346.

> **Recommendation:** It is not recommended to use the Web User Interface to manage a large number of volumes in an Asynchronous PPRC Session. This is due to the fact that during recovery, the state of every volume must be checked individually to determine its state of consistency and action taken on each to either revert or commit the FlashCopy. For a small number of volumes, this is reasonably straightforward. But for a large number, such as hundreds or thousands of volumes, we recommend the use of Multiple Device Manager Replication Manager; see 4.6, “Multiple Device Manager Replication Manager” on page 176, or Asynchronous PPRC Utilities for Open System Environment; see 4.7, “Asynchronous PPRC Utilities for Open System Environment” on page 177.

For any procedure that is run from the WUI, the last screen that comes up is similar to Figure 7-36 which gives you some options. If the task you are doing is only being used once or you do not require the ability to re-run it repeatedly, then from this screen all you do is click the **RUN** button. If the task needs to be repeated or needs to be run from the command line interface (CLI), then you should save it.

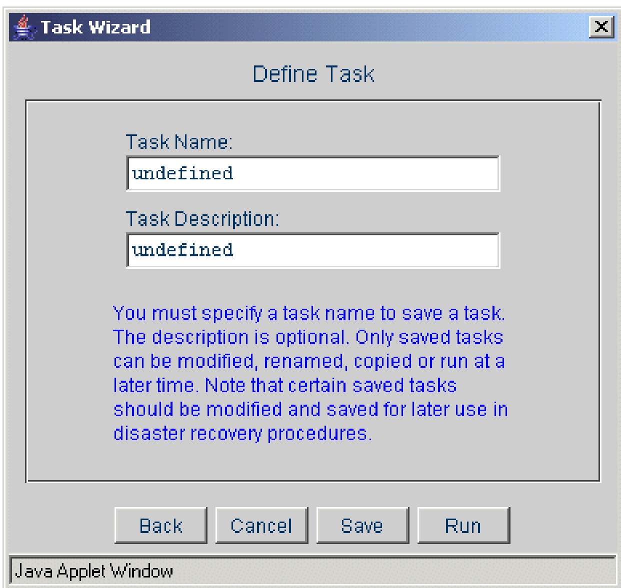



*Figure 7-36 Define Task wizard*

To save a task, give it a unique task name and add a full text description to help you identify its function from the task list. Next, click **SAVE** to permanently save this task, so it can be run again at another time. Tasks can also be called from the CLI command interface.

## 7.10.6 Setting up Asynchronous PPRC

The steps required to set up a volume or volumes in an Asynchronous PPRC relationship are as follows:

1. Establish paths from each primary LSS to the secondary LSS that will have a pair
2. Establish PPRC-XD pairs A →B
3. Establish FlashCopy pairs B →C
4. Define an Asynchronous PPRC session
5. Add the volumes to the session

## Paths

Every PPRC pair relationship needs a PPRC path. These are logical unidirectional connections between LSSs and can be either over ESCON links or FCP. PPRC paths are discussed in 3.10, “PPRC links” on page 125. Only FCP links can be used for Asynchronous PPRC.

These paths are only in one direction, from the primary volume to the secondary volume. If PPRC pairs exist in the reverse direction, then a reverse path must be established. The paths are from one LSS to another LSS, so any volume in an LSS can use a path that exists to the LSS that its secondary is in, but a new path must exist for every LSS to LSS relationship. More than one path can be implemented in parallel and this is recommended for redundancy.

For this example, we will establish one path.

Referring to Figure 7-37, click the **Paths** button to bring up the Paths panel.

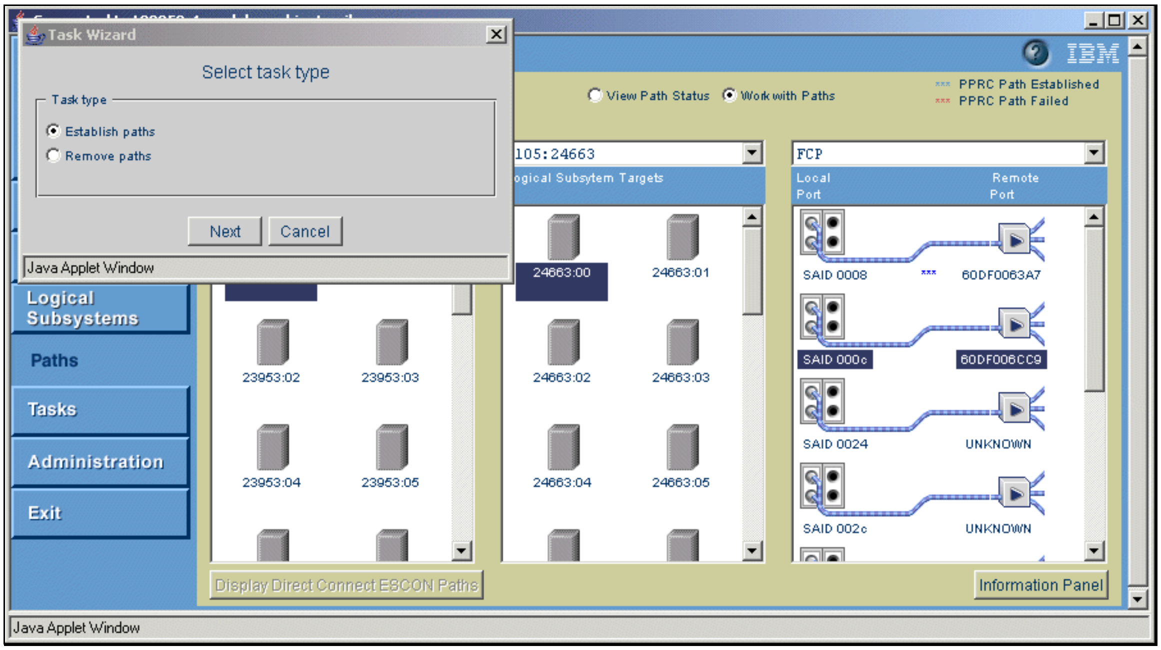



*Figure 7-37 Establishing paths: Paths panel*

In the left panel, click the pull-down and select the source ESS. All ESSs in this Copy Services Domain will be listed. Now left-click the primary LSS that this path will be from. Click the **Work with Paths** button at the top of the panel, then click the pull-down in the center panel and select the ESS that contains the secondary LSS. Left-click the secondary LSS.

The right panel shows the links available. The default display is ESCON. Click the pull-down to select FCP. Next left-click the adapter you wish to use for your link. Now right-click to launch the Task Wizard.

> **Tip:** If you want to configure multiple paths at the same time, left-click the first path, then right-click subsequent paths to enable multiple selection. You need to right-click the last one highlighted to launch the Task Wizard.

The task wizard as shown on Figure 7-38 will launch. Select **Establish Paths**.

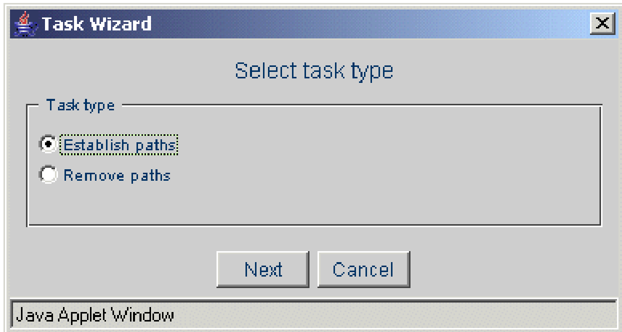



*Figure 7-38 Establishing Paths, Establish Paths*

If a switch is attached to the port, then the Select outgoing ports wizard will appear as shown in Figure 7-39. A pull-down of available ports from the switch will be shown. Select the correct port for the path you are establishing, click **Next**.

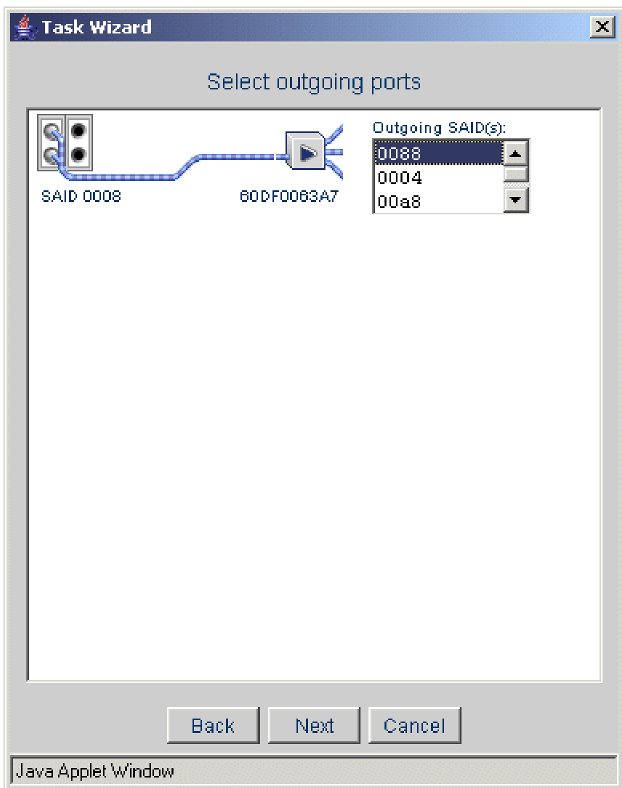



*Figure 7-39 Establishing Paths, outgoing ports*

The Select Path Options wizard panel, will appear next. This allows for the creation of a Consistency Group for all PPRC relationships using this path. This is not required for Asynchronous PPRC, so leave it un-selected. Click **Next**.

The next screen is the Task Wizard. If you wish to save this task for later reuse, fill in the fields and click **Save**. Next click **Run**.

This will return you to the main panel. The path now has 3 blue *** underneath it. This indicates that the path is established from the selected LSS on the left to the selected LSS in the middle.

**Tip:** To display the path, the correct primary and secondary LSSs must be selected. The path panel on the right is unique to the selections made on the left and middle panels.

## Creating PPRC-XD pairs

This portion of Asynchronous PPRC uses the same setup as for a normal PPRC-XD pair. First click the **Volumes** button to display the Volumes panel. Click the pull-down on the left (source) panel and select the ESS:LSS combination that will be your source LSS. Do the same on the right panel to select the target LSS.

For single selections, left-click once on the source and right-click once on the target. Now right-click again on the target to launch the wizard.

You can perform the same operation on multiple volumes. To do this, first click the **Enable Multiple Selection Mode** button at the bottom. Now left-click one volume in the left panel, then right-click its pair in the right panel. Then left-click the next volume source in the left panel and right-click *its* pair in the right panel. Continue, adding as many pairs as required. If you do this out of sequence, the panels will reset and all pairs will be lost, so start again.

Once all the pairs have been highlighted, right-click the last one clicked on the right panel. This will launch the wizard as shown in Figure 7-40, where we have multi selected four volumes.

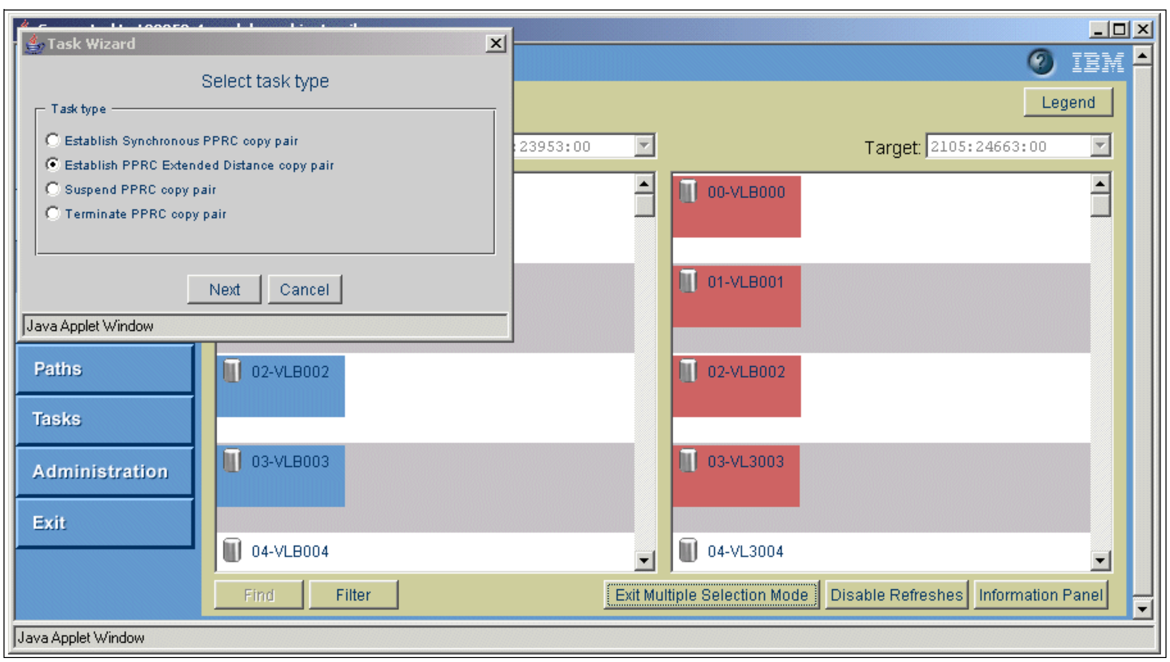



*Figure 7-40 Creating PPRC-XD pairs, select volumes*

Select **Establish PPRC Extended Distance copy pair** and click **Next**.

This will bring up the window shown in Figure 7-41. Select **Copy entire volume** and click **Next**. You will once again get the task save wizard, save this task if you wish. Click **Run**.

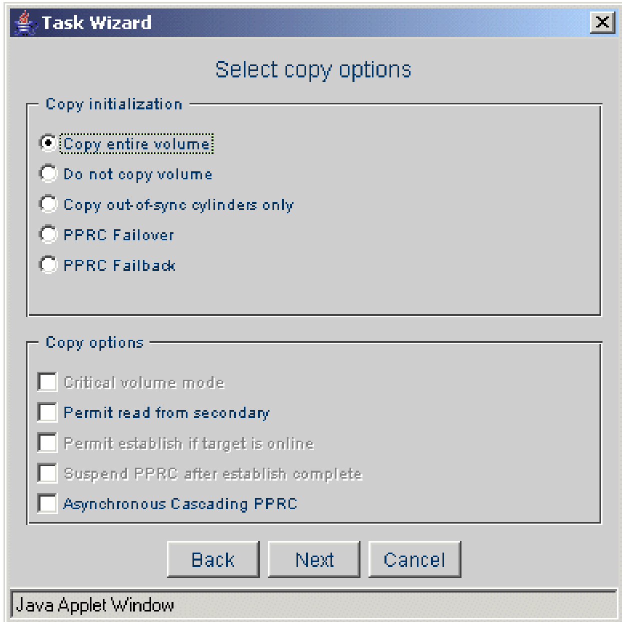



*Figure 7-41 Creating PPRC-XD pairs, copy options*

You will now be returned to the volumes panel and the status of the volumes you selected will have changed to PPRC-XD as shown by the icon next to the volume. We now have our A volumes, the primary (or source) of the PPRC pairs and our B volumes, which are the secondaries (or target). These B volumes are normally located in another ESS at the remote site.

### Creating FlashCopy pairs

Now that you have the PPRC-XD pairs set up, the next step is to create a FlashCopy relationship for each of these. This FlashCopy uses the PPRC secondary as its source, and now a third volume in the remote ESS. The source is our B volume and the target will be our C volume.

The first step is to select the ESS:LSS on the volume panel for our B volumes, currently the targets of our PPRC. Next select the ESS:LSS in the right (target) panel that contains the volume(s) we will be using or the C volumes. Using either multiple selection as described earlier or individually, left-click the source volume on the left and right-click the FlashCopy target on the right. Once all selections are made, right-click again on the last volume highlighted to launch the wizard as shown in Figure 7-42.

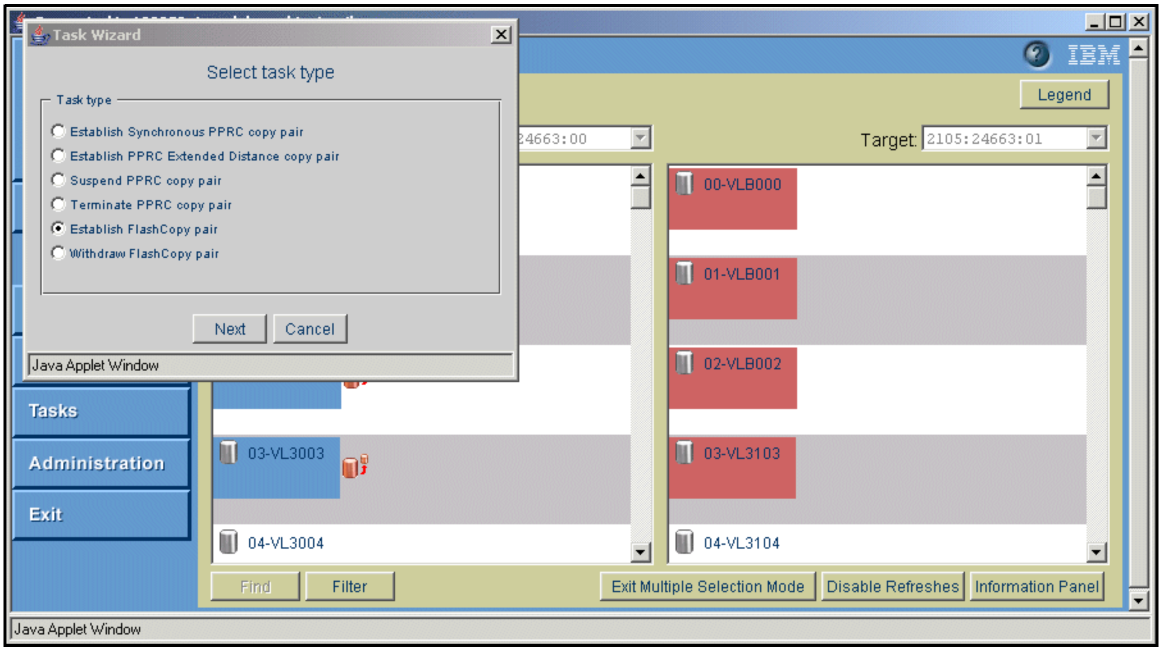



*Figure 7-42 Creating FlashCopy pairs, selecting volumes*

Select **Establish FlashCopy pair** from the window and click **Next**.

This will bring up the copy options task wizard as shown in Figure 7-43. It is important to select the correct option here. Select the options from this panel as follows:

- **No background copy:** This option prevents copying of all the tracks to the target. Only changed tracks are copied. This is necessary to speed up recovery later.
- **Persistent FlashCopy:** This option maintains the relationship after FlashCopies complete. This will be forced by the change recording option below anyway.
- **Inhibit writes to target:** This option prevents the C volume from being altered. We need this intact as our recovery volume.
- **Start change recording:** This option records changes made to the source after a FlashCopy has been made so that when another one is performed, only tracks that changed in that period need to be copied over.

Click **Next**.

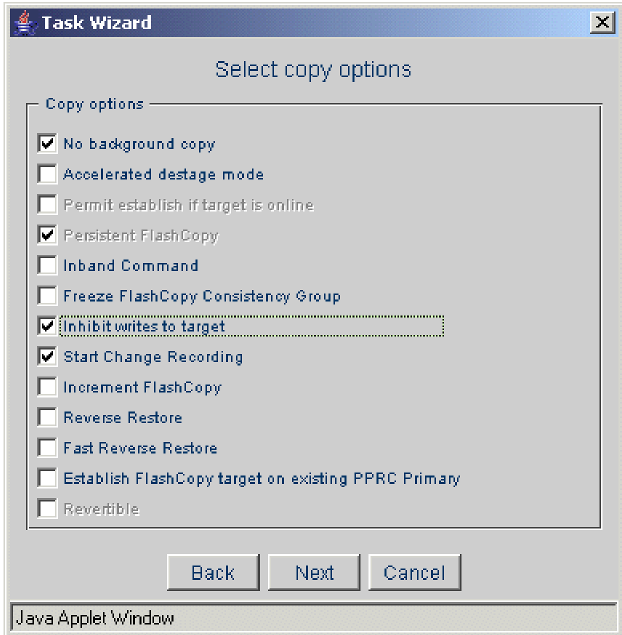



*Figure 7-43 Creating FlashCopy pairs, copy options*

The next window allows you to modify the sequence number of the FlashCopies. This can aid in analyzing multiple Consistency Groups in a large enterprise, otherwise leave the default of 0, no prefix. Click **Next**. You will once again get the task save wizard, save this task if you wish. Click **Run**.

Once you return to the volumes panel, you will see the combined status as shown in Figure 7-44. Notice that the B volumes in the left panel show the PPRC-XD secondary icon, the FlashCopy source icon and the FlashCopy change recording icon. In the right panel are our C volumes, showing FlashCopy target and change recording icons.

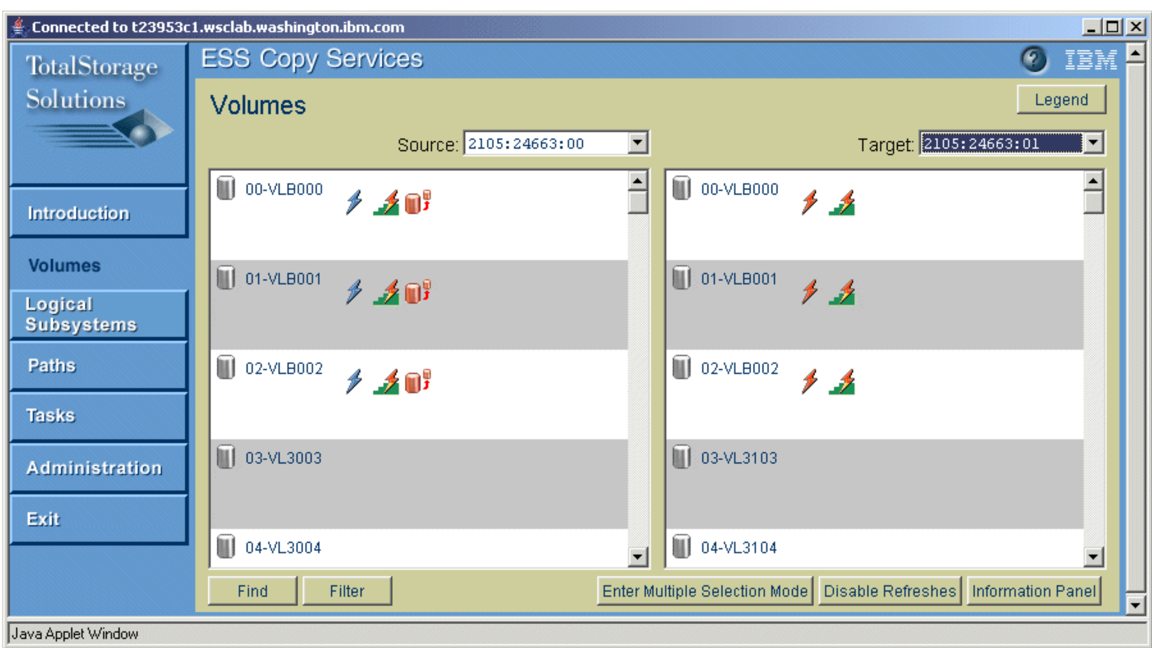



*Figure 7-44 Creating FlashCopy pairs and viewing results*

We now have our volume relationships set up. Next is to define the session that Asynchronous PPRC will use to maintain consistency across its members.

## Define Asynchronous PPRC Session

Asynchronous PPRC uses a session to manage the volumes you wish to keep consistent. All volumes that are members of the session must have consistent C volumes at the remote site. This process is controlled by the Asynchronous PPRC Master.

First we need to define a session for each LSS that contains volumes we will want to be included in that session. Then we need to add those volumes to the session. Note that only one session can be defined for an LSS.

To define a session for an LSS, we first click the **Logical Subsystems** button to bring up the LSS window. Refer to Figure 7-45. Left-click the LSS you are defining to highlight it. Now right-click it to bring up the wizard as shown. Select **Manage Session** and click **Next**.

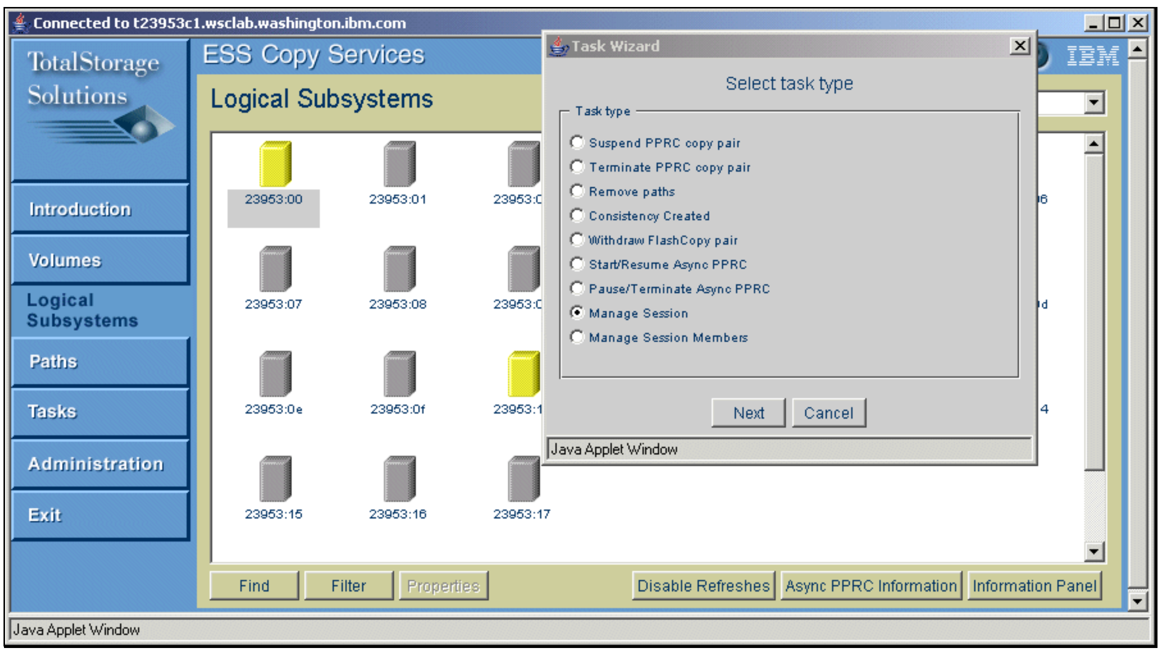



*Figure 7-45 Defining Asynchronous PPRC Session, selecting LSS*

The following window gives you several options. You can close a session you already have defined on this LSS, or you can create one, either with a new session id, or by using a session id already defined in the ESS Copy Services Domain. The pull-down next to the select session id button will show all sessions currently defined in this domain. This is useful to make sure that you have all your LSSs in the same session and prevent a typing error. If this is a new session, as in our case, then enter a number in the id entry field at the bottom. This can be any number from 1 to 255.

When you have the fields filled out, click **Next**. You will once again get the task save wizard, save this task if you wish. Click **Run**.

This will return you to the LSS display panel. Repeat this process for every LSS you have volumes in that are to be included in this session. With a large number of volumes, it will be very useful to save each of these setup tasks and automate them from the CLI using scripts.

Next we need to add our volumes to this newly created session.

## Add volumes to session

As discussed in the previous paragraph, we now need to add the volumes individually to the session we have just created. Again, with a large number of volumes, it will be good practice to save the tasks as you create them, so they can be automated through the CLI or from the tasks management.

Click the **Volumes** button to display the Volumes panel. Singly or multiply, select the volumes you wish to include in this session. Note that with multiple select, after clicking the **Multiple select** button at the bottom, left-click the first source volume, then right-click subsequent volumes. Right-click the last volume you selected to launch the wizard as shown in Figure 7-46.

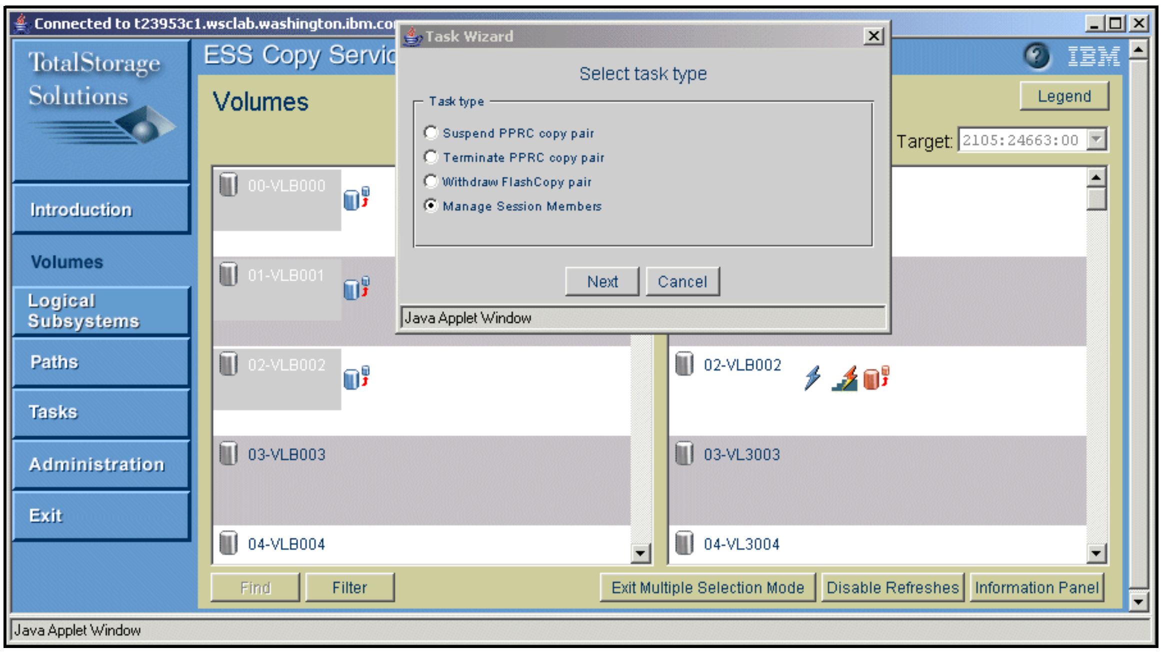



*Figure 7-46 Adding Volumes to Session, selecting volumes*

Select **Manage Session Members**, click **Next**.

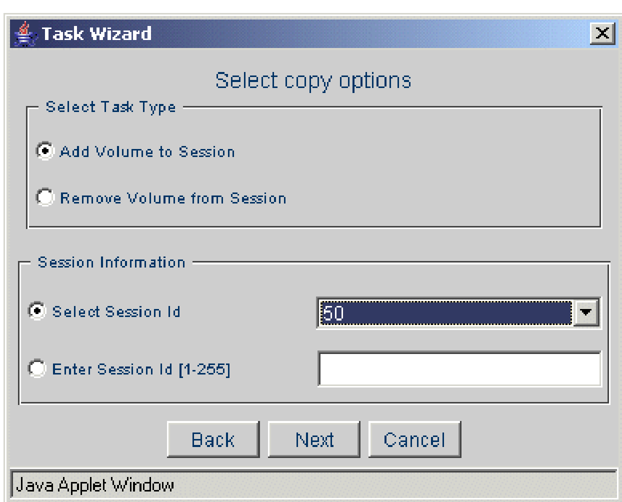



*Figure 7-47 Adding Volumes to session, selecting session*

From the options window (see Figure 7-47), select **Add Volume to Session** and from the pull-down, select a session. We have selected the session we created for this LSS above. You can also use this panel to remove the members from a session. Individual volumes or groups of volumes can be added or removed from any session, even if the session is running.

Once you have selected the session id, click **Next**.

You will once again get the task save wizard, save this task if you wish. Click **Run**.

This will return you to the Volumes panel as shown in Figure 7-48. Notice that there is an icon next to the volume indicating that the volume is *session join pending* (3 small volumes with a larger one beneath and an arrow up). If the session is not yet started, as in our case, this will remain in a pending state until the session starts.

If the session is already started, then this icon will not change to the member icon until it is able to form consistent FlashCopies. This may take a period of time if the PPRC-XD initial copy is still in progress. Once the volume is able to form a FlashCopy in consistency with the other volumes already in the running session, then the Master will join it to the session. From that time on, it will remain a member until removed.

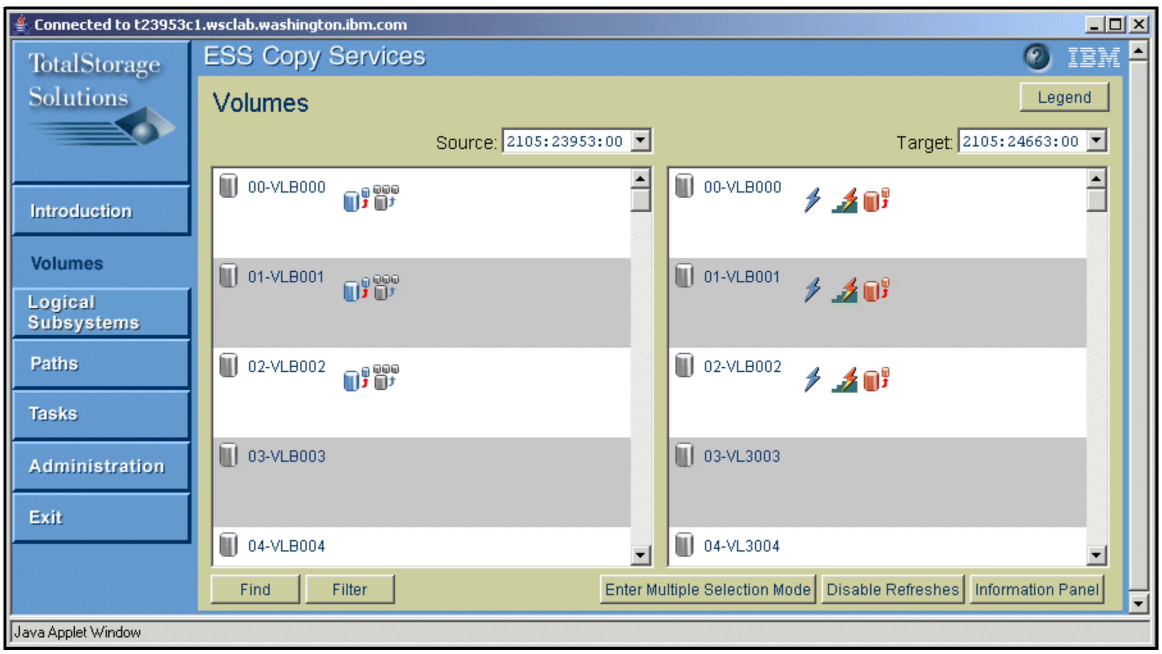



*Figure 7-48 Adding Volumes to Session: Viewing status*

## 7.10.7 Starting Asynchronous PPRC

If we have LSSs with sessions of the same id that we want included in this Asynchronous PPRC session that are in other ESSs, then we must ensure that there are communication paths from the Master ESS to the Subordinate ESSs. These Master - Subordinate communications will pass over any normal PPRC paths that are set up in the Master ESS to Subordinate ESS direction using any LSS in either ESS. If a path exists for another PPRC pair relationship, this can be shared, otherwise a new path needs to be created.

Use the create path procedure earlier in this chapter; see 7.10.1, “Establishing PPRC paths” on page 326. Or, in the section above, where we created the paths for the PPRC-XD, see 7.10.6, “Setting up Asynchronous PPRC” on page 337, to create these Master - Subordinate paths.

## Starting a session

Click the **Logical Subsystem** button to display the LSSs. Choose an LSS in the ESS you want to be the Master.

> **Important:** Once you select an LSS to be used as communication to the Master, then you should use the same LSS for all subsequent communications. While it is possible to use other LSSs in the same ESS, the results may be unpredictable.

Left-click to highlight it, then right-click to launch the wizard as shown in Figure 7-49.

> **Tip:** We recommend you use the first LSS in the ESS as a convention to avoid confusion.

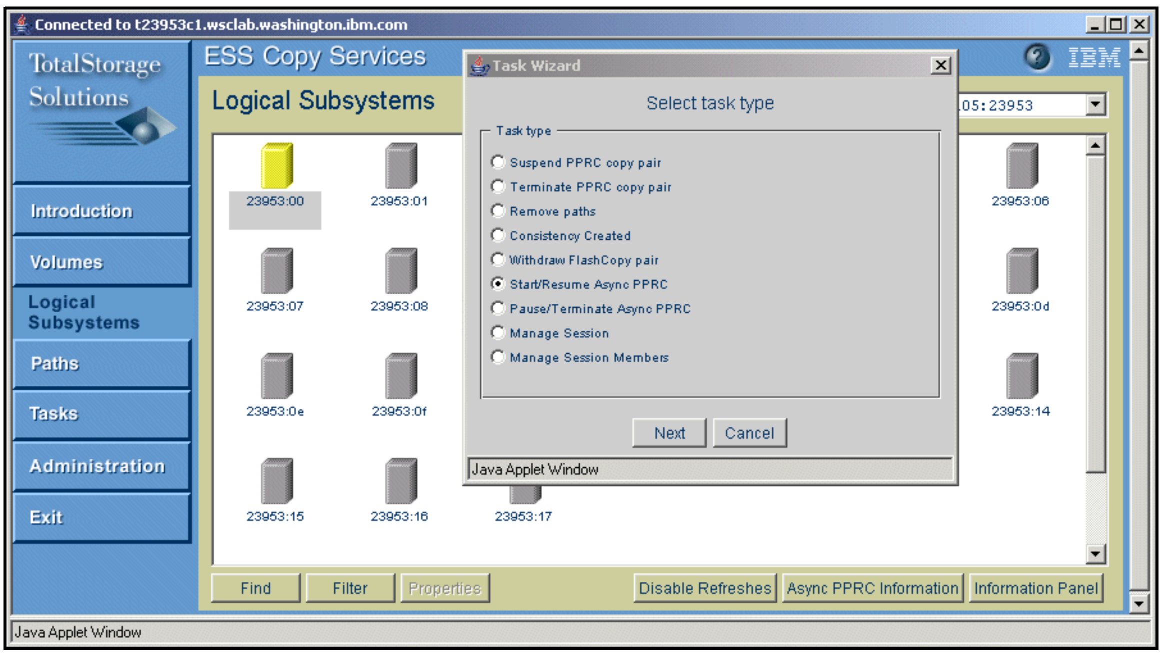



*Figure 7-49 Starting the Asynchronous PPRC session: Selecting an LSS*

> **Note:** Once a session is terminated, then it can be restarted on any other LSS.

Select **Start/Resume Asynchronous PPRC** and click **Next**. This will bring up the options wizard as shown in Figure 7-50. Set the timers to your desired values. At the time of writing the default options are 0, 50 and 30 respectively, these may change. It is recommended that you only change the Consistency Group Interval Timer value and leave the other 2 as the default unless you have a specific reason to change them. The meaning of these 3 timers is as follows:

- **Consistency Group Interval Timer (in seconds):** This is the time from when the last Consistency Group was formed (or when the formation timed out) to when the next one will be started. As the Consistency Groups typically form in about 1-3 seconds then combining these 2 values will give the average time between group formations. A value of 0 will begin forming the next group as soon as the previous one completes.

- **Maximum Coordination Timer (in milliseconds):** This is the maximum time allowed for the formation of consistency across all the volumes at the primary site. If this timer times out, the formation is cancelled.
- **Maximum Consistency Group Drain Time (in seconds):** This is the time from when the point-in-time consistency is formed at the primary site, until the FlashCopies have been started at the remote site; the time to complete the PPRC-XD drain on all volumes.

Select the session id that this Asynchronous PPRC Master will be managing. Click **Next**.

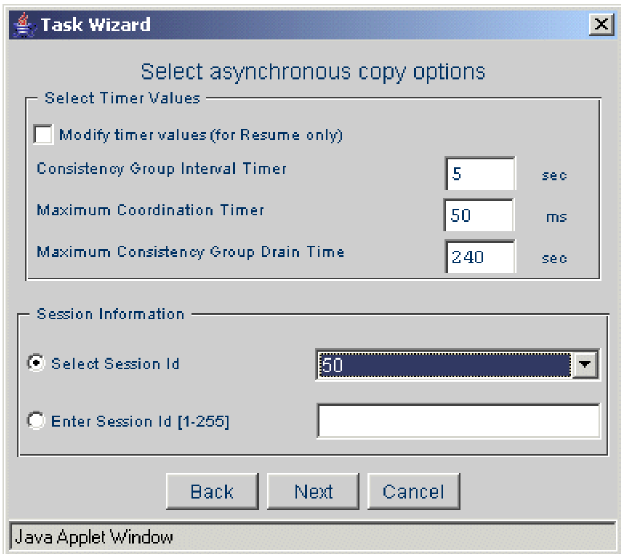



*Figure 7-50 Starting Asynchronous PPRC session, selecting session and values*

The next window allows you to select the Master - Subordinate relationships. These should be ESSs which have LSSs with sessions defined with the same id, or will have in the future. Remember that once the session is started, it must be stopped to change these relationships.

PPRC paths must already be set up as described above before a Subordinate can be selected. Use the pull-downs to create a Master - Subordinate relationship for each Subordinate ESS, then click **Next**.

**Restriction:** Subordinates cannot be added or removed once the session is started. The session must be terminated and restarted to change the Master to Subordinate relationships.

You will once again get the task save wizard, save this task if you wish. Click **Run**.

This will return you to the LSS display as shown in Figure 7-51. Note that the LSS now has a different icon. The color and shading still has the same meaning, but the icon has a small top on it, to indicate it is the Master LSS.

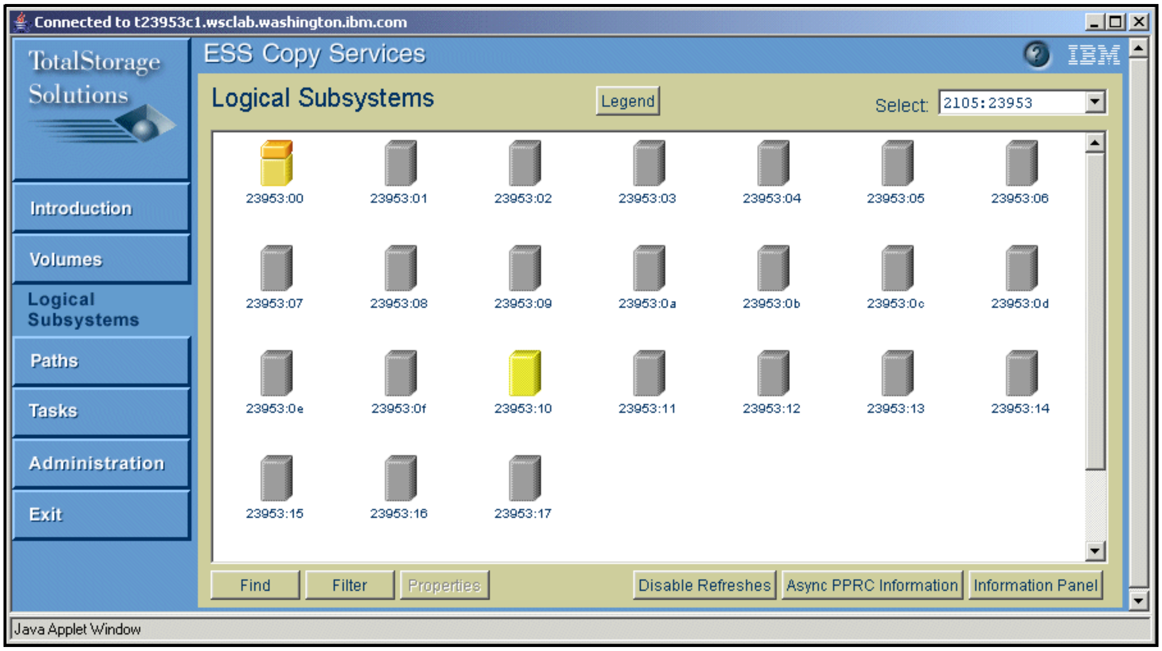



*Figure 7-51 Starting Asynchronous PPRC session: Viewing results*

We can now display the volumes by clicking the **Volumes** button. As can be seen in Figure 7-52, the session icon has changed to session member as seen by the double arrow under the 3 small volumes. If this process was done quickly, there may still be some delay for each volume to join due to the PPRC-XD pairs still draining. As they are ready and have formed their first consistent FlashCopy at the remote site, the primary volumes will show as joined.

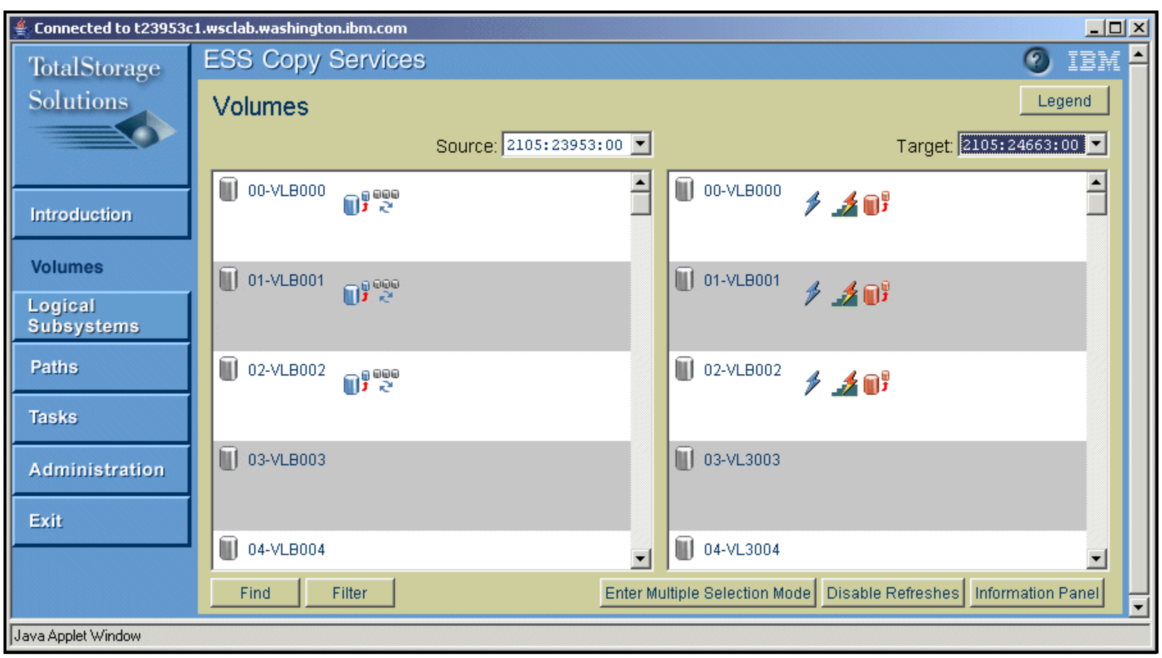



*Figure 7-52 Starting Asynchronous PPRC session, viewing volume status*

For the initial start of the session, the second and subsequent Consistency Groups will not be started until all volumes have completed joining and formed their first consistent FlashCopy.

**Note:** Once the session has all volumes joined and is forming regular consistent FlashCopies, subsequent additions of volumes will not immediately join, but remain *join pending*, until they are able to form consistent FlashCopies. The existing volumes in the session will not be interrupted.

The status of the Asynchronous PPRC session can be displayed from the LSS view. Click the **Logical Subsystems** button, then click the **Async PPRC Information** button at the bottom. A window, as shown in Figure 7-53 will be displayed. This window can be left open for monitoring purposes.

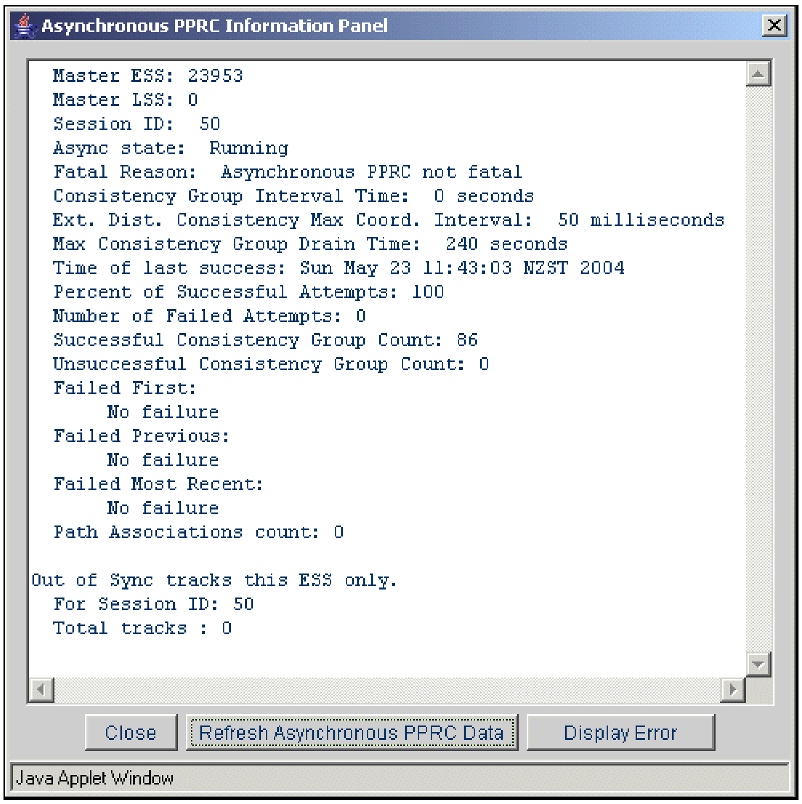



*Figure 7-53 Starting Asynchronous PPRC session: Viewing status panel*

Note the following information:

- The Master ESS serial number, the LSS it was started on and the session id that it is using are displayed.
- The Async Status.
- If there has been a fatal error, the reason is displayed, otherwise `Asynchronous PPRC not fatal`.
- The settings of the three timers are also displayed.
- The time stamp of the last successful Consistency Group and also a percentage of how many have been successful, followed by the actual number of failures.
- How many have been formed since the session was started.
- Three error logs, the first error, and the last two are displayed.
- Path associations and a display of the number of out-of-sync tracks in the session, *but for this ESS only*.

A

# ESS Application Programming Interface (API)

In this appendix, the following topics are discussed:

- The ESS Application Programming Interface (API)
- ESS API components
- The Client application
- Installing the ESS API
- Prerequisites

# The ESS Application Programming Interface (API)

This section introduces the IBM TotalStorage Enterprise Storage Server (ESS) Application Programming Interface (API). The ESS API is to be used in conjunction with a client application.

The terms ESS API and ESS CIM Agent mean the same thing and are used synonymously.

For a more complete understanding and discussion of CIM please refer to *IBM Tivoli Storage Resource Manager: A Practical Introduction,* SG24-6886.

## ESS API characteristics

The use of the ESS API demonstrates IBM's commitment to open standard interfaces. The ESS API is a non-proprietary storage management agent that supports routine LUN management activities, such as LUN creation, mapping, and masking. The ESS API also enables ESS Copy Services configuration and use activities, such as FlashCopy and PPRC management. It supports these activities through the use of the Storage Management Initiative Specification (SMIS), as defined by the Storage Networking Industry Association (SNIA).

The ESS API helps integrate ESS configuration management support into storage resource management (SRM) applications, which allow users to benefit from existing SRM applications and infrastructures. The ESS API also enables the automation of configuration management through customer-written applications.

Common Information Model (CIM) is a management standard published by the Distributed Management Task Force (DMTF). It defines a common set of classes, associations, and object relationships for managing system components (including storage).

The following list provides an overview of the ESS API:

- Provides the means for client applications to have common access for data management applications, such as Tivoli SRM, HP OpenView, and VERITAS.
- ESS CIM Agent 1.2 implements CIM Schema 2.8 for Storage Management.
- Middleware: CIM API only, no user interface.
  - Translates a proprietary device interface to a CIM-compliant interface.
  - Enables resource management applications for ESS devices.
- Each device has its own CIM Agent associated with it.
  - ESS CIM Agent is not a stand-alone product on its own release schedule.
  - It is synchronized with the device schedule.
- Provides management of the ESS.
  - Simplifies ESS configuration and ESS Copy Services administration.
- Allows third-party software to manage the ESS using SNIA CIM protocols.
- ESS LIC level 2.1.0 provides API ESS CLI functionality.
- ESS LIC level 2.3.0 provides API support for the SMI-S Storage Virtualization Initiative (SVI) interfaces.
- ESS LIC level 2.4.0 provides API support for Asynchronous PPRC and two FlashCopy withdraw options, Commit and Revert.
- Provides a standards-based programmatic interface to the ESS.

## ESS API components

A CIM agent consists of the components shown in Figure A-1. The main components are the CIM object manager (CIM/OM), the service location protocol (SLP), and the device provider. The SLP is a directory service that a client application calls to locate the CIM Object Manager. A device can be a storage server, such as the ESS.

The CIM/OM and device provider communicate through method calls made from the CIM/OM to the provider. The device provider communicates with the device through proprietary calls.

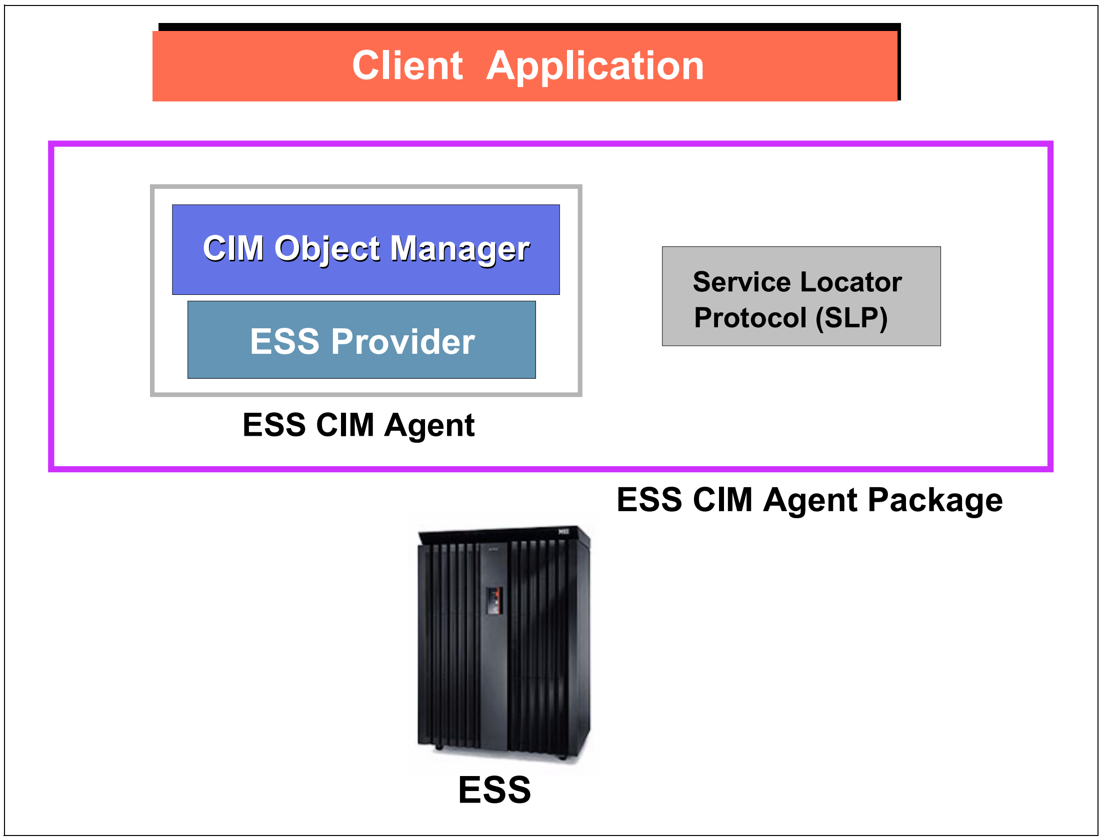



*Figure A-1 Components of the API*

The specific ESS API method of operation is shown in Figure A-2 on page 354. The following list describes the method of operation:

- The ESS CIM Agent registers itself with Service Locator Protocol to enable discovery by the Client application.
- A client discovers the ESS CIM Agent by calling the Service Locator Protocol service (or else it has to have built-in knowledge of the agent's network location).
- A client application makes calls to the ESS CIM Agent.
- The client application and the CIM/OM communicate through CIM Messages.
- The CIM Object Manager calls ESS Provider.
- The ESS CIM Agent (Provider) makes calls to an ESS.

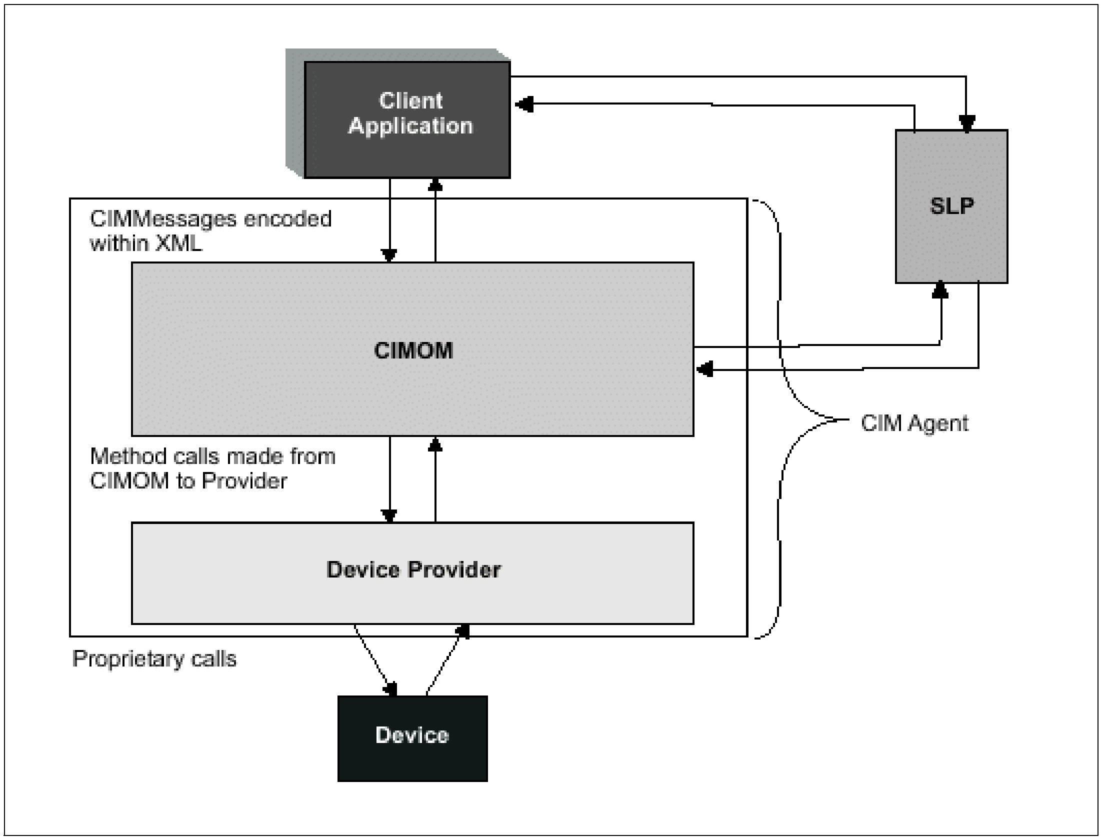


*Figure A-2 API method of operation*

The ESS CIM Agent subcomponents are not exposed to the client. The client application performs requests to the CIM Agent.

### Summary

The following list is a summary of what the ESS API does:

- ESS API (ESS CIM Agent) allows third-party software to manage the ESS using SNIA CIM protocols.
- ESS CIM Agent provides an abstraction of control of ESS operations (including ESS Copy Services) to the data management client applications and does not explicitly expose vendor unique functionality.
- ESS CIM Agent implements a standards-based programmatic interface to the ESS.

## The ESS API interface

The ESS API presents another option for ESS management by complementing the use of the ESS Web-based user interfaces (ESS Copy Services Web User Interface and ESS Specialist), the ESS command line based interfaces (ESS Copy Services CLI and ESS CLI), as well as the z/OS interfaces (TSO commands, ICKDSF commands, and ANTRQST API), as Figure A-3 on page 355 illustrates.

It provides management of the ESS by simplifying ESS configuration and Copy Services administration.

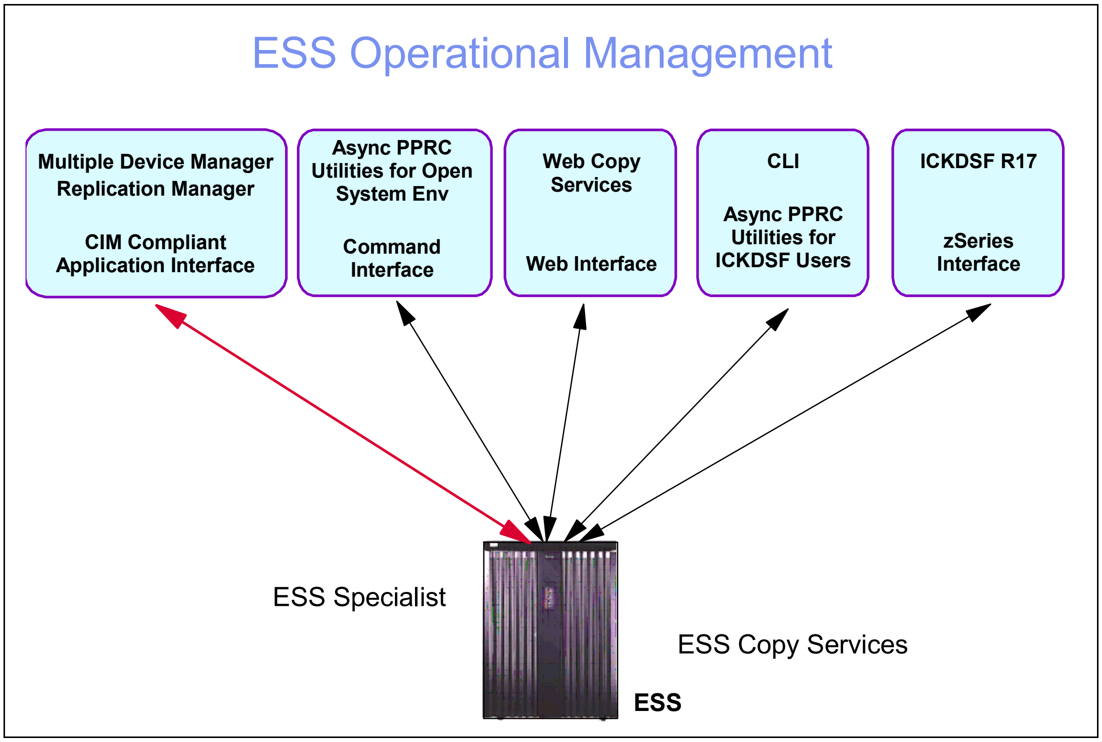



*Figure A-3 ESS management interfaces*

## The client application

The client application can be an application for business continuance or storage administration. The following things can be done by a client application using the ESS CIM Agent services.

ESS 2.1.0 (CIM Agent 1.1) provided these functions:

- Create host connections
- Create LUNs
- Connect and disconnect LUNs from hosts
- Discover all hosts known to the ESS
- Discover the storage configuration of an ESS
- Physical disks
- Logical volumes

ESS 2.3.0 (CIM Agent 1.2) provided all of the ESS 2.1.0 functions plus:

- Create VolumeSpace
- Delete VolumeSpace
- Create CKD Volume
- FlashCopy
- PPRC

ESS 2.4.0 (CIM Agent 1.2.1) provides all of the ESS 2.3.0 functions plus:

- Manage sessions
- Manage Asynchronous PPRC relationships
- Query Asynchronous PPRC status
- FlashCopy Commit and Revert

Using the ESS Web User Interface Copy Services, tasks are created and saved and later can be executed by the ESS Copy Services CLI using commands such as `rsExecuteTask.sh`. The ESS CIM API can enable a client application to do some of the same type of ESS data management functions that a saved task might do.

# Installing the ESS API

The ESS API CIM Agent and CLI are provided with the ESS LIC at no additional charge. The CIM Agent is available to run on AIX, Linux, and Windows 2000 or later operating system environments.

Please refer to the interoperability matrix for the most current information at:

http://www.storage.ibm.com/disk/ess/supserver.htm

Please refer to publication *IBM TotalStorage Enterprise Storage Server Application Programming Interface Reference*, GC35-0489, for instructions on installing the ESS API and CLI on your operating system.

## Prerequisites

In order to install the ESS API in your environment, consider the prerequisites listed in “Hardware” on page 356 and “Software” on page 357. You can contact your IBM representative or IBM Business Partner for help with a MDM and CIM Agent sizing tool to size your environment.

## Hardware

Ensure that your system satisfies the prerequisites for installing the ESS CIM Agent on a Windows 2000 or later, AIX, or Linux operating system before starting the installation.

The following hardware is required:

- Personal computer, workstation, or server with Intel Pentium III or higher processor
- CD-ROM drive
- Video graphics adapter display or better

The following space on your Intel workstation is required:

- 1 GHz Pentium 3 or Pentium 4 processor (2 GHz is better, Pentium 4 recommended.)
- 256 MB RAM minimum (recommend 2GB), depending on your system configuration
- 1GB of disk space
- Up to 50 MB of temporary disk space for installation purposes

The following space on your AIX server is required:

- 1 GHz processor
- 1 GB RAM
- 4 GB disk

**Note:** You might need to increase the total available disk space on your hard drives if the ESS CIM Agent and other associated products are split between more than one logical drive. Also, the ESS CIM Agent might require additional memory to operate if you configure it to manage many devices or devices with large configurations.

For the ESS, the following considerations apply:

- LIC level must be 2.4.0 or later.
- Compatibility with existing ESS Model 750, 800 or Fxx:
  - CIM Agent 1.2.1 supports ESS models F10, F20, 750 and 800 for all non-Copy Services functions
  - Copy Services is supported on Model 750 and 800 machines running the ESS LIC 2.3.0 microcode or higher.
- For the complete interoperability matrix information see:

  http://www.storage.ibm.com/disk/ess/supserver.html

## Software

All prerequisite software must be installed before you start the installation of the ESS CIM Agent. The ESS CIM Agent installation program will check for the existence of the ESS CLI. If the program does not detect the ESS CLI, the installation of the ESS CIM Agent will not complete successfully.

The following software is required:

- Operating systems:
  - Windows 2000 or later
  - AIX 5L Version 5.1 or later
  - Linux Red Hat 7.2

  **Note:** For the latest versions of Red Hat compatibility with the ESS API, refer to:

  http://www.storage.ibm.com/disk/ess/supserver.html

- The workstation/computer/server must be IP connected to ESS(s) and Storage Management Client.
- ESS CLI level 2.4.0.236 or later. This software is on the ESS CLI CD.
- Common Information Model (CIM) Agent. This software is on the CIM Agent for ESS CD.
- Transmission Control Protocol/Internet Protocol (TCP/IP).
- Adobe Acrobat Reader version 4.0 or later.

  You need the Adobe Acrobat Reader to read the License Agreement and product information from the ESS CIM Agent LaunchPad. You can download the Adobe Acrobat Reader from the following Web site:

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

- Vendor storage management products, such as Tivoli SRM.

B

# Geographically Dispersed Parallel Sysplex (GDPS)

This appendix presents the Geographically Dispersed Parallel Sysplex (GDPS) offering. GDPS provides an automated disaster recovery solution for total business continuity in the OS/390 and z/OS environments.

For information on the IBM Installation Services for Geographically Dispersed Parallel Sysplex, refer to the following Web URLs:

```
http://www.ibm.com/servers/eserver/zseries/announce/april2002/gdps.html
http://www.ibm.com/services/its/us/mus62b1.html
```

For additional information on GDPS, contact your IBM representative or e-mail:

```
gdps@us.ibm.com
```

Whitepapers on GDPS include:

- Geographically Dispersed Parallel Sysplex: The Ultimate e-business Availability Solution, GF22-5114.
- Geographically Dispersed Parallel Sysplex: the S/390 Multi-site Application Availability Solution, GF225063.

The whitepapers can be found at:

```
http://www.ibm.com/servers/storage/support
```

# B.1 GDPS overview

GDPS is a multi-site application availability solution that provides the capability to manage the remote copy configuration, automates Parallel Sysplex operational tasks, and performs failure recovery from a single point of control, thereby drastically improving application availability.

GDPS supports all transaction managers (for example, CICS, IMS, WebSphere) and database managers (for example, DB2, IMS, and VSAM), and is enabled by means of key IBM technologies and architectures:

- Parallel Sysplex
- Tivoli Netview for OS/390
- System Automation for OS/390
- IBM TotalStorage Enterprise Storage Server (ESS)
- IBM TotalStorage Peer-to-Peer Virtual Tape Server (PtP VTS)
- Optical Dense Wavelength Division Multiplexer (DWDM)
- Peer-to-Peer Remote Copy (PPRC) architecture
- Extended Remote Copy (XRC) architecture
- Virtual Tape Server Remote Copy architecture

GDPS supports both the synchronous (PPRC) as well as the asynchronous (XRC) forms of remote copy. In addition, it supports the Peer-to-Peer Virtual Tape Server (PtP VTS) form of remote copying tape data.

GDPS consists of production systems, controlling systems, and automation.

## B.1.1 Production systems

The production systems execute the mission critical workload. There must be sufficient processing resource capacity, such as processor capacity, main storage, and channel paths available, which can quickly be brought on-line to restart a system's or site's critical workload. This is typically done by terminating one or more systems executing expendable (non-critical) work and acquiring its processing resource.

The *Capacity Backup (CBU)* feature, available on the IBM 9672, and in the IBM zSeries, is designed to provide additional processing power, which can help you to achieve significant cost savings. The CBU feature has the ability to increment capacity temporarily, when capacity is lost elsewhere in the enterprise. CBU adds central processors (CPs) to the available pool of processors and is activated only in an emergency.

GDPS-CBU management automates the process of dynamically adding reserved central processors, thereby minimizing manual customer intervention and the potential for errors. The outage time for critical workloads can be reduced from hours to minutes.

## B.1.2 Controlling system

The controlling system coordinates GDPS processing. All GDPS functions are initiated and coordinated by the controlling system.

## B.1.3 Automation

All GDPS systems are running GDPS automation based upon Tivoli NetView® for OS/390 and System Automation for OS/390. Each system can monitor the Sysplex cluster, Coupling Facilities, and storage subsystems, and maintain GDPS status. GDPS automation can coexist with an enterprise’s existing automation product.

## B.2 GDPS/PPRC

GDPS/PPRC manages and protects IT services by handling planned and unplanned exception conditions, and maintaining full data integrity across multiple volumes and storage subsystems. By managing both planned and unplanned exception conditions, GDPS/PPRC maximizes application availability and provides business continuity.

### B.2.1 Topology

GDPS/PPRC is designed with the following characteristics in mind:

- Continuous Availability solution
- Near transparent disaster recovery solution
- Recovery Time Objective (RTO) less than an hour
- Recovery Point Objective (RPO) of zero (optional)
- Protects against metropolitan area disasters

Figure B-1 shows a simplified illustration of the physical topology of a GDPS/PPRC implementation. GDPS/PPRC, consists of a base or Parallel Sysplex cluster spread across two sites (site 1 and site 2 in Figure B-1) separated by up to 40 kilometers of fiber with one or more z/OS or OS/390 systems at each site. The multi-site Parallel Sysplex cluster must be configured with redundant hardware, for example, a Coupling Facility, and Sysplex Timer in each site, as well as alternate cross-site connections.

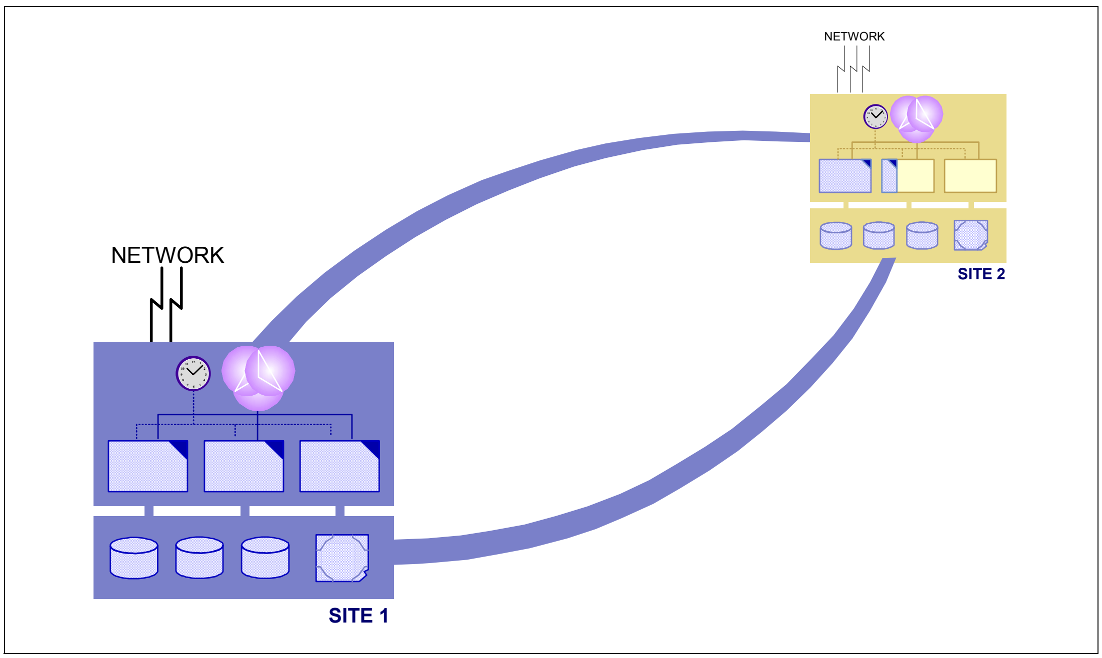



*Figure B-1 GDPS/PPRC - basic topology*

All critical data residing on disk storage subsystems in site 1 (the primary copy of data) is mirrored to the disk storage subsystem in site 2 (the secondary copy of data) through PPRC synchronous remote copy.

GDPS/PPRC provides the Parallel Sysplex cluster continuous availability benefits, and it significantly enhances the capability of an enterprise to recover from disasters and other failures, as well as managing planned actions.

### B.2.2 Planned re-configuration support

GDPS/PPRC planned re-configuration support automates procedures performed by an operations center. These include standard actions to:

- Quiesce a system’s workload and remove the system from the Parallel Sysplex cluster (for example, stop the system prior to a change window)
- IPL a system (for example, start the system after a change window)
- Quiesce a system's workload, remove the system from the Parallel Sysplex cluster, and re-IPL the system (for example, recycle a system to pick up software maintenance)

Standard actions can be initiated against a single system or a group of systems. Additionally, GDPS/PPRC provides customizable scripting capabilities for user defined actions (for example, planned site switch in which the workload is switched from processors in site 1 to processors in site 2).

### B.2.3 Unplanned re-configuration support

GDPS/PPRC unplanned re-configuration support not only automates procedures to handle site failures, but is designed to also minimize the impact and potentially mask a z/OS (or OS/390) system, processor, Coupling Facility, disk subsystem or tape failure, depending on the GDPS/PPRC policies installed. Some examples of unplanned failures that can be managed by GDPS/PPRC are:

- In the event of a z/OS system failure, the failed system and workload can be automatically restarted.
- In the event of processor failure, the failed system(s) and the workload can be restarted on other processors.
- In the event of primary disk storage subsystem failure, a disk re-configuration will allow access to the secondary PPRC volumes, which will contain mirrored data consistent with the primary data.

### B.2.4 GDPS/PPRC HyperSwap

The GDPS/PPRC HyperSwap function is designed to broaden the continuous availability attributes of GDPS/PPRC by extending the Parallel Sysplex redundancy to disk subsystems. The HyperSwap function when managed completely through automation, can significantly decrease the time it takes to complete activities such as switching sites and switching disk subsystems between sites.

Basically, the HyperSwap function can be used to mask some primary PPRC disk subsystem problems or planned disk subsystem maintenance activities, by allowing the primary DASD to be swapped transparently from one site to another without requiring an application or system outage.

## B.3 GDPS/XRC

In addition to providing the automation for unplanned re-configuration support, the primary attribute of most disaster recovery solutions, GDPS/XRC has many other functions such as the planned re-configuration support that is discussed in the current section.

## B.3.1 Topology

GDPS/XRC is designed with following characteristics in mind:

- Disaster recovery solution
- RTO between an hour to two hours
- RPO less than two minutes
- Protects against metropolitan as well as regional disasters (unlimited distance)
- Minimal remote copy performance impact

GDPS/XRC consists of one or more production systems in site 1, where the production system can be a single system, multiple systems sharing disk subsystems, or a base or Parallel Sysplex cluster. Figure B-2 shows a simplified illustration of the physical topology of a GDPS/XRC implementation.

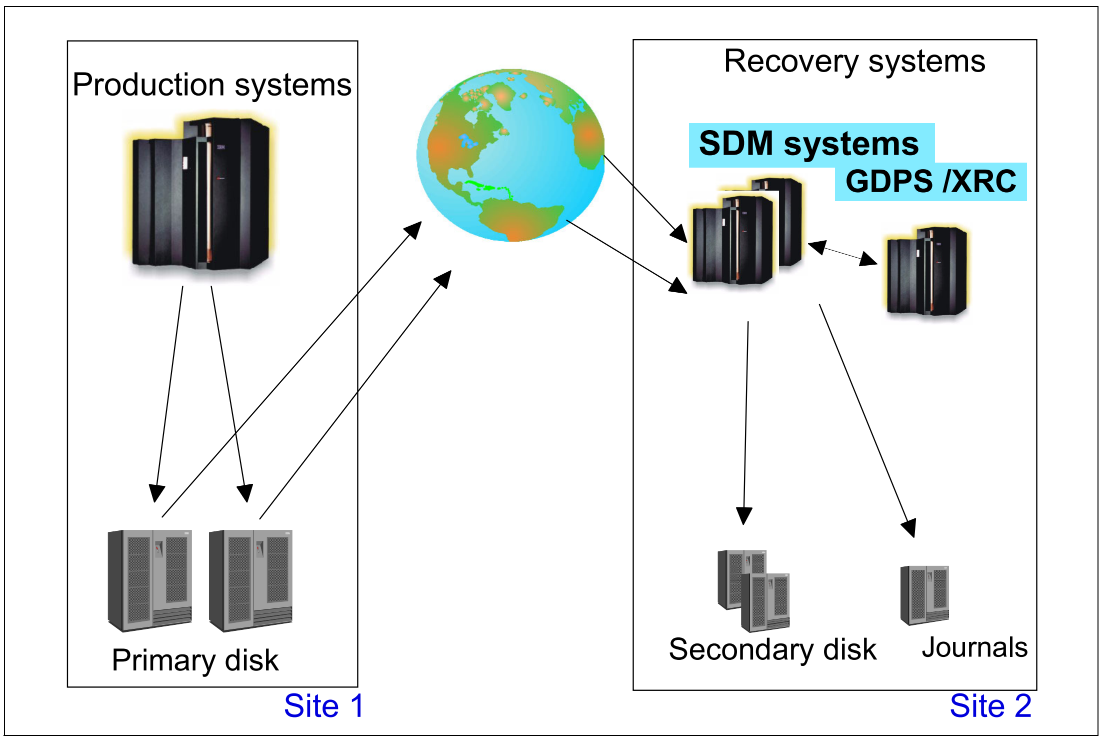



*Figure B-2 GDPS/XRC - topology*

Note that the Parallel Sysplex cluster does not span across sites 1 and 2. Site 2, (the recovery site) can be located at virtually any distance from site 1 (the production site). During normal operations, the XRC System Data Mover (one or more) is executed in site 2, and is in a Base Sysplex with the GDPS controlling system.

All critical data resides on storage subsystems in site 1 (the primary copy of data) and is mirrored to the storage subsystems in site 2 (the secondary copy of data) through XRC asynchronous remote copy.

## B.3.2 Planned re-configuration support

All planned re-configuration actions described in B.2.2, “Planned re-configuration support” on page 362, are provided by GDPS/XRC for the System Data Mover (SDM) Sysplex in the recovery site. For example, GDPS/XRC manages the temporary relocation of the SDM, if it is needed. By managing the SDM Sysplex, GDPS/XRC can also manage the XRC remote copy configuration Configuration Management support.

## B.3.3 Unplanned re-configuration support

It is highly recommended that the disaster recovery solution be completely automated. GDPS/XRC automates the process of recovering the production environment with minimal manual intervention, thus providing significant value in minimizing the duration of the recovery window.

## B.3.4 Coupled System Data Mover support

A single SDM can typically manage approximately 1000 to 2000 volume pairs depending on the write I/O rate. Coupled Extended Remote Copy (CXRC) expands the capability of XRC, by ensuring configurations with many thousands of primary volumes using more than one SDM, and enabling recovery of all the volumes in the configuration to a consistent point-in-time. CXRC provides the scalability that is required to support larger XRC configurations.

# B.4 Functional highlights

Also, the following functions are supported by both GDPS/PPRC and GDPS/XRC:

- Peer to Peer Virtual Tape Server (PtP VTS)
- FlashCopy
- Management of zSeries Operating Systems

### Peer to Peer Virtual Tape Server (PtP VTS) support

GDPS supports Peer to Peer Virtual Tape Server. By extending GDPS support to data resident on tape, the GDPS solution provides continuous availability and near transparent business continuity benefit for both disk subsystems and tape resident data.

Enterprises are no longer forced to develop and use processes that create duplex tapes and maintain the tape copies in alternate sites. For example, previous techniques created two copies of each DBMS (Database Management System) image copy and archived log as part of the batch process and manual transportation of each set of tapes to different locations.

Operational data, data that is used directly by applications supporting end users, is normally found on disk subsystems. However, there is another category of data that supports the operational data, which is typically found on tape subsystems. Support data typically covers migrated data, point-in-time backups, and archive data. For sustained operation in the failover site, the support data is required. Furthermore, several enterprises have mission critical data that only resides on tape.

The PtP VTS provides a hardware based tape duplexing solution. In the event of a planned site switch or a site failure, GDPS will automatically manage the duplexed tapes. GDPS will FREEZE copy operations, so that tape data consistency can be maintained across GDPS managed sites during a switch between the primary and secondary VTSs.

### FlashCopy support

FlashCopy, available on the IBM TotalStorage Enterprise Storage Server (ESS), provides a point-in-time copy of data in a minimum amount of time. FlashCopy enables you to copy or dump data while applications are updating the data.

FlashCopy is automatically invoked before re-synchronization commences (based upon the policy activated), whenever a re-synchronization request is received. This function ensures that there is always a consistent image of data to fall back to, in the event that a disaster occurs, while re-synchronization is in progress.

### Management of zSeries operating systems

In addition to managing images within the base or Parallel Sysplex cluster, GDPS can also manage other production operating systems. These include z/OS, Linux for zSeries, z/VM, and VSE/ESA.

The operating systems have to run on servers that are connected to the same Hardware Management Console (HMC) local area network (LAN) as the Parallel Sysplex cluster images. For example, if the volumes associated with the Linux images are mirrored using PPRC, GDPS can restart these images as part of a planned or unplanned site re-configuration. The Linux for zSeries images can either run as a logical partition (LPAR) or as a guest under z/VM.

## B.5 IBM Global Services (IGS) Offerings

GDPS is available through IBM Global Services only. The following GDPS services, for both PPRC and XRC, are offered:

- Technical Consulting Workshop (TCW)
- Remote Copy Management Facility (RCMF)
- Geographically Dispersed Parallel Sysplex (GDPS)

### Technical Consulting Workshop (TCW)

Typically, the TCW is a two day workshop that helps to determine whether GDPS is the right solution for your business. The objective is that IGS specialists work with customer representatives to understand the business objectives, service requirements, technological directions, business applications, recovery processes, cross-site and I/O requirements. High-level education on GDPS is typically provided, along with the service and implementation process. Various remote and local data protection options are evaluated.

Usually, IGS specialists will present a number of planned and unplanned GDPS re-configuration scenarios, with recommendations on how GDPS can assist you in achieving your objectives.

### Remote Copy Management Facility (RCMF)

With this service, the RCMF automation to manage the remote copy infrastructure is installed, the automation policy customized, and the automation verified along with providing operational education for the enterprise.

### Geographically Dispersed Parallel Sysplex (GDPS)

Typically, this service will consist of the installation of the GDPS automation to manage the remote copy infrastructure, perform routine tasks, and recover from failures, with the automation policy customized, and the automation verified along with providing operational education for the enterprise.

C

# Migrating from RVA to ESS

This appendix discusses the considerations for users of the IBM RAMAC Virtual Array (RVA) SnapShot function who are migrating to the IBM TotalStorage Enterprise Storage Server (ESS).

The material presented in this appendix is based on a paper written by Yufen Davis, Tony Pearson, and John Thompson from IBM.

# C.1 Introduction

The IBM RAMAC Virtual Array (RVA) and the IBM TotalStorage Enterprise Storage Server (ESS) are disk storage subsystems available for zSeries environments. Installations with RVA disks have been able to address their storage requirements by exploiting volume-level and data set level SnapShot capabilities. Although ESS does not match the exact functions or interface syntax of RVA, it provides, in many cases, alternative solutions for the same problems.

This appendix describes the similarities and differences between RVA and ESS, then presents a set of typical scenarios when SnapShot was used in the past with RVA, and how the Data Facility Storage Management Subsystem (DFSMS) feature DFSMSdss can be used instead with ESS.

The examples in this appendix are focused specifically on handling z/OS data sets using DFSMSdss (for DFSMS release 10 and above), employing ESS FlashCopy Version 2 functions available with LIC 2.2.0 such as data set FlashCopy.

# C.2 IBM RAMAC Virtual Array (RVA)

The RVA products do not update data in place. This is a radical departure from traditional disk storage architectures and enables the use of unique and powerful storage subsystem functions. The architecture implemented by the RVA is called *log structured array*.

## C.2.1 SnapShot

IBM RAMAC SnapShot is a virtual data duplicator that enables you to make an almost immediate copy of data by copying the logical pointers to the data. As no data movement takes place, rather new pointers are created when there are new updates written, it is a very fast process and uses no additional disk space to make a copy. Snapped data can be used to enhance critical business applications, data mining, Disaster Recovery, and application development and testing. Greater efficiency is achieved from data that is current and easily refreshed.

SnapShot is available only on the RVA, and requires both a hardware (microcode) feature on the RVA and the IXFP software. SnapShot enables you to make copies of disk volumes, data sets, or VM minidisks. You can make as many copies of a SnapShot source or target as you like. However, both source and target must reside on the same RVA disk subsystem.

## C.2.2 Virtual Concurrent Copy

Virtual Concurrent Copy is another method of making copies of data on the RVA. It is invoked by specifying the `CONCURRENT` keyword on a DFSMSdss `COPY` or `DUMP` command, which makes a “logical” copy of the data's tracks to Working Space Data Sets (WSDS) using SnapShot. Once the logical copy is complete, which takes about the same time as a data set SnapShot, serialization is released and processing of the source data set can continue.

DFSMSdss then physically moves the data to the target data set or volume, which can be allocated on either disk or tape. Once the data has been physically copied to the target data set or volume, it is available for reads or updates.

Virtual Concurrent Copy provides a function compatible with Concurrent Copy on the ESS, and is supported by IMS, DB2, and CICS utilities.

## C.2.3 Peer-to-Peer Remote Copy

The RAMAC Virtual Array supports Peer-to-Peer Remote Copy, but only to other RVA disk subsystems. With PPRC, volumes are mirrored. The user identifies which volumes on the primary system are paired to which volumes on the secondary system. PPRC does not allow RVA to ESS mirroring in either direction.

SnapShot, Virtual Concurrent Copy, and PPRC function only on the disk subsystem contents. Any updates buffered in processor memory are not included. It is up to the user to quiesce the application to ensure that it has written the data to the disk subsystem in a manner that copies are usable without performing restart activities such as the backout of inflight transactions. If a quiesce is not performed, a *power outage* consistent copy of the data is created and restart actions will restore the usability of the data.

# C.3 IBM TotalStorage Enterprise Storage Server

The IBM TotalStorage Enterprise Storage Server (ESS) does not employ a virtual disk architecture. Instead, features like FlashCopy and Concurrent Copy are implemented using the Home Area architecture found in most other traditional disk storage subsystems. The ESS has *logical subsystems* (LSS). From a z/OS perspective, the LSS are seen as logical control units (LCU). There can be up to 256 addresses in an LSS.

## C.3.1 Concurrent Copy

ESS supports both volume and data set Concurrent Copy (CC). It is invoked by specifying the **CONCURRENT** keyword on a DFSMSdss **COPY** or **DUMP** statement, which operates by creating an instant record of all the data to be copied. Once this is logically complete, which takes about the same time as a data set SnapShot, serialization is released and processing of the source data set can continue.

DFSMSdss then physically moves the data to the target data set or volume. Once this physical movement is complete, the data is available for reads or updates. Sidefiles in cache and system storage are used to maintain the source data image.

Concurrent Copy is supported by IMS, DB2 and CICS utilities. When a full volume copy is requested with Concurrent Copy, the user must wait until physical complete before the target volume can be copied-back to the source in a recovery situation.

Refer to Chapter 6, “Concurrent Copy” on page 273 for details on Concurrent Copy.

## C.3.2 FlashCopy

FlashCopy is a licensed feature on ESS and enables the user to make a fast data copy. FlashCopy operates by creating an instant record of all the data to be copied, and making the source and target copies immediately available for an application update.

Data movement can optionally take place asynchronously as a background task. FlashCopy enables you to make copies of Count Key Data (CKD) or Enhanced Count Key Data (ECKD™) disk volumes in a zSeries environment. When a copy is requested with FlashCopy, the user must wait until the background copy within the ESS is complete before the target volume can be copied back to the source volume in a recovery situation.

FlashCopy Version 2 is an optional feature to the ESS Model 800, F20, and F10. Among the new functions and enhancements provided in FlashCopy V2 are data set FlashCopy, multiple relationship FlashCopy, and elimination of the LSS constraints. In contrast to volume FlashCopy (FlashCopy V1), data set FlashCopy operates on sets of tracks instead of the entire volume, which enables a user to perform a FlashCopy of a data set using z/OS DFSMSdss.

Also with FlashCopy V2, a source can have FlashCopy relationships with multiple targets simultaneously. You can initiate up to 12 FlashCopy relationships on a given volume, track, or data set, without the need to first wait for or cause previous relationships to end. A FlashCopy target can only have a relationship with a single source at any given time. A FlashCopy source and target relationship can span logical subsystems within an ESS.

Refer to Chapter 2, “FlashCopy” on page 13 for details on FlashCopy.

### C.3.3 Peer-to-Peer Remote Copy

The ESS supports Peer-to-Peer Remote Copy (PPRC). With PPRC volumes are mirrored. The user identifies which volumes on the primary system are paired to which volumes on the secondary system. The primary volumes cannot be FlashCopy target volumes, unless the PPRC sessions are stopped and the pairs fully reestablished. PPRC does not allow RVA to ESS mirroring in either direction.

Refer to Chapter 3, “Peer-to-Peer Remote Copy” on page 69 for details on PPRC.

### C.3.4 XRC

The IBM TotalStorage Enterprise Storage Server also supports Extended Remote Copy (XRC). An XRC session identifies all the primary volumes in one system, and mirrors these to secondary volumes attached to a second system. The XRC function is not limited to ESCON distances, and can use communications protocols to allow for greater distances between primary and secondary copies since it transfers asynchronously to the remote site. Refer to Chapter 5, “Extended Remote Copy” on page 207 for detailed information on XRC.

Concurrent Copy, FlashCopy, PPRC and XRC function only on the disk subsystem contents. Any updates buffered in processor memory are not included. It is up to the user to quiesce the application to ensure that it has written the data to the disk subsystem in a manner that copies are usable without performing restart activities such as the backout of inflight transactions. If a quiesce is not performed, a “power outage consistent” copy of the data is created and restart actions will restore the usability of the data.

## C.4 IXFP

The IXFP program, SIBBATCH, is used to invoke SnapShot directly from batch jobs. It can issue a SnapShot against a single volume, or a single data set. If SnapShot cannot be used for the operation, and the DATAMOVERNAME(DSS) keyword is specified, SIBBATCH will invoke DFSMSdss to perform the data movement. In this case, if the source and target are in the same ESS, DFSMSdss will automatically attempt to invoke FlashCopy. This facilitates an easy migration from RVA to ESS for SIBBATCH users.

IKJEFT01 is the TSO batch monitor program. It can be used to invoke IXFP TSO commands, such as SIBADMIN. These can be incorporated into clists or REXX execs.

## C.5 DFSMSdss

The DFSMSdss program, ADRDSSU, is used to invoke COPY, DUMP, and RESTORE processing. It can process one or more volumes, or one or more data sets. If the source and target volumes are both on the same RAMAC Virtual Array, DFSMSdss will automatically attempt to invoke SnapShot. If the source and target of a COPY operation are both in the same ESS, FlashCopy will be attempted automatically.

There are three modes that can be used by DFSMSdss when requesting to `COPY` a disk volume, as illustrated in Figure C-1.

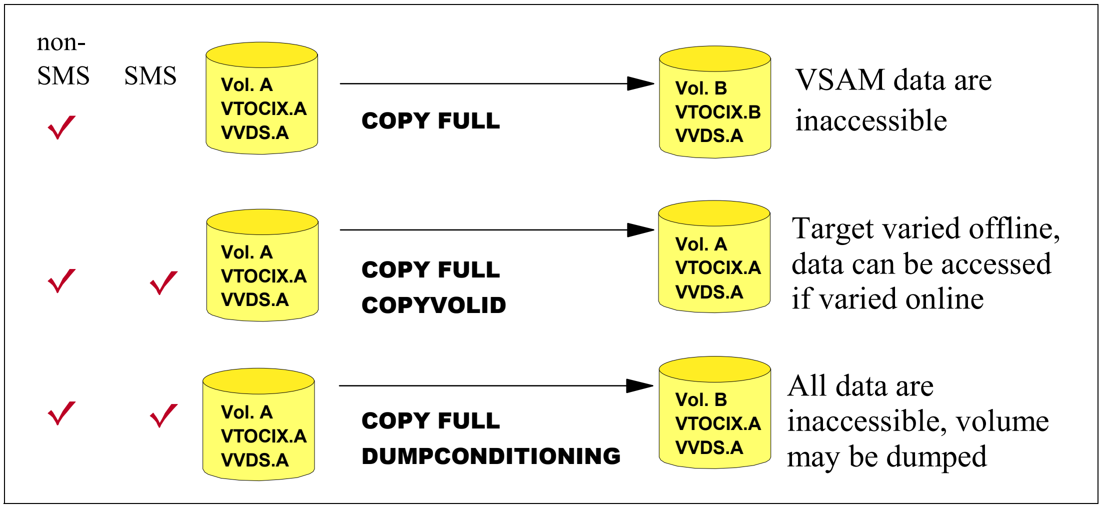



*Figure C-1 COPY FULL modes*

- **COPY FULL:** For non-SMS volumes, DFSMSdss can copy volume A to volume B. The volume label, VTOC and VTOC Index (VTOCIX) remain associated with volume B, but the VVDS will refer to volume A. This allows uncataloged non-SMS data sets to be referenced directly from volume B. VSAM data sets are required to be cataloged and to have valid VVR entries in the VVDS, so these are inaccessible on the target device.
- **COPY FULL COPYVOLID:** For both SMS and non-SMS volumes, DFSMSdss can copy volume A to volume B. The volume label, VTOC, VTOC index, and VVDS all refer to volume A. However, you cannot have two volumes online in a single system with the same volume serial, so DFSMSdss automatically varies the target device offline. The data can be accessed from the primary system if the source is varied offline, and then the target is brought online. A secondary system could vary the device online and access VSAM data sets and SMS-managed non-VSAM data sets, provided catalog entries are available to refer to the data contained on the target device.
- **COPY FULL DUMPCONDITIONING:** For both SMS and non-SMS volumes, when `DUMPCONDITIONING` is specified, DFSMSdss can copy volume A to volume B, but will rename both the VTOC index and VVDS to match volume A. Although this will mean that all data will be inaccessible, it will allow a `DUMP` of the target volume, since it remains online. When the volume is restored, it only needs to be re-labeled to volume A, using the ICKDSF `REFORMAT` function.

  With a follow-on change provided by APAR OW48234, a `DUMP FULL` of this target device will modify the tape records so that it will appear as if the tape was created directly from volume A, allowing the volume to be restored without requiring ICKDSF.

If `COPY FULL` is specified with the `CONCURRENT` keyword, DFSMSdss will attempt to use FlashCopy first. If FlashCopy is not supported or the volumes are not eligible for FlashCopy at this time, DFSMSdss will then attempt to establish a Concurrent Copy session. If this fails, DFSMSdss will produce a copy using standard data movement. In either case, the ADR734I message will be issued at a point where other applications can be started.

If the `CONCURRENT` keyword is not specified, DFSMSdss will attempt to use FlashCopy first, and if unavailable, use standard data movement.

# C.6 Migrating from SIBBATCH to DFSMSdss

IXFP SIBBATCH can continue to be used for both RVA and ESS disk subsystems. If you specify DFDSS as the DATAMOVER option for SIBBATCH, then on an ESS a FlashCopy will be performed instead of a SnapShot, without any changes to JCL. Some installations may chose to retain IXFP to reduce the amount of JCL modifications required to introduce ESS into an existing RVA environment.

DFSMSdss supports both the RVA and the ESS disk subsystems, and is designed to invoke SnapShot or FlashCopy as appropriate, depending on the storage hardware capabilities. SIBBATCH users migrating from RAMAC Virtual Array to the ESS might also consider migrating from SIBBATCH to DFSMSdss for the following benefits:

- DFSMSdss is designed to process a list of data sets in a single `COPY` operation, SIBBATCH can only process one data set at a time. DFSMSdss provides powerful filtering capabilities that are designed to allow the user to process only the data sets that are desired, without having to specify the full name of each data set.
- During data set `COPY` operations, if the source data set resides on an ESS, DFSMSdss is designed to attempt to allocate the target data set on a volume within the same ESS to increase the possibility of using FlashCopy to perform the copy.
- DFSMSdss is designed to process multiple volume and data set `COPY` operations in parallel in a single batch job. SIBBATCH can only process one SNAP operation at a time within a single batch job.
- SIBBATCH only provides a single function, duplicating data. DFSMSdss is designed to perform this function and many more including, but not limited to backup and restore, reducing free space fragmentation, and converting data to and from SMS management.

Here are some additional considerations when migrating from RVA SnapShot to ESS FlashCopy:

- With FlashCopy V2, a FlashCopy source can have up to 12 concurrent FlashCopy targets at the same time. FlashCopy also has limits to the number of relationships that can exist on a volume and within the ESS (these limits are far beyond what is practically necessary). With SnapShot you can make as many copies of a source or target as you like.
- You cannot FlashCopy a target extent until the background copy completes, so you cannot immediately FlashCopy back to the original source from the target. However, FlashCopy can be used on other extents on the target volume.
- FlashCopy is much more serviceable than SnapShot. The software can provide much more information about why FlashCopy could not be used for a particular operation or why an attempt to use FlashCopy failed, than what is provided by the SnapShot software.

## C.6.1 Migrating volume copy operations from SnapShot to FlashCopy

These are some considerations when migrating from an environment that exploits SnapShot for volume copy to an environment that exploits FlashCopy:

- Serialization for FlashCopy is the same as for SNAP VOLUME, that is, DFSMSdss holds serialization on the VTOC. In order to prevent application update of the data, you must stop or quiesce, or otherwise prevent applications from updating the volumes until the copies are complete. With FlashCopy, the copy is considered complete when the FlashCopy relationship is established.
- When you issue a SNAP VOLUME command, you have several alternatives for how you want to process the target volume through the COPYVOLID and CONDVOL keywords. Figure C-2 illustrates the different options available with DFSMSdss in comparison with SnapShot.

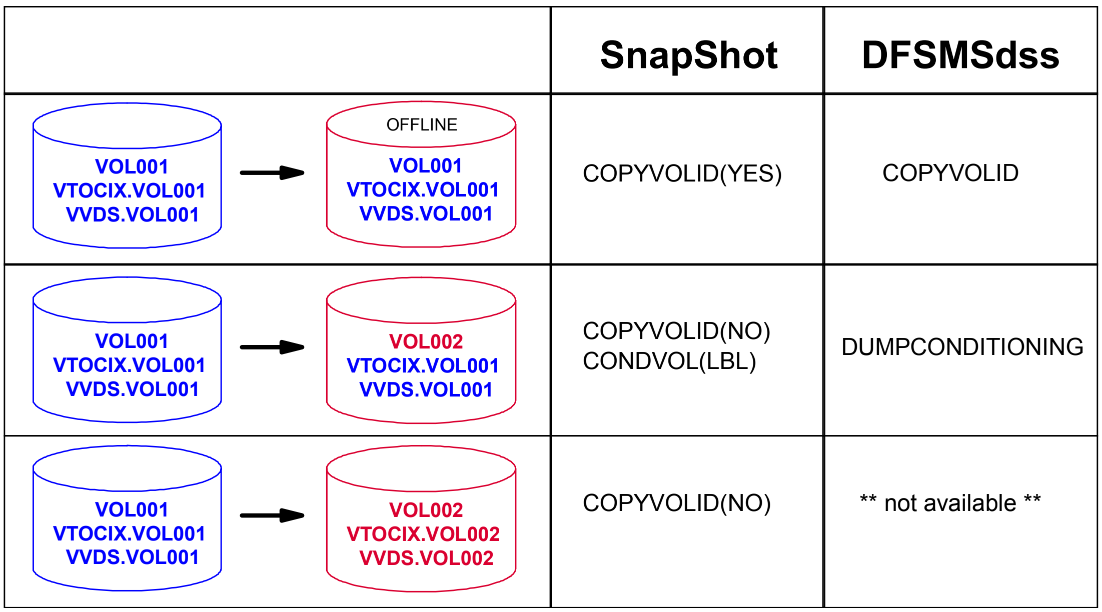



*Figure C-2 SNAP VOLUME vs. DFSMSdss COPY FULL*

Changes to your operations will be necessary only if you are currently using SnapShot with the COPYVOLID(NO) option. Alternatively, if you plan to retain the IXFP software, for example in a mixed ESS/RVA environment, you may run the SNAP VOLUME commands specifying DFSMSDSS as the DATAMOVER.

Table C-1 relates the keywords used with IXFP SIBBATCH to those used with DFSMSdss, for volume copy operations.

*Table C-1 SNAP VOLUME command to DFSMSdss COPY FULL*

| SIBBATCH Keyword | DFSMSdss Keyword | Comments |
|---|---|---|
| CONDITIONVOLUME | N/A or DUMPCONDITIONING | There is no corresponding DFSMSdss keyword for the ALL subparameter of the SIBBATCH CONDITIONVOLUME keyword. The corresponding DFSMSdss keyword for the LABEL subparameter of the SIBBATCH CONDITIONVOLUME keyword is DUMPCONDITIONING. |

| SIBBATCH Keyword | DFSMSdss Keyword | Comments |
|---|---|---|
| COPYVOLID | COPYVOLID | |
| DATAMOVERNAME | N/A | |
| DEBUG | DEBUG(FRMSG(MINIMAL \| SUMMARIZED \| DETAILED)) or DEBUG = FRMSG | DEBUG = FRMSG is specified as a parameter on the DFSMSdss EXEC statement |
| INDDNAME | INDDNAME | |
| OUTDDNAME | OUTDDNAME | |
| REPLACE | N/A | The default processing for DFSMSdss is the same as SIBBATCH processing when REPLACE is specified. |
| SOURCE | N/A or INDYNAM | There is no corresponding DFSMSdss keyword for the UNIT subparameter of the SIBBATCH SOURCE keyword. The corresponding DFSMSdss keyword for the VOLUME subparameter of the SIBBATCH SOURCE keyword is INDYNAM. |
| TARGET | N/A or OUTDYNAM | There is no corresponding DFSMSdss keyword for the UNIT subparameter of the SIBBATCH TARGET keyword. The corresponding DFSMSdss keyword for the VOLUME subparameter of the SIBBATCH TARGET keyword is OUTDYNAM. |
| TOLERATEENQFAILURE | TOLERATE(ENQFAILURE) | |
| TRACE | N/A | |

## C.6.2 Migrating data set copy operations from SnapShot to FlashCopy

These are some considerations when migrating from RVA SnapShot to ESS FlashCopy for data set copy operations:

- DFSMSdss can process a list of data sets in a single `COPY` operation, SIBBATCH can only process one data set at a time. DFSMSdss provides powerful filtering capabilities that allow the user to process only the data sets that are desired, without having to specify the full name of each data set.
- DFSMSdss does not have the capability to REPLACE and RENAME in the same command. SnapShot using IXFP SIBBATCH has the capability to specify REPLACE and RENAME.
- DFSMSdss does not use fast replication for non-SMS multivolume indexed VSAM data sets copied to multivolume target. SIBBATCH with DATAMOVER(DSS) will use FlashCopy to move these data sets.

There are two options for migrating from data set-level SnapShot to data set FlashCopy: Either continue using the IXFP program, or use DFSMSdss instead of IXFP. Table C-2 relates the keywords used by the two programs for data set copy operations.

*Table C-2 SNAP DATASET command o DFSMSdss COPY DATASET*

| SIBBATCH Keyword | DFSMSdss Keyword | Comments |
|---|---|---|
| CATALOG | CATALOG, RECATALOG | RECATALOG can be used to add catalog entry to specific catalog which may be outside of standard search order |
| DATACLASS | N/A | |
| DATAMOVERNAME | N/A | DFSMSdss may use underlying utilities to move certain data. No external options are provided. |
| DEBUG | DEBUG(FRMSG(MINIMAL \| SUMMARIZED \| DETAILED)) or DEBUG = FRMSG | DEBUG = FRMSG is specified as a parameter on the DFSMSdss EXEC statement |
| ESOTERIC | N/A | DFSMSdss uses standard ACS routines to determine allocation. You can also specify the volume of the target data set for non-SMS allocations. |
| FORCE | FORCE | |
| HOSTCOPYMODE (SHARED \| EXCLUSIVE) | SHARE \| (default) | If SHARE keyword is not specified with DFSMSdss, EXCLUSIVE is the default. |
| INDDNAME | INDDNAME | |
| MANAGEMENTCLASS | MGMTCLAS | |
| OUTDDNAME | OUTDDNAME | |
| RELATE | SPHERE | When processing an alternate index without also processing the base cluster, DFSMSdss automatically relates the target alternate index to the source base cluster. In this case, there is no DFSMSdss keyword that corresponds to the SIBBATCH RELATE keyword.<br>When processing the entire sphere using the DFSMSdss SPHERE keyword, DFSMSdss relates the alternate index(es) to the target base cluster. |
| REPLACE(YES \| NO) | REPLACE \| (default) | If REPLACE keyword is not specified for DFSMSdss, then existing data sets will not be replaced. |
| SOURCE | DATASET | DFSMSdss supports filters that allow multiple data sets to be processed. |
| STORAGECLASS | STORCLAS | |

| SIBBATCH Keyword | DFSMSdss Keyword | Comments |
|---|---|---|
| TARGET | RENAMEU | DFSMSdss supports using the same or different name for a target than the source data set. The RENAMEU keyword is used to specify if the target is to be different. |
| TOLERATEENQFAILURE (YES I NO) | TOLERATE(ENQFAILURE) I (default) | If TOLERATE(ENQFAILURE) is not specified for DFSMSdss, the data set must be serialized. |
| TRACE | N/A | Although DFSMSdss does not cut GTF trace records, this does not affect operation of data movement. |
| VOLUME | OUTDYNAM | |
| VOLUMECOUNT | VOLCOUNT | |

# C.7 Migration scenarios

This section provides examples on how typical implementations of SnapShot may be replaced with DFSMSdss and FlashCopy.

## C.7.1 Backup for Disaster Recovery

**User requirement:** A mission-critical application has its data stored on many volumes contained within the user's disk subsystems. The user is concerned that a disaster could impact its business continuance (Figure C-3).

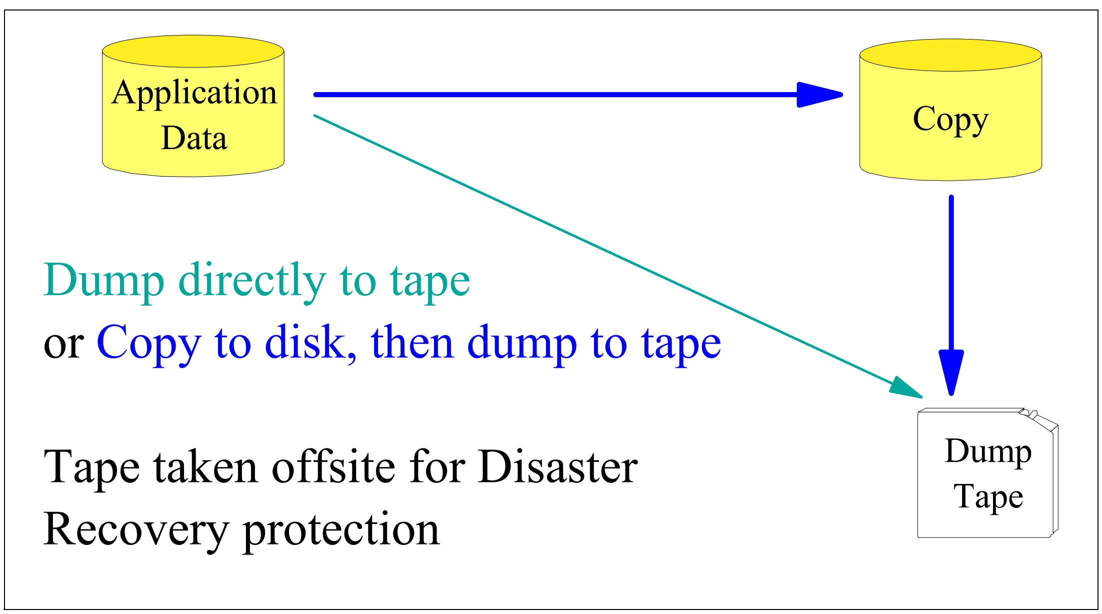



*Figure C-3 User requirement - backup for Disaster Recovery*

As shown in Figure C-3, the user might have implemented two solutions: Dump directly to tape, or copy to disk then dump to tape. The user might be dumping directly to tape using Virtual Concurrent Copy. However, if the dump is not created successfully because of hardware errors in the tape drive or media, the dump cannot be restarted, and updates may have already been made to the source data set. To avoid this, the user might be employing volume-level SnapShot to minimize the impact to the application. The target disk copies are then dumped to tape cartridges that are taken offsite. If the dump is unsuccessful, the SnapShot image is still available to retry the dump again.

**Migration approach:** Volume-level copies can be handled using FlashCopy. The source volumes are copied to target volumes using the DFSMSdss **`COPY FULL`** command with **`DUMPCONDITIONING`**. This creates a target volume which retains its original target volume serial, but has VTOC index and VVDS that match the source volume.

The DFSMSdss **`DUMP FULL`** command can be used to dump these target volumes to tapes that are taken offsite. If the dump is unsuccessful, the FlashCopy image is still available to retry the dump again.

## C.7.2 Instream backup for job restart

**User requirements:** A critical, long-running batch job stream makes updates to a data set. The user is concerned that a job abend in a later step would require restarting the job from the very beginning (Figure C-4).

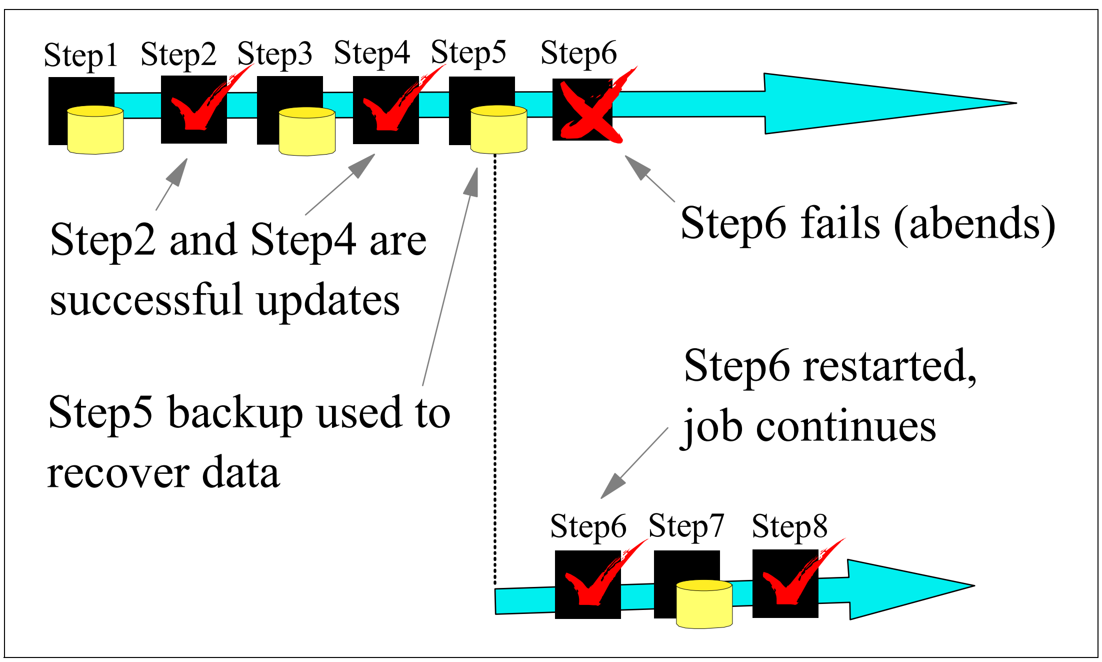



*Figure C-4 User requirement - instream backup for job restart*

In this scenario, the user might have employed data set SnapShot throughout the job stream. In Figure C-4, steps 1, 3, 5 and 7 perform the SnapShot. Steps 2, 4, 6 and 8 are application steps that update the data set. If step 6 failed, for example, the user could snap back the copy made in step 5, and restart the job at step 6.

**Migration approach:** Data set FlashCopy can be invoked from DFSMSdss.

The backup in step 1 is done through **`COPY DATASET`** command with the **`FASTREPLICATION(PREFERRED)`** keyword. This indicates that FlashCopy is desired, but to go ahead and perform the copy the traditional way if for any reason the copy is not eligible at that time for this process.

To minimize impact to the application, steps 3, 5, and 7 request **`COPY`** with **`FASTREPLICATION(REQUIRED)`**, which indicates that the FlashCopy function is required, and to skip the backup if FlashCopy is unavailable or data set is not eligible for FlashCopy.

In the event that step 6 failed, and the backup in step 5 was skipped or discarded, the user would recover the data from the backup in step 3, and rerun the job stream beginning at step 4 instead. Background copy must be completed before FlashCopy can be used to copy data from the backup to the source volumes; alternatively, the data may be copied by traditional means.

## C.7.3 Data mining

**User requirement:** A copy of a multi-volume production VSAM database is needed for data mining application. The user is concerned that large structured queries will impact the production workload environment (Figure C-5).

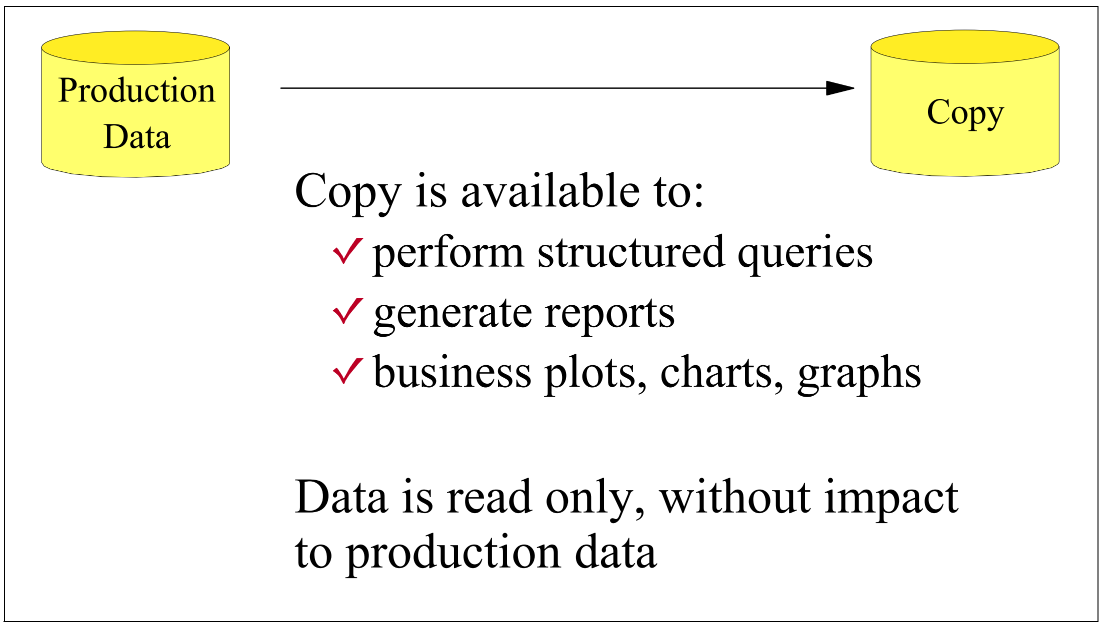



*Figure C-5 User requirement - data mining*

As shown in Figure C-5, the user is using volume-level SnapShot with the COPYVOLID(YES) keyword to make the copies with minimal impact to production workload. Once the copies were made, they are brought online to another system which performs the structured queries and generated reports, business plots, and graphs.

**Migration approach:** FlashCopy can be used to copy the volumes. Using DFSMSdss **`COPY FULL COPYVOLID`**, the target volumes have the same volume serials as the source volumes. DFSMSdss is designed to vary the target devices offline. The copies are brought online to another system which performs the structured queries and generates the reports, business plots, and graphs.

## C.7.4 Application development

**User requirement:** Application developers need a copy of production data sets to perform a series of tests. These data sets are scattered across many volumes, and intermixed with other data. The requirement is that the application testers can use realistic data, but without impacting production workload operations (Figure C-6).

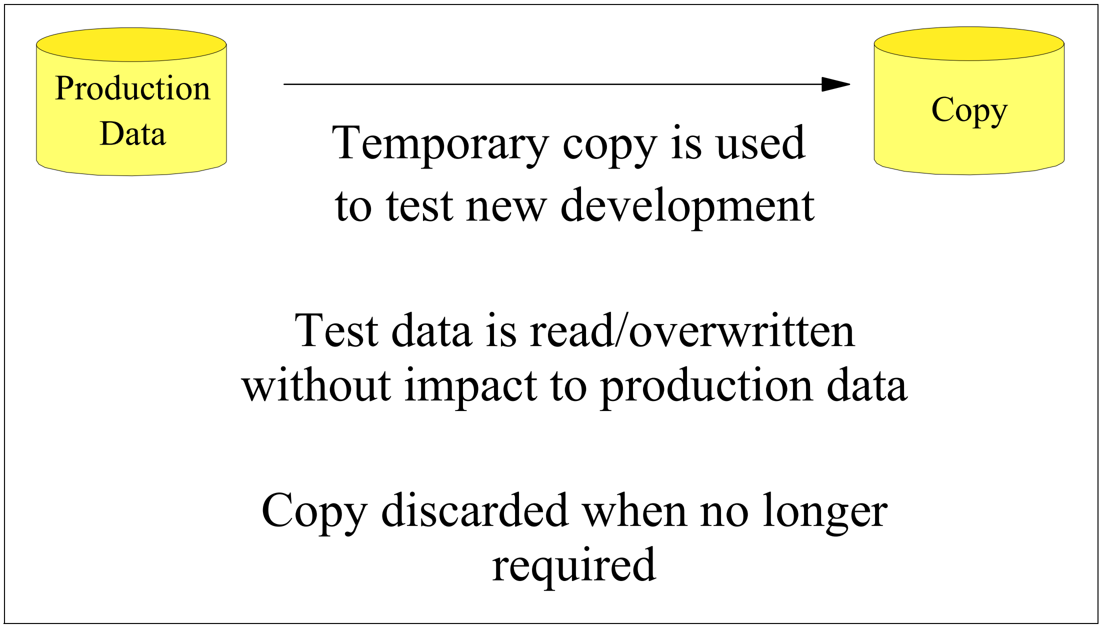



*Figure C-6 User requirement: application development*

In this scenario, the user might have implemented data set-level SnapShot, renaming the data sets to a new high level qualifier accessible to the developers. This method is also called *application cloning* and can be used for training new employees or for other purposes besides testing. As presented in Figure C-6, the test data can be updated without impact to production data and is discarded when no longer required.

**Migration approach:** Using the DFSMSdss `COPY DATASET` command, copies of the required data sets are made to a new set of volumes. The `COPY` command can rename all the data sets to a new high-level qualifier. The developers can run their tests on these volumes without impacting the performance of production workloads.

## C.7.5 Multiple read access

**User requirement:** Three mission-critical applications that run concurrently require fast performance to read the same data set (Figure C-7).

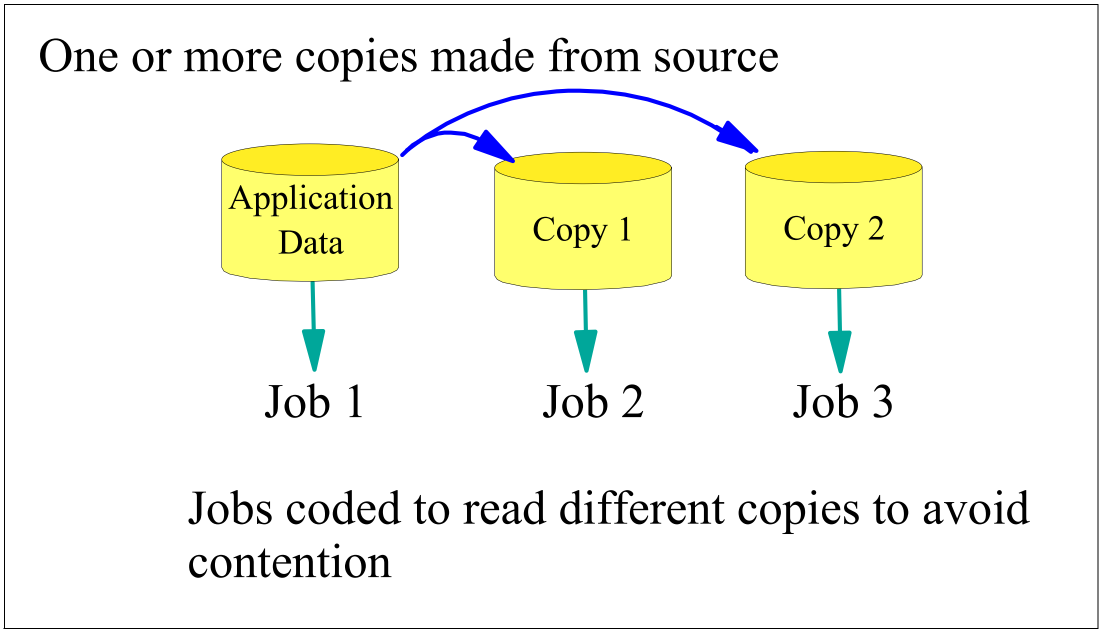



*Figure C-7 User requirement - multiple read access*

In this scenario, the user might have implemented data set SnapShot to make three identical copies of the data (see Figure C-7). Volume A contains the original data set X; volume B contains the first SnapShot, data set Y; volume C contains the second SnapShot, data set Z. Application Job1 reads the original data set X; application job2 reads the first copy, data set Y; application job3 reads the second copy, data set Z.

If all the jobs were running on the same system, Hiperbatch™ may have been used instead.

**Migration approach:** ESS is a fast disk storage subsystem. To provide concurrent read access, it offers Parallel Access Volumes (PAV). With PAV, the same volume is given multiple UCB device numbers to allow concurrent access. In our example, volume A has a base UCB 100, and two alias UCBs 101 and 102. The applications all read data set X on volume A, but the z/OS operating system is designed to automatically direct the different read requests through different UCB device numbers to avoid contention. This helps reduce or eliminate the IOSQ time normally associated with device contention.

With Dynamic PAV, the ESS allows a pool of alias UCB device numbers to be available. Workload Manager in goal mode is designed to automatically detect the volume that requires more alias UCB device numbers, and assigns to that volume.

When applications are running on different z/OS images, ESS offers similar support called Multiple Allegiance (MA). MA is designed to provide concurrent access to volumes between z/OS images. This helps reduce or eliminate the PEND time normally associated with multi-host contentions.

## C.8 References

The following Web sites are helpful for reference material concerning ESS and DFSMS:

`http://www.storage.ibm.com/hardsoft/products/ess/ess.htm`
`http://www.storage.ibm.com/software/sms/smshome.htm`

The following redbooks can be referenced for useful information:

- *IBM TotalStorage Enterprise Storage Server Model 800,* SG24-6424
- *IBM TotalStorage Enterprise Storage Server: Implementing the ESS in Your Environment,* SG24-5420-01

D

# ESS Copy Services feature codes

In this appendix we summarize the feature codes used to configure and order the available options for the Copy Services functions for the ESS, including Copy Services Version 2 for PPRC and FlashCopy.

**Important:** When reading the information in this appendix, these considerations apply:

- Features are described in this section for technical illustration purposes. Information about prerequisites and co-requisites among the features is not always included.
- The descriptions do not include information on whether the features are priced or available at no additional cost.
- Not all the ESS features are listed in this section, only the ones needed to enable any of the copy functions discussed in this redbook.

The complete list of features and complete ordering instructions and prerequisites can be found in the IBM TotalStorage Enterprise Storage Server Model 800 hardware announcement letters on the Web at:

`http://www.ibm.com`

## Advanced functions

The IBM TotalStorage Enterprise Storage Server (ESS) advanced functions enhance the capabilities of the ESS:

- Extended Remote Copy (XRC) is a combined hardware and software business continuance solution for the zSeries and S/390 environments providing asynchronous mirroring between two ESSs at global distances; see Chapter 5, “Extended Remote Copy” on page 207.
- Peer-to-Peer Remote Copy (PPRC) is a hardware-based business continuance solution designed to provide synchronous mirroring between two ESSs that can be located up to 300 km from each other, depending on the path type. PPRC is described in detail in Chapter 3, “Peer-to-Peer Remote Copy” on page 69.
  - PPRC V1 includes these functions:
    - Synchronous PPRC is provided.
    - PPRC Extended Distance (PPRC-XD) remote copy function for non-synchronous mirroring between two ESSs over continental distances is provided (the distance is only limited by the network and channel extenders technology capabilities).
  - PPRC V2 includes additional functions:
    - Asynchronous Cascading PPRC for the Fxx and 800 models is provided with ESS LIC 2.2.0 or higher level.
    - PPRC with FCP for the ESS 800 and ESS 750 is provided with LIC 2.3.0 or higher.
    - Asynchronous PPRC for a two site disaster recovery solution for the ESS 800 and ESS 750 is provided with ESS LIC 2.4.0.
- FlashCopy is designed to provide a point-in-time instant copy capability for logical volumes in the ESS. FlashCopy is described in detail in Chapter 2, “FlashCopy” on page 13.
  - FlashCopy V1 requires that the point-in-time copy of an ESS logical volume be in the same logical subsystem as the source volume.
  - FlashCopy V2 includes additional functions:
    - Data Set Level Copy (zSeries environment only)
    - Multiple relationship FlashCopy
    - Improved Performance/reduced establish time (up to 10 times)
    - Consistency Groups
    - Inband commands over PPRC Links
    - Spanning of LSS boundaries
- To order these advanced functions, you need the Authorization features, together with the corresponding ESS features for the advanced functions.

**Attention:** The following combinations of versions of the advanced copy functions are not supported on the *same* ESS:

- FlashCopy V1 and FlashCopy V2
- PPRC V1 and PPRC V2
- FlashCopy V1 and PPRC V2

Copy Services advanced features need to be installed on each ESS that is involved in any Copy Services operation. This means that both the ESS containing the primary volume and the ESS containing the secondary volume must have the minimum features to support the operation being performed on the copy pair. Here are some examples:

- If a Synchronous PPRC pair is being established, then both ESSs must have a minimum of PPRC V1.
- If the path is using PPRC over FCP, then PPRC V2 is required as a minimum on both.
- With a Synchronous PPRC over ESCON pair, it is acceptable for one ESS to have PPRC V1 and the other PPRC V2, as the minimum requirement is met on both ESSs.
- If PPRC V2 is used in an Asynchronous Cascading configuration, PPRC V2 must be purchased for the intermediate ESS.

# Feature Codes

The advanced functions of the ESS are ordered as optional features, either initially when ordering the ESS, or later as a field upgrade to the ESS.

The advanced function feature codes on the Fxx models are based on the *RAID-5 effective capacity*. The advanced function feature codes for the Model 800 and Model 750 are based on *physical capacity*.

The initial activation of an advanced function and the installation of a larger license is a concurrent activity assuming that the appropriate level of microcode is installed on the machine. The removal of a license to deactivate an advanced function is a disruptive activity and requires a machine IML. To upgrade from a V1 to a V2 feature is a concurrent activity.

# ESS Model Exx and ESS Model Fxx

Peer-to-Peer Remote Copy (PPRC) V1 and FlashCopy V1 are applicable to E10, E20, F10, or F20 environments. Table D-1 lists the Copy Services Feature Codes for V1.

*Table D-1 Copy Services Feature Codes V1*

| Feature Code | Description | Requirement | ESS support |
|---|---|---|---|
| 1830 - 1835 | FlashCopy | LIC 1.3.0 or later | Models E10, E20, F10, or F20 |
| 1820 - 1825 | PPRC/PPRC-XD | LIC 1.3.0 or later | Models E10, E20, F10, or F20 |
| 1830 - 1838 | FlashCopy | LIC 1.5.0 or later | Models F20 |
| 1820 - 1828 | PPRC/PPRC-XD | LIC 1.5.0 or later | Models F20 |

PPRC/PPRC-XD V2 and FlashCopy V2 are applicable to F10 or F20 environments. Table D-2 lists the Copy Services Feature Codes for V2.

*Table D-2 Copy Services Feature Codes V2*

| Feature Code | Description | Requirement | ESS support |
|---|---|---|---|
| 1860- 1868 | FlashCopy | LIC 2.2.0 | Models F10, F20 |
| 1850 - 1858 | PPRC/PPRC-XD | LIC 2.2.0 | Model F10, F20 |

The advanced function feature codes on the Exx and Fxx models are based on the *RAID-5 effective capacity*. Table D-3 lists the Feature Codes for the ESS models FXX at the different capacities.

*Table D-3 PPRC and FlashCopy V1 and V2 Feature Codes for the ESS Models Fxx*

| ESS Effective Capacity | XRC | PPRC Version 1 | FlashCopy Version 1 | PPRC Version 2 | FlashCopy Version 2 |
|---|---|---|---|---|---|
| | Feature Code | Feature Code | Feature Code | Feature Code | Feature Code |
| Up to 0.5 TB | 1810 | 1820 | 1830 | 1850 | 1860 |
| Up to 1 TB | 1811 | 1821 | 1831 | 1851 | 1861 |
| Up to 2 TB | 1812 | 1822 | 1832 | 1852 | 1862 |
| Up to 4 TB | 1813 | 1823 | 1833 | 1853 | 1863 |
| Up to 8 TB | 1814 | 1824 | 1834 | 1854 | 1864 |
| Up to 12 TB | 1815 | 1825 | 1835 | 1855 | 1865 |
| Up to 16 TB | 1816 | 1826 | 1836 | 1856 | 1866 |
| Up to 20 TB | 1817 | 1827 | 1837 | 1857 | 1867 |
| Up to 25 TB | 1818 | 1828 | 1838 | 1858 | 1868 |

Here are some things to note about the feature codes:

- FlashCopy and PPRC V1 feature codes, 183x and 182x, can be upgraded to FlashCopy and PPRC V2 feature codes, 186x and 185x, using *Feature Exchange*.
- The FlashCopy and PPRC V1 to FlashCopy and PPRC V2 upgrade must be for a license of the same or greater capacity.
- A feature exchange of a lower capacity level (a numerically lower feature number) is not supported.
- FlashCopy and PPRC V2 (186x, 185x) to FlashCopy and PPRC V1 (183x, 182x) downgrades are not supported.
- Asynchronous PPRC and PPRC over FCP are not supported on the Fxx models with PPRC V2.

# ESS Model 800

The ESS Model 800 advanced functions are enabled and authorized based upon the *physical capacity* of the ESS.

The IBM 2240 ESS Function Authorization feature numbers provide a set of pricing tiers for the ESS advanced functions. These tiers provide increased granularity (as compared to earlier models) with pricing matched to the physical capacity of the ESS Model 800. They are used for billing purposes only.

The ESS Model 800 (2105-800) feature numbers (8xxx) and the IBM 2240 ESS Function Authorization feature numbers (8xxx) are co-requisites and must always correspond to one another (these are shown in Table D-4):

- The ESS Model 800 feature numbers (81xx, 82xx, 83xx, 85xx, and 86xx) enable a given function on the ESS at a given capacity level. PPRC and FlashCopy enabling and authorization must be equal to or greater than the total *physical capacity* of the ESS.
- The ESS Function Authorization feature numbers (81xx, 82xx, 83xx, 85xx, and 86xx) authorize use of the given Advanced Function at the given capacity level on the ESS machine for which it was purchased.

See Example D-1 for an example of ordering the Advanced Copy Services for an IBM ESS 800.

*Example: D-1 Ordering Advanced Copy Services*

To order FlashCopy V2 and PPRC V2 for an IBM ESS 800 with 4 TB of physical capacity, the client would specify:

- ESS Model 800 (2105-800) feature of 8503 and 8603
- ESS Function Authorization numbers of 2240-PRC (8503) and 2240-FLC (8603)

*Table D-4 ESS Model 800 (2105-800) and IBM 2240 ESS function authorization features*

| Physical Capacity tier | ESS Model 800 (2105-800) feature codes IBM 2240 ESS Function Authorization feature numbers | | | | |
|---|---|---|---|---|---|
| | Extended Remote Copy | Peer-to-Peer Remote Copy 2240-PRC | | FlashCopy 2240-FLC | |
| | | Version 1 | Version 2 | Version 1 | Version 2 |
| Up to 1 TB | 8100 | 8200 | 8500 | 8300 | 8600 |
| Up to 2 TB | 8101 | 8201 | 8501 | 8301 | 8601 |
| Up to 3 TB | 8102 | 8202 | 8502 | 8302 | 8602 |
| Up to 4 TB | 8103 | 8203 | 8503 | 8303 | 8603 |
| Up to 5 TB | 8104 | 8204 | 8504 | 8304 | 8604 |
| Up to 6 TB | 8105 | 8205 | 8505 | 8305 | 8605 |
| Up to 8 TB | 8106 | 8206 | 8506 | 8306 | 8606 |
| Up to 10 TB | 8107 | 8207 | 8507 | 8307 | 8607 |
| Up to 12 TB | 8108 | 8208 | 8508 | 8308 | 8608 |
| Up to 16 TB | 8109 | 8209 | 8509 | 8309 | 8609 |
| Up to 20 TB | 8110 | 8210 | 8510 | 8310 | 8610 |
| Up to 25 TB | 8111 | 8211 | 8511 | 8311 | 8611 |
| Up to 30 TB | 8112 | 8212 | 8512 | 8312 | 8612 |
| Up to 40 TB | 8113 | 8213 | 8513 | 8313 | 8613 |
| Up to 50 TB | 8114 | 8214 | 8514 | 8314 | 8614 |
| Up to 60 TB | 8115 | 8215 | 8515 | 8315 | 8615 |
| Inactive 0TB | 8199 | 8299 | 8599 | 8399 | 8699 |

Here are some other things to remember:

- Upgrades from PPRC and FlashCopy (features #82xx and #83xx, respectively) to PPRC Version 2 and FlashCopy Version 2 (features #85xx and #86xx, respectively) should be processed using the *Feature Exchange* option.
- The feature exchange can be made to either the same or a higher capacity level.
- The feature exchange must be processed against both the ESS Model 800 8xxx feature number and IBM 2240 ESS Function Authorization 8xxx feature number.
- Downgrades are not permitted.

# PPRC over Fibre Channel links

PPRC over Fibre Channel is supported on the ESS Model 800 and ESS Model 750, and is provided with PPRC Version 2. PPRC Version 2 is an optional feature for the ESS (#85xx). PPRC over Fibre Channel also requires that the ESS is at a minimum LIC level 2.3.0.

# Asynchronous PPRC

To indicate the use of Asynchronous PPRC, marketing representatives will update their client's records to use the new feature code, 9912, Asynchronous PPRC indicator. It is only used for administrative purposes.

Along with PPRC V2, FlashCopy V2 is required on the secondary ESS. If Asynchronous PPRC will be used during failback from the secondary ESS machine(s), FlashCopy Version 2 must also be enabled on the primary machine(s).

# ESS Model 750

The ESS Model 750 was announced on April 7, 2004 as a new member of the ESS family, packaged for affordability and well-suited for clients with midrange capacity and performance need.

As with the ESS Model 800, the advanced functions require the selection of IBM 2105 Model 750 feature numbers and the purchase of the matching IBM 2240 ESS Function Authorization feature numbers:

- The 750 feature numbers (82xx, 83xx, 85xx, and 86xx) enable a given function on the ESS at a given capacity level.
- The ESS Function Authorization feature numbers (82xx, 83xx, 85xx, and 86xx) authorize use of the given Advanced Function at the given capacity level on the ESS machine for which it was purchased.

The ESS Model 750 feature numbers (8xxx) and the ESS Function Authorization feature numbers (8xxx) are co-requisites and must always correspond to one another; see Table D-5.

> **Note:** XRC is not supported on the ESS Model 750.

*Table D-5 ESS Model 750 and IBM 2240 ESS Authorization features*

| **Physical Capacity Tiers** | **ESS Model 750 (2105-750) feature codes**<br>**IBM 2240 Function Authorization feature numbers** | | | |
|---|---|---|---|---|
| | **Peer-to-Peer Remote Copy**<br>**2240-PRC** | | **FlashCopy**<br>**2240-FLC** | |
| | **Version 1** | **Version 2** | **Version 1** | **Version 2** |
| Up to 1 TB | 8200 | 8500 | 8300 | 8600 |
| Up to 2 TB | 8201 | 8501 | 8301 | 8601 |
| Up to 3 TB | 8202 | 8502 | 8302 | 8602 |
| Up to 4 TB | 8203 | 8503 | 8303 | 8603 |
| Up to 5 TB | 8204 | 8504 | 8304 | 8604 |

Here are some other things to remember:

- Upgrades from PPRC and FlashCopy (features #82xx and #83xx, respectively) to PPRC Version 2 and FlashCopy Version 2 (features #85xx and #86xx, respectively) should be processed using the *Feature Exchange* option.
- The feature exchange can be made to either the same or a higher capacity level.
- The feature exchange must be processed against both the ESS Model 750 8xxx feature number and IBM 2240 ESS Function Authorization 8xxx feature number.
- Downgrades are not permitted.

E

# System Adapter ID (SAID)

In this appendix we provide some additional explanation for System Adapter IDs (SAIDs).

# SAID definition with ESCON links

For PPRC primary to secondary unit (channel to control unit) communication, a maximum of eight ESCON links using a modified ESCON protocol can be configured.

ESCON channels provide 160 Mbps point-to-point links. While PPRC can be bi-directional, these links are uni-directional. The primary unit ESCON port (the one in channel mode) has to be dedicated for PPRC. The ESCON port on the secondary unit can be also used for S/390 host attachment, provided an ESCON director is used and the host is connected to it.

The ESCON adapters are specified by their System Adapter ID (SAID). To alleviate some of the confusion regarding these terms, we will now explain adapter IDs: SA IDs and SA tags.

Adapter IDs are internal identifications assigned to each external port on the machine and are used for communication between the adapters and the SMP. They are also the adapter numbers we use in the state save formatter to format data for a particular adapter.

An SA Tag is a device used to find the physical location of an port in the machine. It is part of the ESCON node descriptor and is used for link-fault isolation. It is also used for identifying a path through which a PPRC path is established. The functional specification also uses the term SA ID to refer to the tag. Recently, a new definition of the SA tag was approved in the functional specification, and code will soon be promoted to switch to the new format. This will affect each and every friend chain or TSO command or Web interface that issues an establish PPRC path subsystem function.

Figure E-1 shows the how the adapter IDs and tags map to physical ESCON port locations.

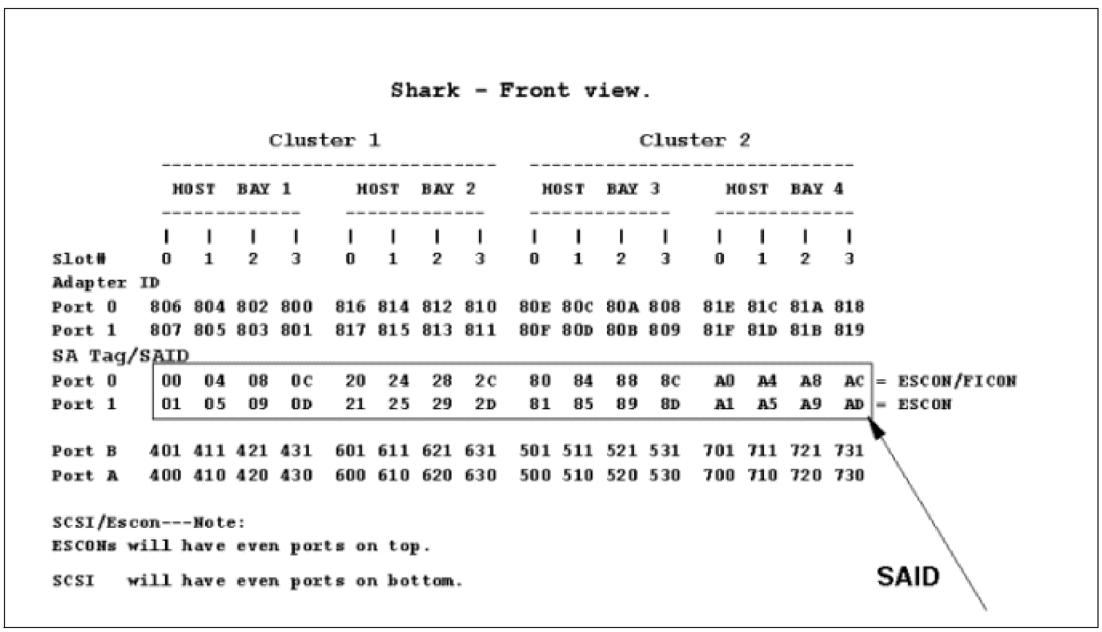



*Figure E-1 Physical ESCON port location*

# PPRC paths using ESCON links

To use PPRC, you must have ESCON host adapter ports on both the primary and secondary ESS for the peer-to-peer links. An ESCON path can either be operating as a control unit or a PPRC channel. For highest availability, IBM recommends that each ESS should have more than one ESCON adapter for PPRC connectivity. See the SAID numbers in Figure E-2.

Before PPRC pairs can be established, logical paths must be defined between the logical control unit images. The ESS supports up to 16 CKD logical control unit images and up to 16 FB controller images. An ESCON adapter supports up to 64 logical paths. A pair of LSSs can be connected with up to eight logical paths. You establish logical paths between control unit images of the same type over physical ESCON links

A path is used to send data between the source and target of PPRC pairs. The physical path consists of the ESCON connection between two Enterprise Storage Servers while a logical path describes the connection of the PPRC source and targets. There could be multiple logical paths established over a single physical path.

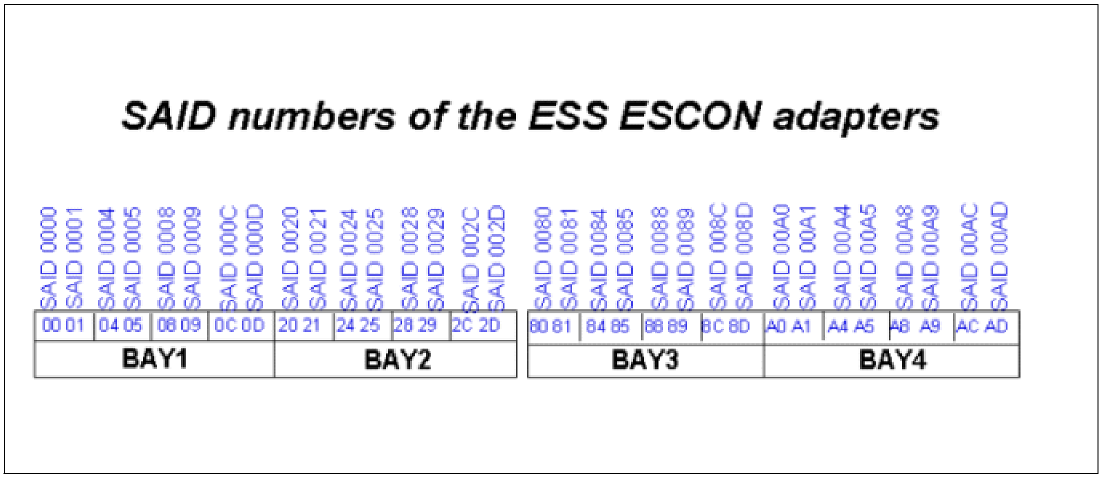



*Figure E-2 SAID numbers*

An ESCON port operating as a PPRC channel can support logical paths between many control-unit images whether it connects directly to another server or goes through a switch. The ESCON port has a maximum of 64 PPRC channels. A primary control-unit image can have a relationship with up to four secondary logical control-unit images. Each control-unit image pair can operate over a maximum of eight PPRC channels.

To initiate a PPRC operation through the ESSNet for a logical device, both the primary and secondary ESSs must be attached to the same ESSNet. Both the primary and secondary ESSs must also be in the same ESS Copy Services server domain. If the two ESSs are not on the same local ESSNet, you must extend the ESSNet through a local area network (LAN) to connect the two local ESSNets.

## SAID usage example

In Figure E-3 on page 394, the two ESS subsystems have been configured with identical configurations, and the two ESS subsystems have been connected with two ESCON links. To establish a path from the primary ESS on the left to the secondary ESS on the right, the user must determine the connection of the ESCON cables and record which system adapter IDs to connect the controller.

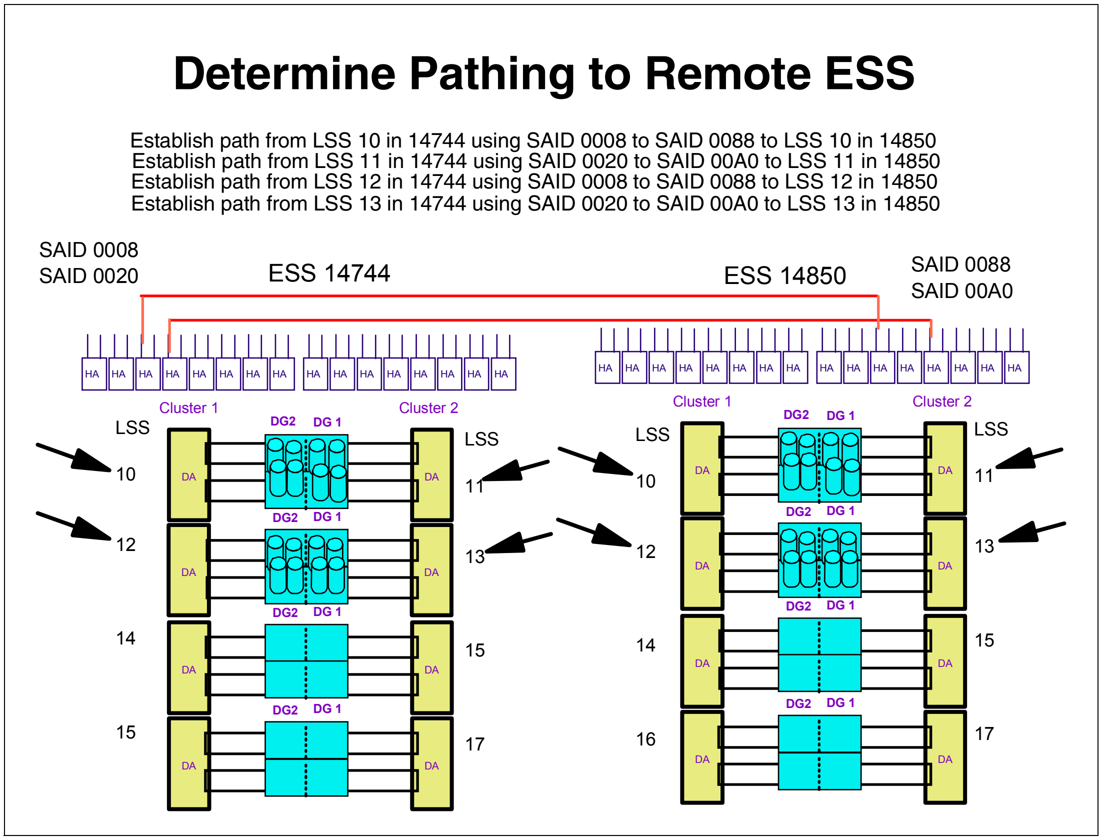



Figure E-3 Determine pathing to remote ESS

Using the chart, the path between two ESS subsystem can be determined. The user can then select the appropriate SAID, connect it to the remote ESS, and access the ESS that contains volumes that will be used in the PPRC pairing.

# FCP System Adapter ID

The Fiber Channel ports used for PPRC links are also specified by their *System Adapter* ID (SAID).

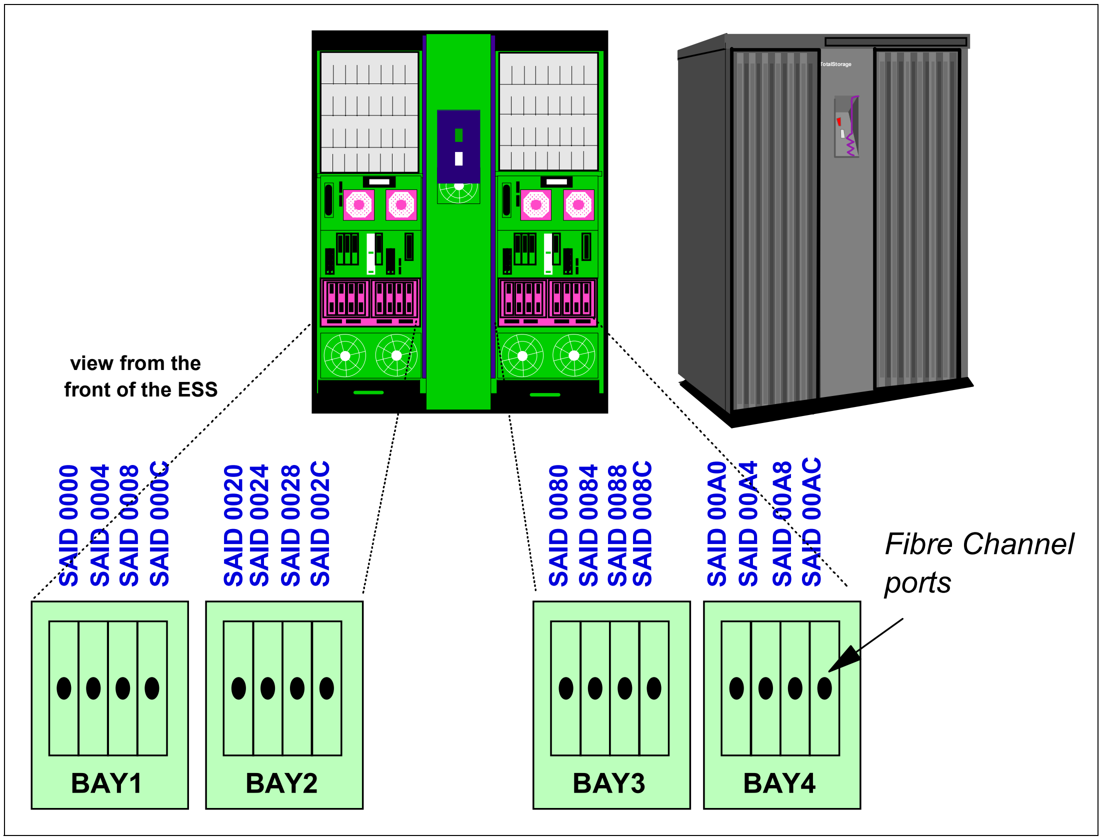



*Figure E-4 System Adapter ID (SAID) identification with Fibre Channel ports*

Figure E-4 illustrates how the system adapter IDs (SAIDs) are designated when the host adapter bays are seen from the front of the ESS. The first two values are zeros, and the last two values are the hexadecimal SAID byte values that correspond to the Fiber Channel port interface of the ESS.

# Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

## IBM Redbooks

For information on ordering these publications, see “How to get IBM Redbooks” on page 398. Note that some of the documents referenced here may be available in softcopy only.


- *The IBM TotalStorage Solutions Handbook,* SG24-5250
- *IBM TotalStorage Solutions for Disaster Recovery,* SG24-6547
- *IBM TotalStorage Enterprise Storage Server Implementing ESS Copy Services in Open Environments*, SG24-5757
- *IBM TotalStorage Enterprise Storage Server Model 800,* SG24-6424
- *IBM Tivoli Storage Resource Manager: A Practical Introduction,* SG24-6886
- *IBM TotalStorage Enterprise Storage Server PPRC Extended Distance*, SG24-6568
- *IBM TotalStorage Multiple Device Manager Usage Guide*, SG24-7097


## Other publications

These publications are also relevant as further information sources:


- *z/OS DFSMS Advanced Copy Services,* SC35-0428
- *z/OS DFSMSdfp Advanced Services*, SC26-7400
- *DFSMS Extended Remote Copy Installation Planning Guide*, GC35-0481
- *DFSMS Extended Remote Copy Reference Information for Advanced Users*, GC35-0482
- *Device Support Facilities User’s Guide and Reference,* GC35-0033
- *z/OS DFSMSdss Storage Administration Guide*, SC35-0423
- *z/OS DFSMSdss Storage Administration Reference,* SC35-0424
- *IBM TotalStorage Enterprise Storage Server Host Systems Attachment Guide*, SC26-7446
- *IBM TotalStorage Enterprise Storage Server Introduction and Planning Guide,* GC26-7444
- *IBM TotalStorage Enterprise Storage Server User’s Guide,* SC26-7445
- *IBM TotalStorage Enterprise Storage Server Web Interface User’s Guide,* SC26-7448
- *IBM TotalStorage Enterprise Storage Server Command-Line Interfaces User’s Guide,* SC26-7494
- *IBM TotalStorage Enterprise Storage Server Application Programming Interface Reference*, GC35-0489
- *IBM TotalStorage SAN Multiple Device Manager Command-Line Interface Guide*, SC26-7585
- *IBM TotalStorage SAN Multiple Device Manager Configuration Guide*, SC26-7586
- *IBM TotalStorage SAN Multiple Device Manager CIM Agent Developer’s Reference*, SC26-7587

## Online resources

These Web sites and URLs are also relevant as further information sources:

- ESS support information

  `http://www.storage.ibm.com/disk/ess/supserver.html`
- ESS Web site

  `http://www.storage.ibm.com/hardsoft/products/ess/ess.html`
- DFSMS SDM Copy Services

  `http://www.storage.ibm.com/software/sms/sdm/index.html`

## How to get IBM Redbooks

You can search for, view, or download Redbooks, Redpapers, Hints and Tips, draft publications and Additional materials, as well as order hardcopy Redbooks or CD-ROMs, at this Web site:

**ibm.com**/redbooks

## Help from IBM

IBM Support and downloads

**ibm.com**/support

IBM Global Services

**ibm.com**/services

# Index

## E

## F



## G

## T



## U



## V

# 8

# Solutions

This chapter gives a few suggestions of solutions that can be implemented with ESS Copy Services. Also presented are examples of procedures for managing outage and recovery situations using PPRC.

The chapter begins with a brief overview of Disaster Recovery. As this is the main purpose for ESS Copy Services, it is worth reviewing at this point.

Examples are given in the following areas:

- Database integrity and recovery
- PPRC-XD solutions
- Planned outage procedures
- Asynchronous Cascading PPRC solutions
- Asynchronous PPRC solutions
- PPRC and FlashCopy combined
- FlashCopy solutions
- GDPS

## 8.1 Disaster Recovery

In this section we discuss the various solutions that can be used for Disaster Recovery.

### 8.1.1 What is Disaster Recovery?

How would a shutdown of your IT system affect your business? What about a site disaster? Is your business-critical processing and data protected from a site disaster? Do you put off system maintenance and upgrades to avoid system downtime? Figure 8-1 illustrates the nature of Disaster Recovery.

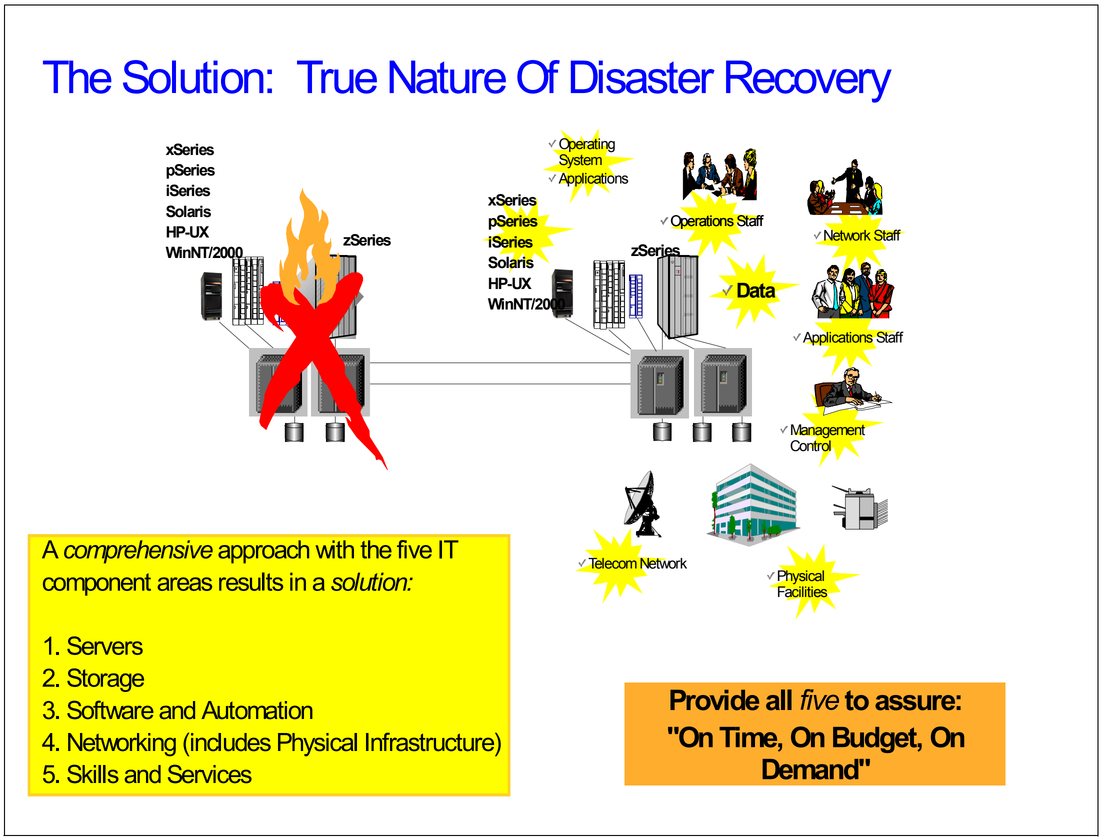



*Figure 8-1 Disaster recovery components*

In today's highly competitive e-business world, outages can have a devastating impact on a business — they can even mean its demise. Disaster Recovery is much more than just mirroring the disk data; rather (as Figure 8-1 illustrates), Disaster Recovery is a total business continuance solution comprising five major IT components:

- Servers
- Storage
- Software and automation
- Networking
- Services for integration

A disaster recovery implementation that only covers the storage component will leave the organization open to significant additional costs and time requirements if the other components are not covered.

## 8.1.2 Business objectives for Disaster Recovery

As shown in Figure 8-2, to design a cost-effective solution, you start determining the following objectives by application or business line:

- **Recovery Time Objective (RTO):** What is the business cost-justified elapsed time to recovery?
- **Recovery Point Objective (RPO):** When the RTO is met, what amount of data is needed to be recreated or is lost?
- **Network Recovery Objective (NRO):** How long does it take to switch the entire network over to the backup data center?

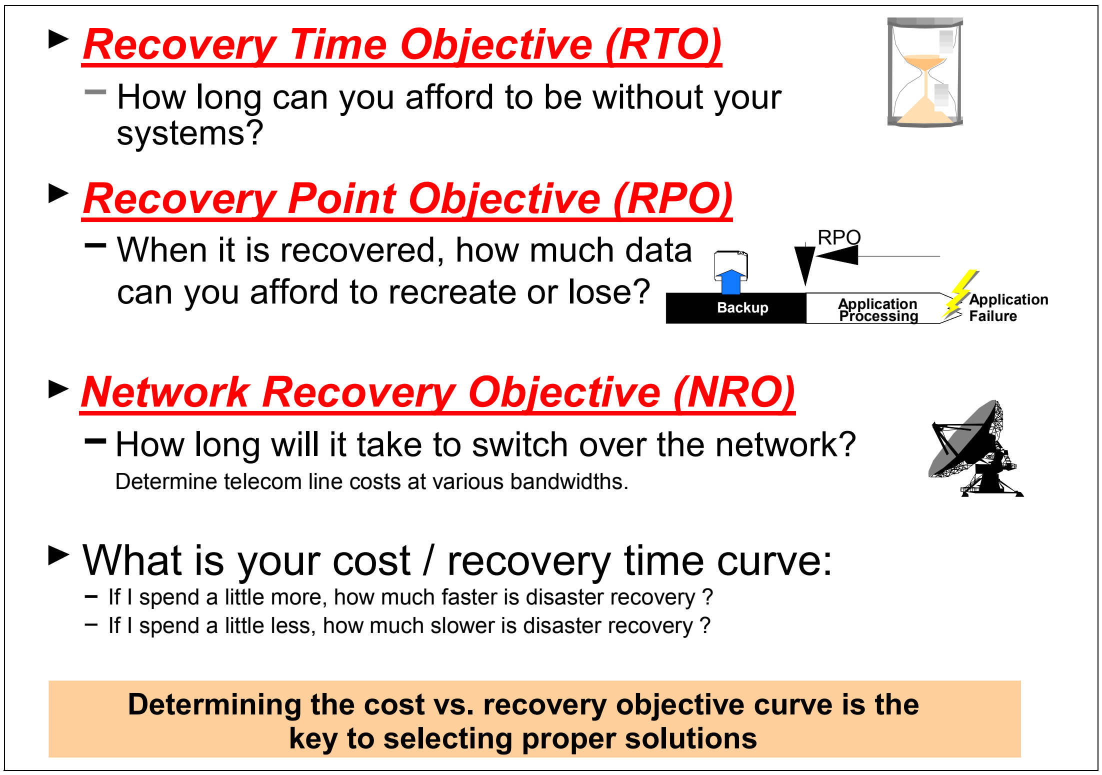



*Figure 8-2 Business objectives for Disaster Recovery*

The first two objectives (RTO and RPO) can often be balanced against each other to optimize the cost-benefit ratio. When the third objective (NRO) comes into play, networking issues will come into consideration. For example, there is no need to purchase a 30 minute RTO solution if the network provider requires 2 hours to switch the network.

### Planned versus unplanned outages

Planned outages are just as effective at removing service from the end users as unplanned outages — and they are much more frequent. Yet typically, disaster recovery solution cost justification is attempted based on the unplanned outage cost alone. So, a realistic return of investment analysis should always consider the planned outages effects.

### 8.1.3 Tiers of Disaster Recovery

When looking for a solution, you will always want to position it according to its effectiveness and the corresponding investment that must be done. This leads us to the tier considerations (refer to Figure 8-3).

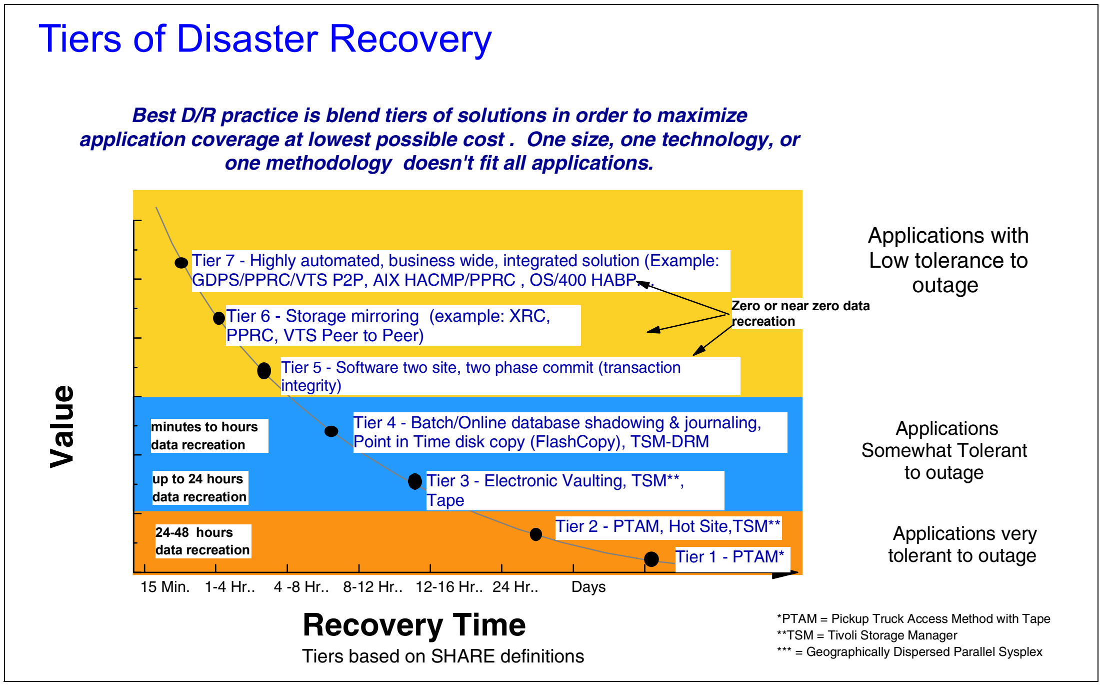



*Figure 8-3 Tiers for Disaster Recovery*

The chart in Figure 8-3 is a standard disaster recovery industry tier chart, showing the different solutions that can be implemented with their different recover-ability characteristics.

In general, there are three main bands. Within each band, there are tiers. The tiers vary as follows:

- Tier 0 (no recovery capability)
- Tier 2, 3 (tape intensive recovery methods)
- Tier 4 (disk mirroring or disk point-in-time facilities)
- Tier 5 (software/ database specific recovery)
- Tier 6 (near zero or zero data loss disk mirroring)
- And finally, tier 7, which is a completely automated self-managed take of servers, storage, software, and automation, networking all and integrated with services into the user's application

### 8.1.4 Hardware failure data consistency versus transaction data consistency

There are two levels of data consistency to be considered:

- **Hardware failure data consistency:** An example might be if the primary machine room were to experience an instantaneous power failure.
- **Transaction consistency:** This refers to application or database levels in which multiple transactions are interlocked, and if any operation is unsuccessful, then the application will back out the corresponding companion operations, thus preserving data consistency.

Therefore, software recovery on top of hardware recovery is essential and cannot be avoided.

## 8.2 Database data integrity and recovery

This section discusses the considerations for maintaining the write sequence integrity on the secondary volumes for the database systems, and the considerations for protecting them against a rolling disaster. Some of the recommendations are aimed for getting a restartable copy of the data, because a database *restart* is easier to do than a *recover*.

A restart requires that the state of all the data at the secondary site is consistent. This is achieved in a PPRC environment by eventually freezing all required secondary volumes when an error occurs. Then it must be decided whether it is more important to keep running the application (freeze and go) or making sure that the secondary site data keeps current and in sync with the primary site data — freeze and stop.

Adequately freezing the required secondary volumes — freeze and go — will ensure that the data is consistent from an application stand point, but any updates that are made from there on onto the primary volumes will not be reflected on the secondary site volumes — PPRC starts change recording for future re-sync. Therefore, in the event of a disaster, before the re-synchronization is done, some data will be lost.

Another way to ensure data consistency at the recovery site is to temporarily suspend the updates at the primary site — freeze and stop. Immediately after the freeze of the PPRC volume pairs is complete the secondary volumes will be holding a consistent point-in-time copy of the data — current with the application site data. DB2® provides this ability of suspending write I/Os, as explained in “DB2 suspend/resume log write activity” on page 414.

If an error occurs and no freeze of the required volumes is done, then the secondary volumes will keep on mirroring the primary updates — except for those involved in the error, which eventually will be suspended and thus not receiving any updates. This can leave the recovery data in a state that is not consistent from the application stand point, thus needing to restore previous backups and forward recover to the latest saved logs.

Using automation techniques, like the GDPS service offering with its policies, allows you to maintain data consistency and ensure a fast restart in the event of a disaster. GDPS is discussed in Appendix B, “Geographically Dispersed Parallel Sysplex (GDPS)” on page 359.

### 8.2.1 DB2

Data currency between sites can be achieved by putting all database volumes in PPRC pair relationships. DB2 databases, DB2 control information (BSDS, DB2 catalog, DB2 directory), and DB2 active logs should all be remote copied to the recovery site.

It must be carefully considered what would happen if a remote copy pair fails. When using PPRC, the effect of setting the critical attribute combined with Consistency Groups must be considered (see “Consistency Group and critical attribute combination” on page 89). This option can stop applications if all primary and secondary volumes cannot be kept in sync. This approach places more value on data currency than application availability.

When performing Disaster Recovery, recovery of the secondary volumes is performed first. Once the volumes at the recovery site are available for system use, then the same procedures should be followed for the recovery and restart of the application, as would be the case of a local failure at the application site.

#### Critical attribute enabled

When the critical option is enabled for a volume pair, the primary volume returns a permanent error to all I/O writes in case the secondary PPRC volume cannot be updated. This will depend on which option is configured on the ESS, whether light or heavy (see “Critical attribute" on page 88). For most applications, a permanent write error causes the application to terminate, but DB2's internal recovery allows processing to continue, and can lead to loss of availability.

If a write to a DB2 log volume receives a permanent write error, DB2 can redirect the I/O to another log data set. If a write to a tablespace volume receives such an error, DB2 adds the pages now in error to the Logical Page List (LPL).

To maintain DB2 integrity at the secondary system, all volumes that are used by DB2 must be in PPRC pairings. In an SMS environment you can avoid this problem by ensuring that all volumes in a particular SMS storage group are in PPRC pairings. All data will therefore continue to be reflected at the recovery site.

If a volume has been logged as needing recovery, some tables and partitions may have pages placed in LPL. If the error is due to all paths failing between a ESS pair, a growing number of pages of tablespaces could be added to LPL as subsequent writes encounter the PPRC permanent errors. As a result, the application primary site will require many DB2 recoveries, and the restart at the recovery site will require the same recoveries for all affected tablespaces. The DASD ERP issues an `IEA491E` message for each pair that suspends to the system initiating the I/O, and an `IEA494I` message is broadcast to any of the attached systems where the volume is online.

LPL recovery is initiated using a DB2 Start Database command. No image copy is needed, and the log is applied only from the point of the permanent I/O error to currency. It usually takes from seconds to minutes, depending on how many tablespaces and indexes have pages in LPL.

### Critical attribute not enabled

When the critical attribute is not enabled, the primary volume will not return an error condition to the I/O writes — even if the secondary volume cannot be successfully updated. PPRC will be aware of the problem, but the application will be able to proceed updating the primary volumes.

This error is not passed to DB2. The DASD ERP receives the write error from the storage controller. The sense information received identifies the error as a write failure to the secondary device, and a permanent error is not issued on the primary. The DASD ERP issues an `IEA491E` message for each pair that suspends to the system initiating the I/O, and an `IEA494I` message is broadcast to any of the attached systems to which the volume is online.

The ESS suspends the PPRC pairing. All subsequent writes to the primary should succeed, as the copy to the secondary is not attempted. ERP retries the failed write operation. The application therefore continues with minimal impact.

The secondary volume no longer reflects any updates and is subsequently out of synchronization with the primary and other secondary volumes.

### DB2 suspend/resume log write activity

There is a DB2 function that allows controlled suspension of application writes. After suspension of the updates to the DB2 subsystem, the database at the application site and at the recovery site will be at the same point of consistency. All pairs can be PPRC suspended or *frozen* at this time, and then resume the writes at the application site while doing a FlashCopy of the secondary volumes for a tertiary database safety copy. After the copy is taken, **`CESTPAIR`** can be used with **`MODE(RESYNC)`** to re-synchronize the primary and the secondary volumes.

This function allows users to recover the DB2 database to a point-in-time without having to experience an extended outage, or without having to stop or quiesce the DB2 subsystem.

The following command can be used to suspend writes on DB2:

```
SET LOG SUSPEND
```

When this command is issued, the following actions will take place:

- A system checkpoint is taken (in a non-data sharing environment).
- A log-write latch is obtained to prevent any further creation of log records.
- Any unwritten log buffers are written to DASD.
- The BSDS is updated with the high written RBA.
- DB2 sends a message **DSNJ372I**, informing that the update activity has been suspended.

The log-write latch being held prevents any further updates to the database until update activity is resumed by an operator command. The following command can be used to resume write activity on DB2:

```
SET LOG RESUME
```

This command releases the log-write latch, so the application write I/Os are allowed. DB2 also sends the message **DSNJ373I** informing that the update activity has been resumed.

PPRC and DB2 implementations are also discussed in 8.6.2, “Split mirror backup/recovery for a database” on page 450.

## 8.2.2 PPRC and IMS

IMS™ provides its own DASD volume and data set duplexing facilities to protect against DASD failure, through a separate optional feature. This feature is called IMS Remote Site Recovery (RSR).

IMS RSR allows you to resume online operations from a secondary site with minimal delay and minimum loss of data. RSR is IMS specific and covers only IMS data; however, it makes full use of IMS-specific functions and features.

If an application is IMS only, it is recommended to use IMS RSR to manage the database remote copy. If the installation requires a real-time copy of more than IMS data, then PPRC or XRC is recommended.

### *IMS Recovery with PPRC*

If all pairs are full duplex, then IMS restart can be done on the recovery site. IMS will manage the entire operation. Using the GDPS or RCMF service offerings or implementing an automation of the freeze function will ensure that the secondary volumes hold a consistent copy of the data. This is a simple approach to ensure that in the event of a disaster, the applications can be restarted in the shortest time.

If volumes are individually suspended, the DASD ERP issues the message `IEA491E`. If the paths were defined with the Consistency Group option, this message contains the timestamp of the failure and the suspension. Data sets can then be forward-recovered by using the data from DBRC of the IMS Database Manager. However, if volumes are suspended or pending, this will have a considerable impact on the duration of forward recovery. Timestamp based recovery is a complicated procedure in IMS; the IMS Recovery Saver utility program can be used to establish a timestamp recovery position.

## 8.3 PPRC-XD solutions

As already discussed, PPRC-XD is an excellent solution for:

- Remote data copy
- Remote data migration
- Off-site backup
- Transmission of inactive database logs
- Application disaster recovery solutions based on periodic point-in-time copies of the data, if the application tolerates the quiesce of its I/O activity for a short period.

This section illustrates examples of implementations for remote backup solutions, data migration solutions, and solutions involving the transmission of database logs.

### 8.3.1 PPRC-XD for remote data migration

Data migrations to a remote site can be done with PPRC Extended Distance in a much shorter time with minimum disruption to the production process. Figure 8-4 illustrates a very simple procedure for migration of data to a remote site.

The application site volumes to be migrated can be receiving updates from the application, while keeping a fuzzy copy at the remote site. At a designated cut-over point, the application writes are quiesced for a very short time (or the application is stopped) to let the secondaries catch-up with the normal PPRC-XD processing. As soon as the out-of-sync tracks for all the volumes is zero, all data has been copied to the remote location. Terminate volume pairs at the primary location, and restart the application using the set of volumes at the remote location.

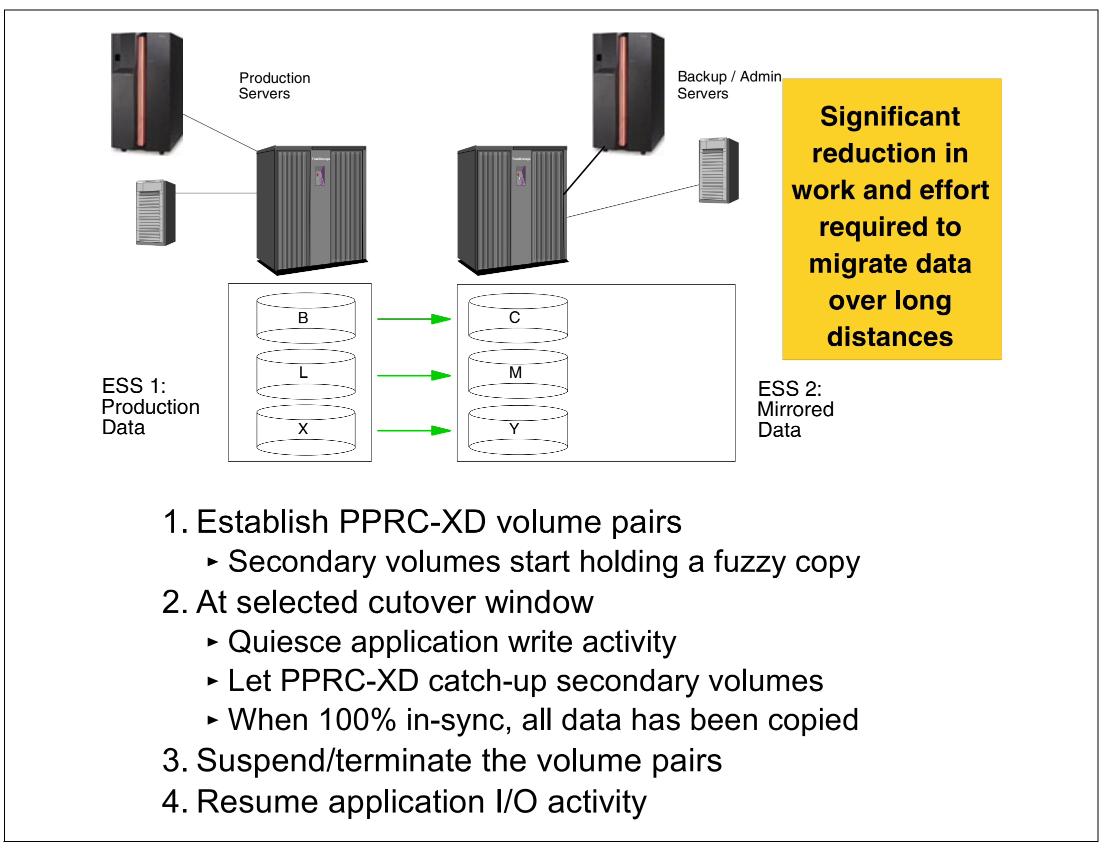



*Figure 8-4 Remote data migration*

## 8.3.2 Using tapes

A more sophisticated use of PPRC Extended Distance by complementing it with tapes can be implemented when needing to migrate a huge amount of data (for instance, 100 TB) from a production site to a remote site in an overnight schedule.

The idea is to capture a fuzzy copy of the data onto tapes after a synchronous mirroring (PPRC-SYNC) with the NOCOPY option has been established and immediately suspended. Tapes are sent to the remote site, and there they are restored on the terminated secondaries. Then a PPRC-XD relationship is established between the volume pairs, to send the updates that occurred since the tape captures were done while the primaries were suspended. An interesting characteristic of this solution is that it does not demand a huge telecommunication bandwidth, and has no application outage. You can use the following procedure to migrate data using PPRC-XD and tapes:

1. Establish paths between the primaries and secondaries.
2. Establish PPRC-SYNC pairs with the **Do not copy volume** and **Suspend PPRC after establish** options using the ESS Copy Services Web User Interface. This will build a PPRC relationship between the local and remote volumes, without doing the initial copy, and the PPRC copy pair will be suspended immediately.
3. At this time the primaries remain receiving updates (the application is active) and those updates are not copied onto the secondary volumes. However, the primary ESS is keeping record of these updates because the volumes are suspended.
4. Dump the primary volumes to tapes.
5. Deliver tapes to the remote location.
6. Terminate the secondary volumes.
7. Restore the tapes onto the secondary volumes.
8. Establish XD pairs between primary and secondary volumes. Now the pairs are re-synchronized, and only out-of-sync tracks are transferred to the secondary volumes.
9. When the volumes are 100% in-sync, the data has been migrated. Now terminate the pairs in order to have access to the secondary volumes.
10. Stop the application at the primary location.
11. Restart the application at the remote location using the migrated data.

## 8.3.3 Database logs transmission

Database recovery solutions based on switching active database logs, and transmitting the currently inactive log and the bootstrap data set (BSDS), can be efficiently implemented with PPRC Extended Distance as follows (see Figure 8-5):

1. The application dynamically switches from *log1* to *log2* at the production site. Now *log2* is the active log, while *log1* is inactive.
2. *Log1* is transferred to the backup site using PPRC-XD.
3. At the backup site, *log1* is applied to the shadow database.

For this solution, the logs can be defined in custom ESS volumes that PPRC will be mirroring.

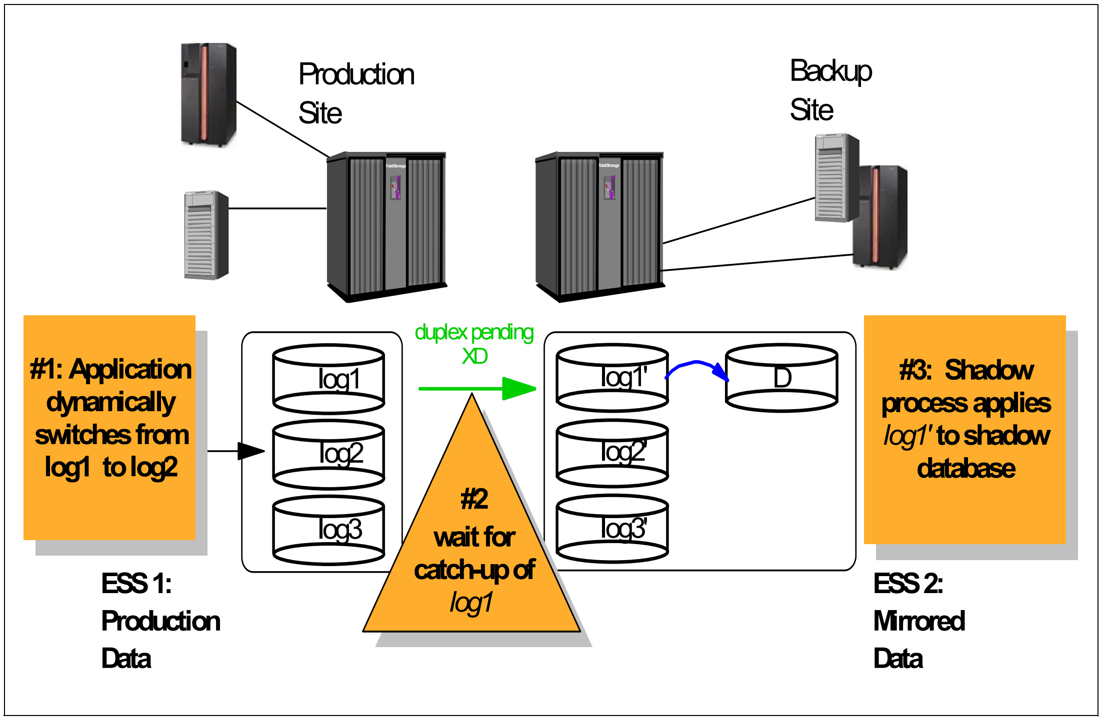



*Figure 8-5 Remote transmission of database logs*

## 8.3.4 Off-site backups

Either for the remote backup of volumes after your installation's batch window processing, or for remote point-in-time backups that your installation may issue at appropriate application checkpoints, PPRC-XD provides an efficient solution with less bandwidth demand.

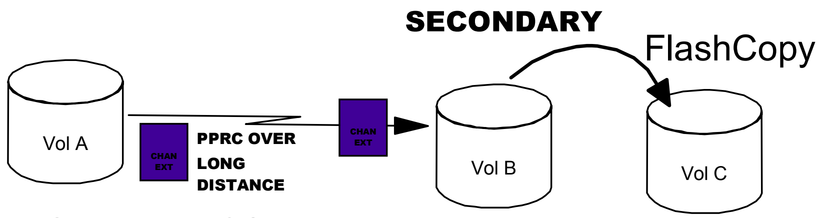
### End of batch window remote backup

The backups that are part of any production processing environment can now be sent to remote sites, and in a faster manner using PPRC-XD. The volumes holding the batch production output files can be mirrored with PPRC-XD while the processing jobs are updating them. As each job finishes, no more updates are applied, so in a very short time the job's set of volumes will automatically catch-up with PPRC-XD doing its normal processing. You can query these volumes to check when there are no more out-of-sync tracks, and then proceed with a FlashCopy of the secondary volumes in order to obtain a tertiary backup copy at the remote site. This use of PPRC-XD will allow you to substantially shrink your batch processing window.

### Point-in-time backups and split mirroring

For point-in-time backups and split mirroring implementations, PPRC Extended Distance minimizes the application quiesce interval — just for the short XD catch-up — as compared to other implementations. At the same time, PPRC-XD is a solution with minimum telecommunication bandwidth requirements, as compared to synchronous techniques. These characteristics allow for more frequent point-in-time backups and so the recovery point objective (RPO) improves substantially.

Figure 8-6 summarizes at a high level the basic setup for these types of implementations, when used over long distances.




- Sequence of Operation:
  - Establish PPRC Extended Distance pairs A-B volumes
  - Run in Extended Distance Mode until Consistent Copy needed
  - When Consistent Copy Needed, quiesce the application
  - Issue Establish PPRC pair command to *Go Synchronous*
    - Allow pairs to synchronize
  - Issue FREEZE command
  - Issue RUN Command when Consistency Group is Created
  - Issue FlashCopy B-C volumes (no impact to host)
  - When FlashCopy is initiated, Resume the applications
  - Reestablish PPRC Paths
  - Reestablish PPRC Pairs in Extended Distance Mode
  - Go back to Step 2.

*Figure 8-6 PPRC-XD for a split mirror implementation*

The point-in-time backups and split mirroring implementations remain the same, but with PPRC Extended Distance there are some additional benefits. Besides sending the copied data to distant remote locations, you will also make better use of your existing bandwidth.

## 8.3.5 Mixed environment (XD and SYNC)

In a remote mirroring environment, the performance of the application write operations is affected by the available bandwidth of the telecommunications infrastructure. This is more critical when the transmission is synchronous. Because of this, you might consider using these expensive bandwidth resources to mirror a mix of some more important volumes using the SYNC option, while the rest of the data is mirrored with the XD option. This consideration can be extended to the case where you need to cover more volumes, under the PPRC protected range, while not being able to increase your bandwidth.

### Initial establish for numerous volume set

When having to establish several copy pairs at once (in synchronous mode), you may find PPRC Extended Distance a very helpful aid.

If your PPRC setup involves several volume pairs that need to be established in synchronous mode at the same time, then when triggering this process, some volume pairs will become synchronized before others. In this situation, where some volumes are already synchronized while others are in the process of synchronization, the performance of the whole transition and of the application can be affected. If this is the case, you may find a good solution by initially establishing the PPRC volume pairs as extended distance pairs (XD option) and once the bulk transfer is near to completion, you transition the volume pairs to synchronous.

## 8.3.6 PPRC-XD for application recovery solutions

When PPRC-XD is used for application recovery solutions, automated procedures can be implemented in order to efficiently execute the steps involved in building global application consistency at the recovery site. The following events that occur when a consistent checkpoint must be taken can be automated:

- The quiesce of the application write activity and the succeeding resumption.
- Managing the go-to-SYNC transition window, where a quiesce of the application write activity is traded for a slowdown of the application I/O performance — if the ESSs are within the synchronous supported distance.

In z/OS and OS/390 environments, when automating, you will find the state change messages that the system issues when volumes transition from one PPRC state to a different one very useful. The **IEA494I** and **IEA491E** messages can be detected by automation routines, and used to suspend secondary PPRC operations.

The go-to-SYNC catch-up transition requires that you either temporarily quiesce all the write activity to the primary volumes, or do it when the application write activity is low. The recommended approach is to quiesce the application write activity, and wait until there are no more writes before the catch-up is triggered.

Consequently, when scheduling the quiesce of your application writes you will also be considering some lead time to build consistency on the recovery data. This planned outage can be minimized when properly automated, and the whole process can prove to be very efficient when recovering an application after its outage.

When making your secondary volumes fully synchronous with their respective primaries, the commanded go-to-SYNC approach is not the only one you can use. As already mentioned, you can just quiesce the application writes and let PPRC-XD normal operation to do the catch-up, while you query PPRC to check for no more out-of-sync tracks.

**Note:** The following sections present possible setups for getting a consistent point-in-time copy of your data at the recovery site. Nevertheless, keep in mind that the optimum setup (variations of steps, sequence and timings) will be application dependent, and you will be able to determine it by testing when doing your actual implementation.

### Initial catch-up of the volumes

This example illustrates a sequence of steps to obtain a consistent, point-in-time secondary copy when commencing with the catch-up of the application set of volumes. The result will be a point-in-time copy with global consistency (from the application perspective) across all the secondary volumes.

### *Go-to-SYNC and suspend*

Figure 8-7 shows what happens during a consistent point-in-time copy.

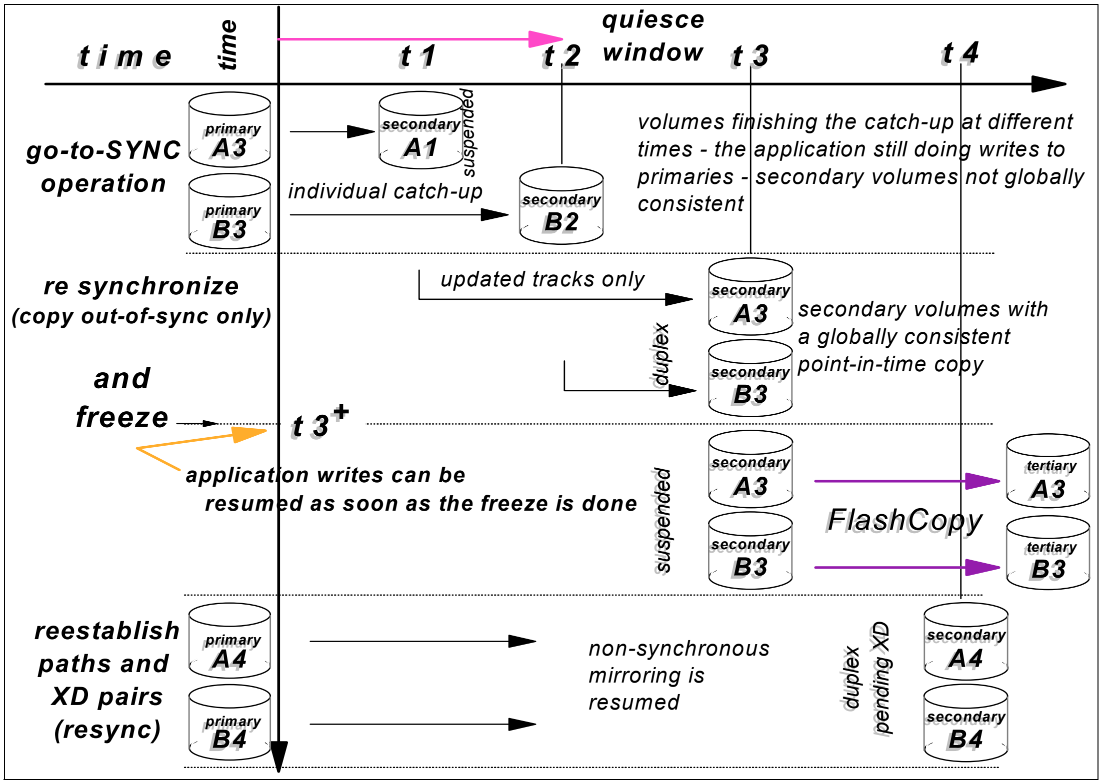



*Figure 8-7 Consistent point-in-time copy - Starting with a go-to-SYNC*

Please refer to Figure 8-7 to follow the explanation presented in this section. As the application is running and making updates onto the primary volumes, you may commence the catch-up by first doing a *go-to-SYNC and then suspend* the set of volumes where the application is updating data. This step should be done during a very low (or none) application write activity window.

In order to suspend immediately upon reaching the duplex state to avoid any synchronous update over the long distance, you can use the option **Suspend PPRC after establish** if you are working with the Web User Interface.

For TSO users, the *go-to-SYNC and suspend* operation should be automated. Because the volumes will be transitioning — from the duplex pending XD state to the duplex state — this change of status will be notified to the host with the `IEA494I` message. These messages can be intercepted and acted upon by automation routines to suspend the volumes as they reach the duplex state.

Upon finishing this initial *go-to-SYNC and suspend* step, you have a set of suspended secondary volumes, which individually have a consistent copy of their primary suspended counterparts. But as you can see from Figure 8-7 the point-in-time is not the same *across* all the secondary volumes. The application was doing write updates to the primary volumes and some of these updates — as the volumes were individually finishing their catch-up and reaching the suspended state at different moments — were not mirrored, but recorded by the primary ESS for later re-synchronization.

### *Quiesce application writes*

In order to reach global consistency across the secondary volumes, you should quiesce the application writes. This quiesce will allow for a succeeding re-synchronization of the volumes, but this time reaching a point-in-time with global consistency.

The need of a quiesce, in order to build consistency at the recovery site, is a trade-off against the write performance penalty inherent to a long distance synchronous implementation, which does not require the quiesce.

> **Note:** At PPRC synchronous supported distances, this quiesce could be skipped. Refer to "Point-in-time within synchronous range" on page 423 for further explanation.

### *Build consistent point-in-time and restart application writes*

With the application not doing any writes, you now re-synchronize the suspended pairs (`copy out-of-sync tracks only` / `RESYNC`).

One alternative could be to re-synchronize the pairs, and then suspend them again immediately upon reaching the duplex state. This operation can be implemented by creating a major task that groups several individual suspend tasks, or by creating a TSO command list that executes several individual suspend commands — one for each of the volume pairs in consideration.

A more efficient way of doing this is to re-synchronize the pairs, and once they reach the duplex state, then freeze the respective LSS. This will simultaneously suspend all the volume pairs within the affected LSS, and also terminate the paths between the LSSs.

Either of the two alternatives leaves the secondary volumes with a globally consistent point-in-time copy of the primary volumes, and also both alternatives leave the volumes suspended. This suspended state enables you to restart the application write activity on the primary volumes, while you FlashCopy the secondary volumes that now have a consistent point-in-time copy.

### *Restart application writes*

The application write activity can be restarted as soon as the freeze is done. These updates will not be propagated to the secondary volumes — as volume pairs are suspended and LSS paths are discontinued. In the meanwhile, PPRC will keep track of the updated tracks on the primary volumes to be able to do an incremental re-synchronization when the XD pairs are later re-established.

### *FlashCopy consistent checkpoint data*

Once the FlashCopy relationship is established, you can bring the primary and secondary volumes out of the suspended state. Bringing them into the duplex-pending XD state — an incremental re-synchronization (copy out-of-sync only) — will resume the mirroring on the secondary volumes.

Before reestablishing the volume pairs' XD relationships, you must redefine the LSS paths that were terminated by the freeze operation.

Now the procedure is complete, the consistent point-in-time copy is on the tertiary volumes, and we are ready for the next checkpoint.

## Point-in-time within synchronous range

At Synchronous PPRC supported distances, the previous procedure can be done differently, skipping the quiesce and the re-synchronization steps. If you temporarily trade-off some application write performance, then the procedure would be limited to:

1. Trigger a catch-up commanding PPRC-XD to go-to-SYNC but *not suspending*. This should be done when the application write activity is very low, and volumes are at least 90% in-sync; the catch-up will be short and the application write performance impact less onerous.
2. Detect when *all* volumes are in duplex state — secondary copies become globally consistent. Queries and detecting `IEA494I` messages will give this information.
3. Upon the volumes pairs reaching the duplex state, do a freeze.
4. Now you have a secondary consistent point-in-time copy, that you can FlashCopy. Also, at this moment, the application is already relieved of the write penalty because volumes are suspended because of the freeze.
5. Finally, you reestablish the paths, reestablish the XD pairs, thus the non-synchronous mirroring starts again until the next catch-up.

For this particular implementation, the distance must be less than 103 km for ESCON or 300 km if using FCP.

## Initial quiesce of the application

As already mentioned, you can set up different procedures in order to build global consistency in the data at the recovery site. At the time of actual implementation at your installation, you will be testing to determine which is the most efficient and less disruptive implementation.

The example presented in Figure 8-8 illustrates an alternate procedure where you first start by waiting for the application writes to quiesce completely, and then let PPRC-XD continue on its own to complete the catch-up of the volumes.

**Tip:** There will be no `IEA494I` messages when the volumes reach 100% synchronization because you have not requested any volume state change.

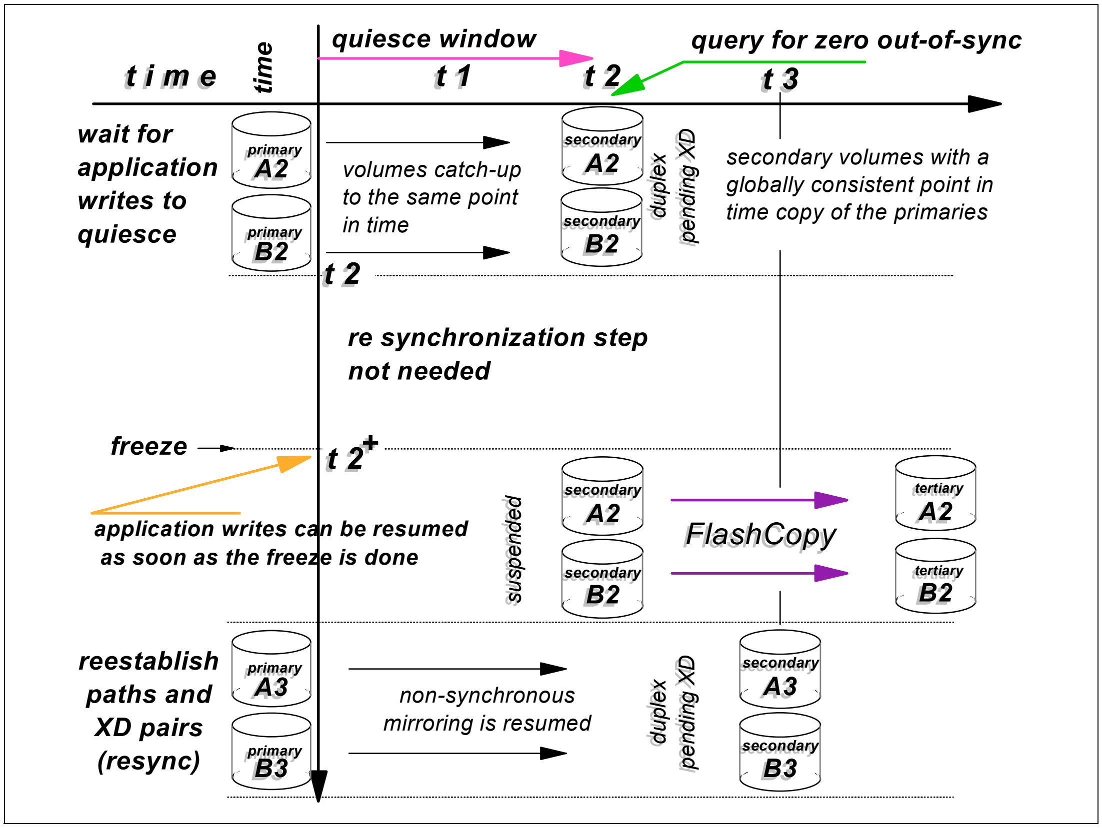



*Figure 8-8 Consistent point-in-time copy - Waiting for application writes to quiesce*

The more granular PPRC queries will allow you to determine with more accuracy, when there are no more out-of-sync tracks. At this moment, your secondary volumes hold a point-in-time copy of the primary application volumes with global consistency.

As you see in this example, because PPRC-XD puts the volumes in a 100% synchronization point, there is no need to re-synchronize the pairs, as in the previous example. You proceed to freeze the LSSs, which still have the volumes in duplex pending XD state. Also, upon completing the freeze, you proceed to do a FlashCopy onto the tertiary volumes.

As soon as the FlashCopy relationship are established, you can now re-initiate the XD mirroring onto the secondary volumes. To do so, you reestablish the paths and the pairs.

## 8.4 PPRC planned outage procedures

The required downtime for systems maintenance is becoming less and less tolerable. In a global economy, companies require systems with a *continuous operation* capability. PPRC provides a solution for continued operation using an alternate site, during planned system outages such as major hardware changes or electrical maintenance.

Failover from a primary to a secondary site and failback from the secondary to the primary site can be managed by any one of the following methods:

- The TSO PPRC commands procedure
- The ESS Copy Services WUI procedure using the PPRC Failover and Failback modes.
- Geographically Dispersed Parallel Sysplex (GDPS) implementation.

This section describes examples of failover and failback planned outage procedures for the TSO and the Web interface implementations. The Web implementation takes advantage of the benefits that the PPRC Failover and Failback modes provide for this situations. GDPS is not discussed in this section, as it is covered in detail in Appendix B, “Geographically Dispersed Parallel Sysplex (GDPS)” on page 359.

### 8.4.1 Planned outage procedure with TSO commands

This section describes a procedure that can be used to switch from the production site to the recovery site for a planned event. This procedure is implemented using TSO commands for controlling the PPRC functions.

The procedure, which is illustrated in Figure 8-9 and Figure 8-10, has the following sequence of steps:

1. Start with the volume pairs in full duplex, that is the recovery site (site B) volumes are 100% in sync with the production site (site A) volumes. The system at site B is up and running.
2. At site A, quiesce the system that is going to be moved, so that the updates to the primary volumes are stopped.
3. Delete the PPRC pairs, so the volumes change from the duplex state to the simplex state.
4. Delete the PPRC paths to allow a PPRC direction reversal in the next step.
5. At site B, establish PPRC paths from site B to site A.
6. Use the **NOCOPY** parameter of **CESTPAIR** command to establish PPRC pairs relationships from site B to site A.

   Specifying **NOCOPY** results in that only those tracks on the primary volume that are updated after this command is issued, are to be copied to the secondary volume. This mode is specified only if the volumes are exact copies of each other already. With this option the PPRC relationship is established without having to copy data.
7. Suspend the PPRC volume pairs. PPRC starts change recording on site B to keep record of the tracks that will be updated while the applications run at site B — this will allow a fast re-synchronization later, when doing the failback to site A.
8. Shut down the system at site A for the scheduled maintenance or upgrade activity, and restart the system at site B, also restarting the production applications.

When everything is ready to return to the application site, follow the steps detailed below:

1. At site B, use the **RESYNC** parameter of the **CESTPAIR** command to establish the pairs. This will copy only all the updated tracks back to the volumes in site A, and leave the volume pairs 100% in sync in duplex state.

   Specifying **RESYNC** tells PPRC to re-establish a suspended copy pair. Before issuing this option, verify that both the primary and secondary volumes are in a suspended state, or the secondary is in a simplex state, and the primary is suspended. PPRC only copies the tracks that were updated during the period of suspension.
2. At site B, quiesce the applications so the update activity on the volumes is stopped.
3. Delete the PPRC pairs, so the volumes change from the duplex state to the simplex state.
4. Delete the PPRC paths to allow a PPRC direction reversal in the next step.
5. At site A: establish PPRC paths from site A to site B.
6. Use the **NOCOPY** parameter of **CESTPAIR** to establish PPRC pairs from site A (primaries) to site B (secondaries). This last step has reversed the direction of the PPRC copy back to the original configuration where the production is run at site A, and the applications can now be restarted at site A.

It is also possible to use P/DAS in this procedure. For detailed information on using P/DAS, refer to following publication: *z/OS DFSMS Advanced Copy Services,* SC35-0428.

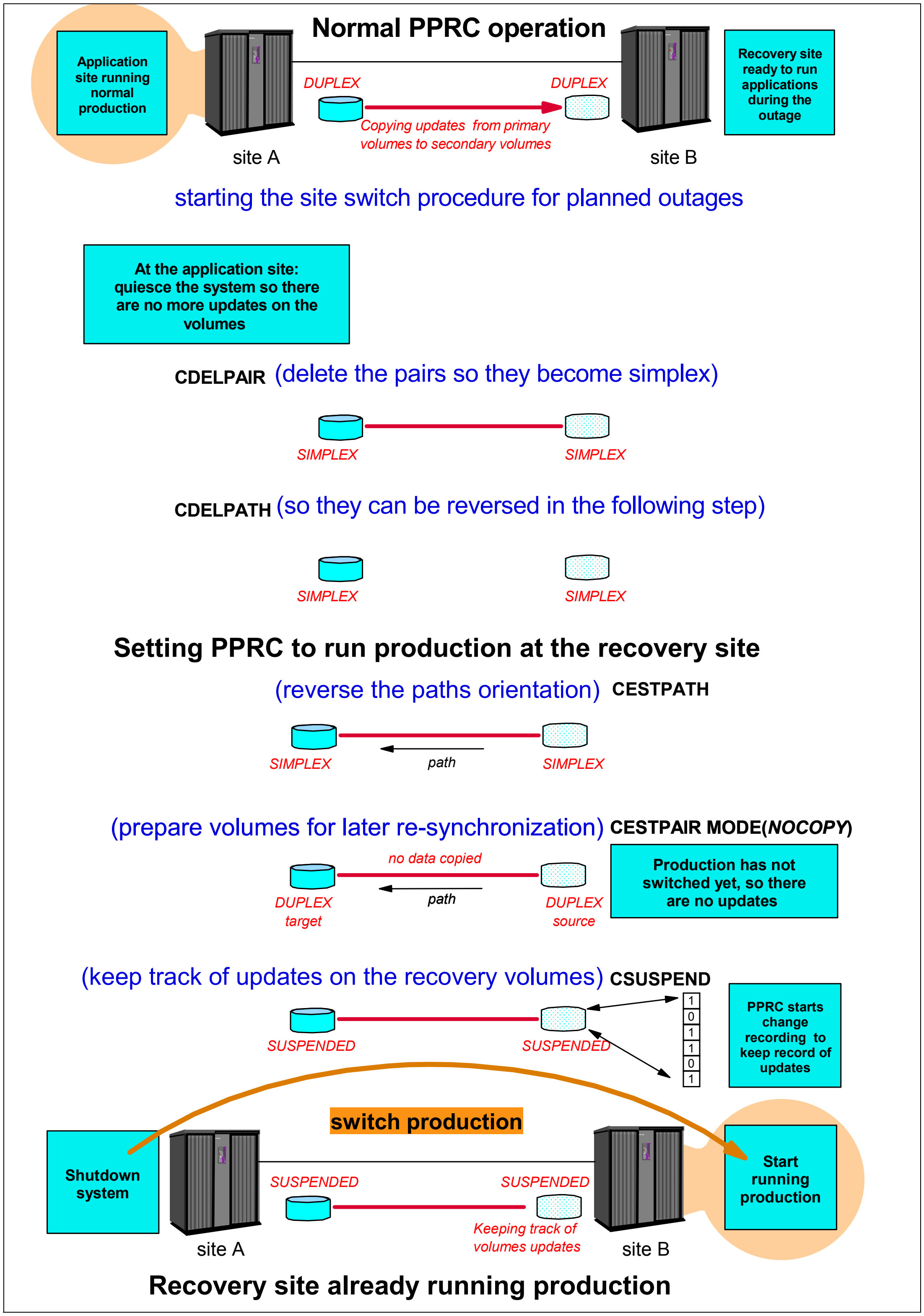



*Figure 8-9 Planned outage - switching to the recovery site*

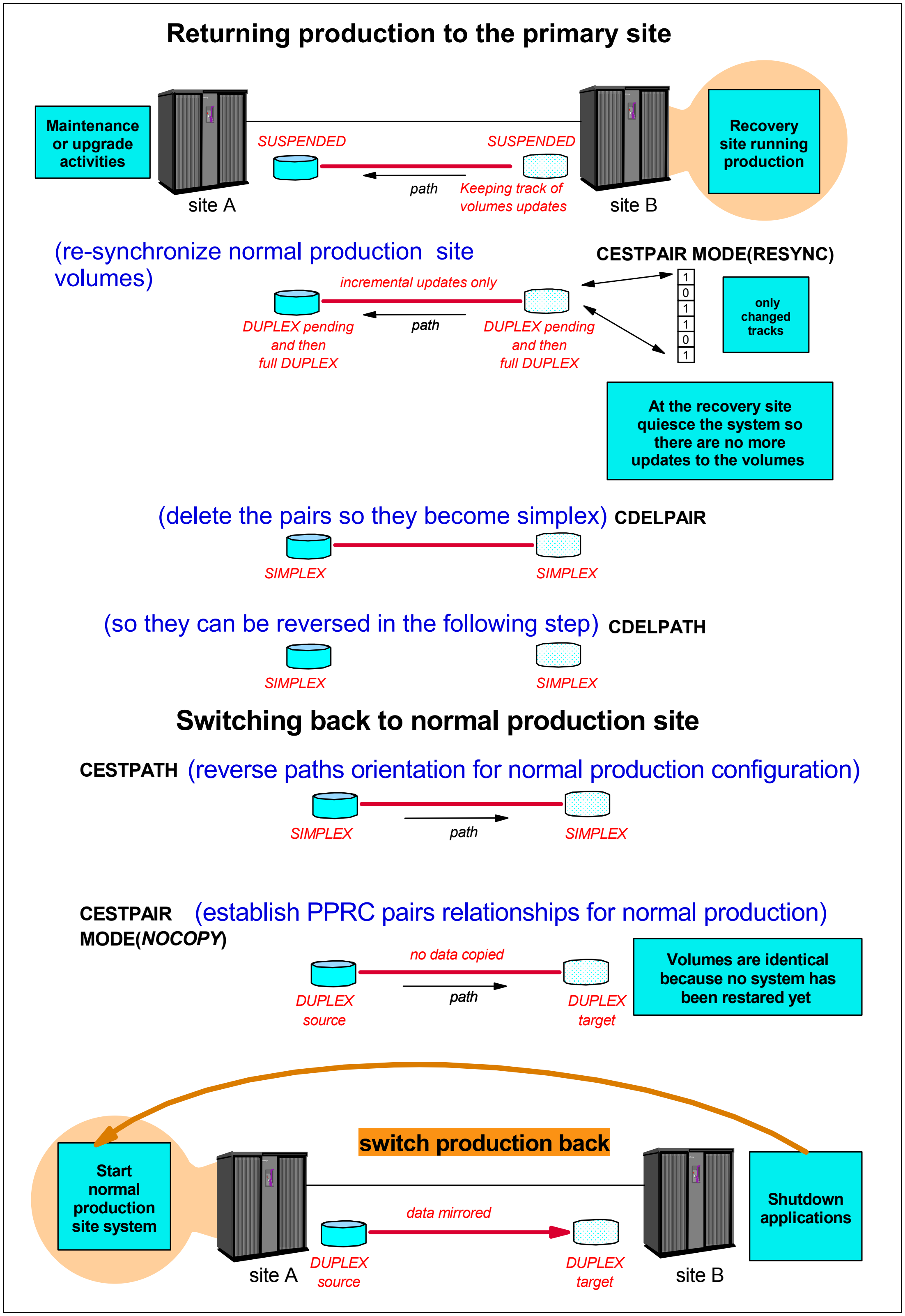



*Figure 8-10 Planned outage - re-synchronizing to return to normal production*

Automated procedures can be used to generate batch jobs to switch sites with only a short disruption to service.

## 8.4.2 Planned outage procedure using PPRC Failover and Failback mode

This section illustrates a complete failover/failback procedure for a planned outage using the ESS Copy Services Web User Interface (WUI), and taking advantage of the PPRC Failover and Failback modes (see Figure 8-11).

As you will realize from this example, the PPRC Failover and Failback modes help in reducing the time required to synchronize PPRC volumes after switching between the application and the recovery sites. These functions are invoked using the ESS Copy Services WUI or the ICKDSF `PPRCOPY ESTPAIR` command. Failover and Failback functions are described in 3.5.1, "PPRC Failover and Failback modes" on page 91.

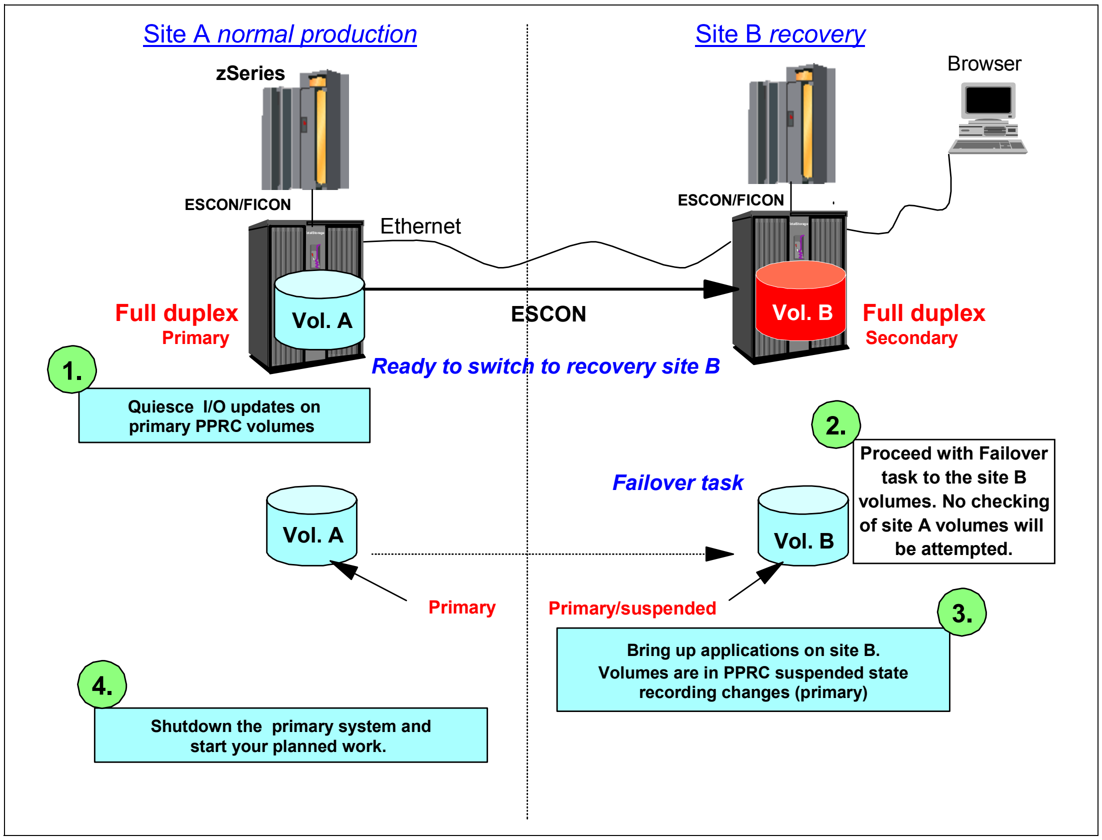



*Figure 8-11 Ready to switch to recovery site - Failover task*

Figure 8-11 illustrates the failover procedure required to do the initial switch from the normal production site (site A) to the recovery site (site B), and the subsequent shutdown of the application site (site A). The steps are:

1. At site A, verify that all volumes are in full duplex state. Then stop the applications, so the primary volumes have no more I/O updates.
2. At site B, invoke an already saved task that invokes the PPRC Failover Mode function to be used on site B volumes.

   Figure 8-12 shows the copy option that should be selected when creating the task, which establishes pairs from B to A. The option **PPRC Failover** will make the volumes at site B transition from PPRC secondary to PPRC primary when the task is executed. It also causes the pairs to be suspended. The volumes at site B will then start recording changes for the entire period while in Failover mode. The original state of the volumes at site A will be preserved as it was when the failover was initiated.

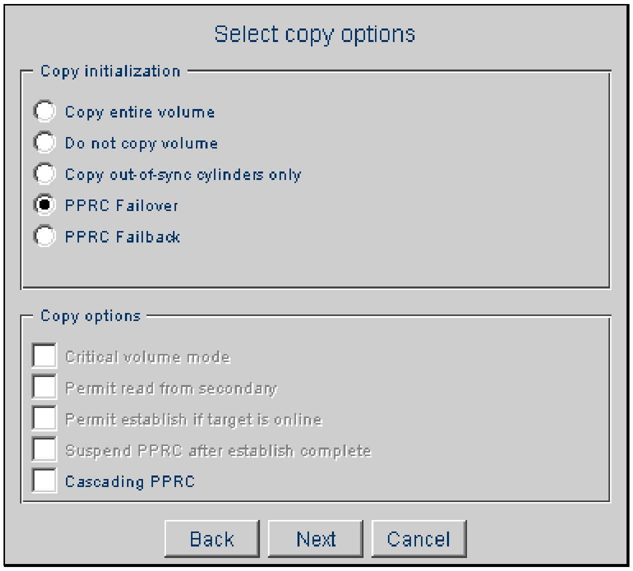



*Figure 8-12 Failover task creation - Copy options*

3. When the PPRC failover is successfully completed, shut down the system at site A and start your applications at site B.

Figure 8-13 illustrates the first part of the failback procedure required to start switching back to the application site (site A).

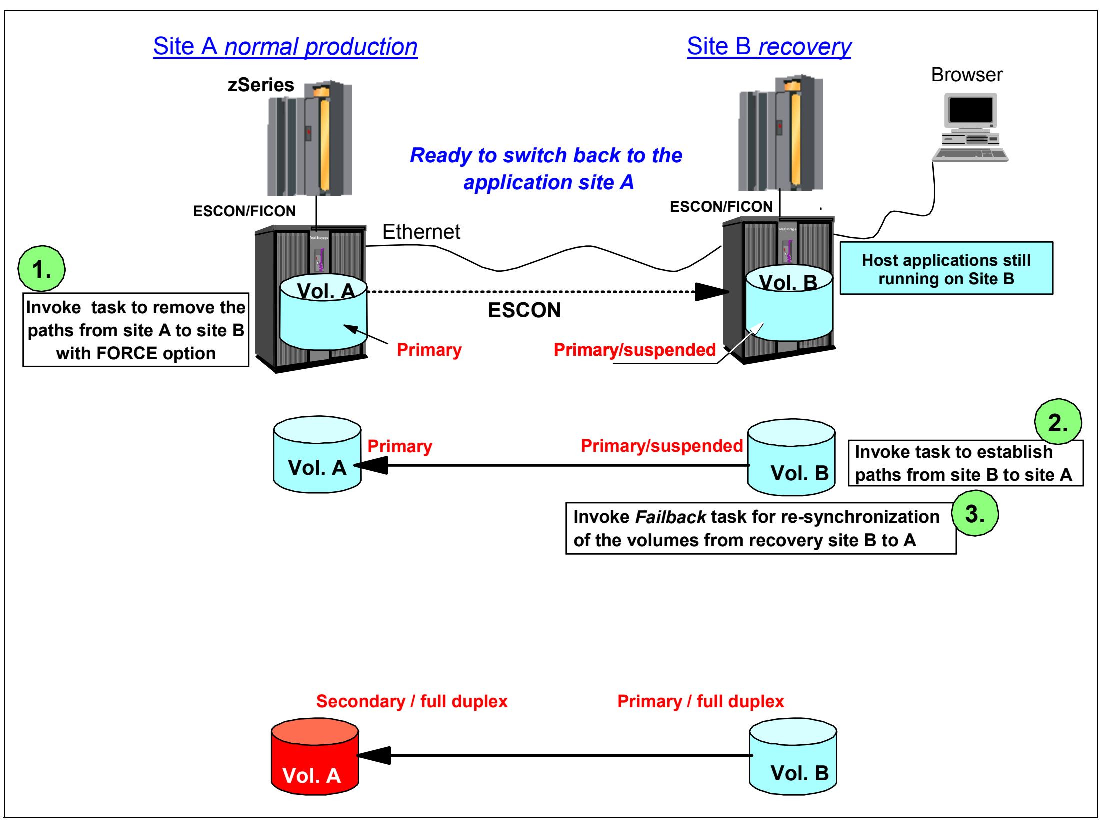



*Figure 8-13 Switching back to the application site (site A) - Failback task*

1. From site A, execute the tasks for removing the paths from site A to site B, forcing the deletion of the paths even if pairs exist. Figure 8-14 shows the option selected for the removal of the paths when the task was created. The state of the PPRC volume pairs will not be modified after the execution of this task.

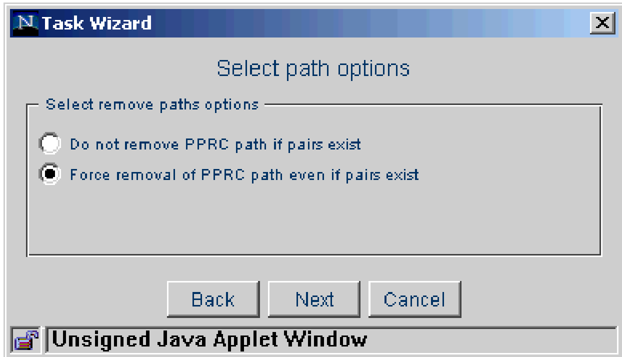



*Figure 8-14 Remove paths task creation - path options*

2. From site B, execute the tasks for establishing paths from site B to site A. Figure 8-15 shows the options selected for establishing the paths when the task was created.

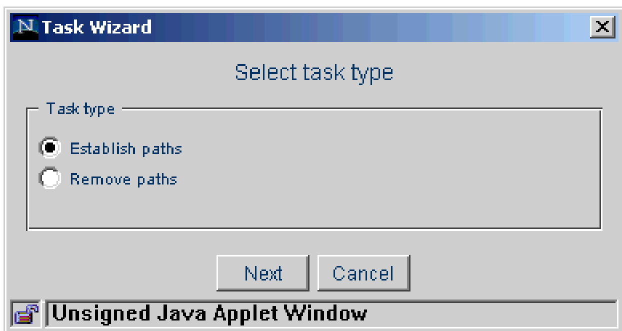



*Figure 8-15 Establish paths task creation*

3. At site B, execute the task that invokes the PPRC Failback Mode function to be used on site B volumes.

   Figure 8-16 shows the copy option that should be selected when creating the task, which establishes pairs from B to A. The option **PPRC Failback** will re-synchronize the volumes from B back to A, and leave the volumes at site A in-sync and in secondary duplex state. Since the volumes at site B were suspended, only the changed tracks will be copied during the re-synchronization from B to A. This requires that the volumes at site A must be in the correct state (duplex), and have not been subject to any updates since the beginning of this procedure.

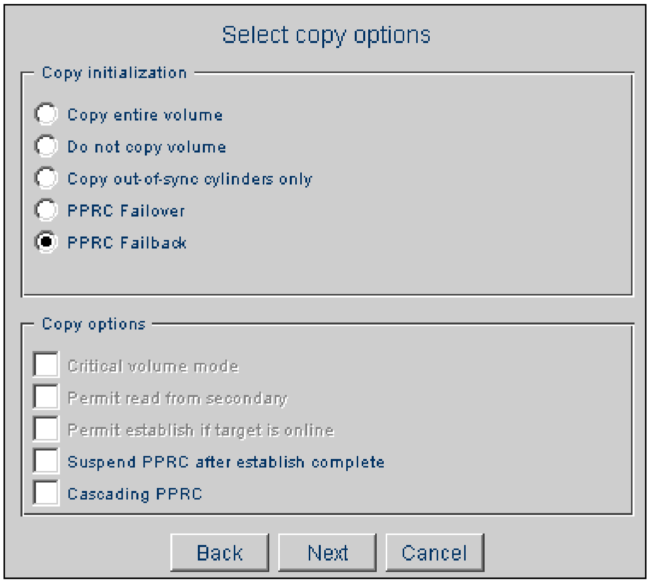



*Figure 8-16 Failback task example*

Figure 8-17 illustrates the final part of the procedure to failback to the application site (site A), and restart normal production processing.

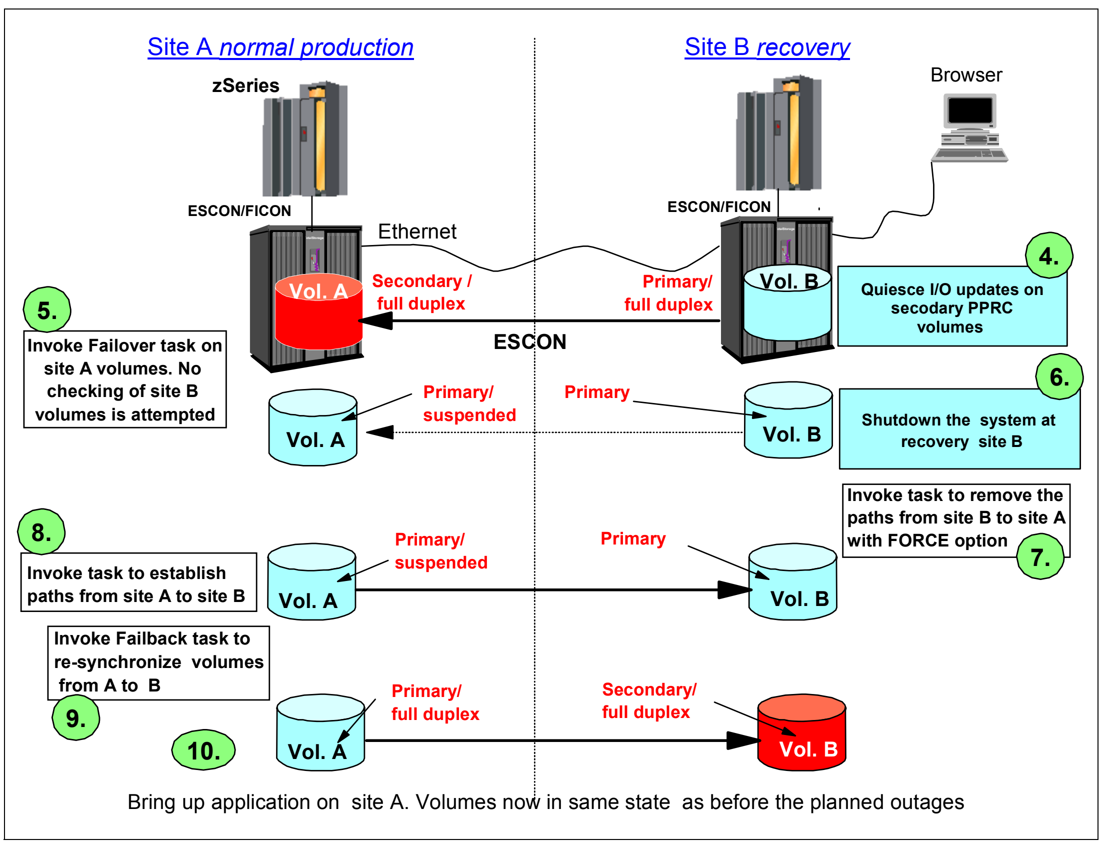



*Figure 8-17 Switching back to the application site (site A) - Failover and failback tasks*

4. At site B, check that the PPRC failback procedure has completed, and all the volumes are in duplex state. Stop the applications on site B, so the volumes at site B are not subject to I/O updates.

5. At site A, invoke the task that invokes the PPRC Failover Mode function to be used on site A volumes.

   The option **PPRC Failover** will make the volumes at site A transition from PPRC secondary to PPRC primary when the task is executed. It also causes the pairs to be suspended. The volumes at site A will then start recording changes for the entire period while in Failover mode. The original state of the volumes at site B will be preserved as it was when the Failover was initiated.

6. When the PPRC Failover is successfully completed, shut down the site B system and start applications at site A.
7. From site B execute the tasks for removing the paths from site B to site A, forcing the deletion of the paths even if pairs exist (see task options in Figure 8-14). The PPRC state of the volume pairs will not be modified after the execution of this task.
8. From site A execute the tasks for establishing paths from site A to site B (see task options in Figure 8-15).
9. At site A execute the task that invokes the PPRC Failback Mode function to be used on the site A volumes.

   Figure 8-16 shows the copy options that should be selected when creating the task, which establishes pairs from A to B. The option **PPRC Failback** will re-synchronize the volumes from A back to B, and leave the volumes at site B in-sync and in secondary duplex state. Since the volumes at site A were suspended, only the changed tracks will be copied during the re-synchronization from A to B. This requires that the volumes at site B must be in the correct state (duplex), and have not been subject to any updates in the meanwhile.

Now the PPRC environment is in the same condition as before the planned outage procedure started. We recommend that you create and test a procedure like this one, so you are prepared to minimize the disruptions when you carry out a planned outage.

## 8.5 Asynchronous Cascading PPRC solutions

This section describes two basic Asynchronous Cascading PPRC implementations:

- Asynchronous Cascading PPRC implementation for three sites
- Asynchronous Cascading PPRC implementation for two sites

Various factors determine which solution is more appropriate. Indeed, one consideration is the recovery point objective (RPO) that is tolerable for your organization.

### 8.5.1 Three-site implementation

In the three-site implementation, the ESSs are located in three different sites, with the remote site at a distance such that a regional disaster at the local site should not affect it (see Figure 8-18).

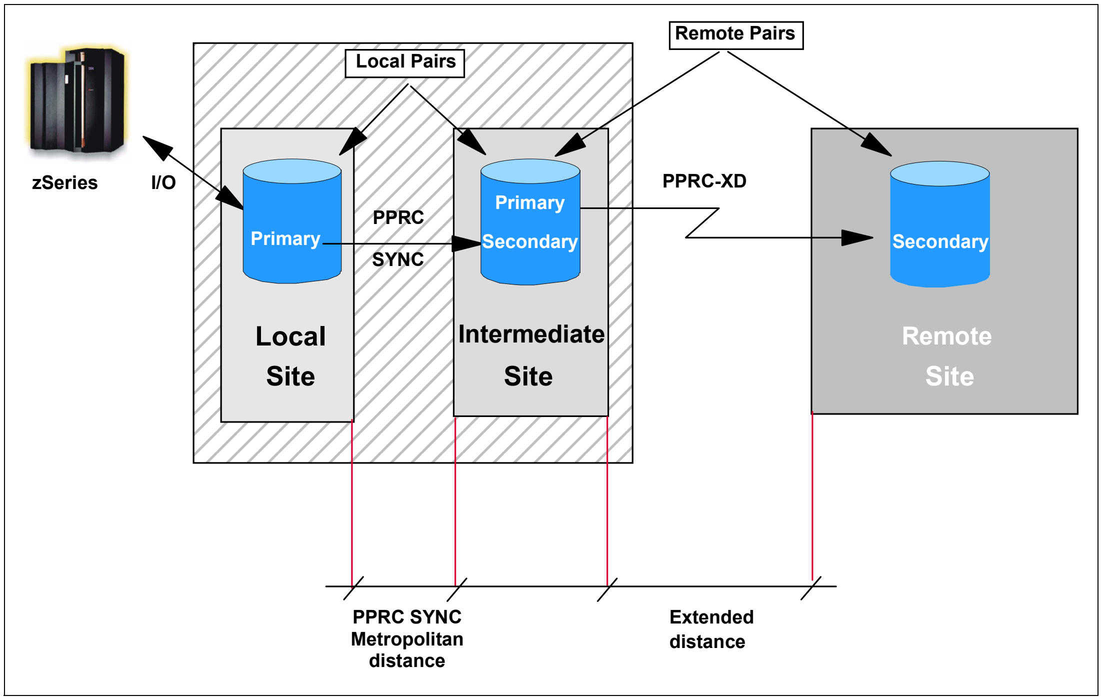



*Figure 8-18 Three-site Asynchronous Cascading PPRC configuration*

The intermediate site should be in a safe environment separate from the primary site, with its own power supplies, thus reducing the possibilities for unplanned outages affecting both the local and the intermediate sites at the same time.

In this three-site implementation, if an outage occurs at the local site, the intermediate site volumes will be holding a point-in-time copy of the data at the time of the outage. PPRC-XD will do a catch-up, and so after a while the remote site volumes will be 100% in synchronization with the intermediate site volumes. This way the remote site volumes also will be holding the point-in-time copy of the data at the time that the outage occurred at the local primary site.

If the outage occurs at the intermediate site, the data at the local site is not affected at all. Applications at the local site continue to run as normal. In the meantime, until the intermediate site is recovered, there is only one copy of data.

Finally, if the remote site is affected by an outage the local and the intermediate sites are not affected. The applications at the local site continue to run, while the intermediate site mirrors the local site data.

## 8.5.2 Two-site implementation

The two-site Asynchronous Cascading PPRC implementation consists of the local and remote sites only (see Figure 8-19).

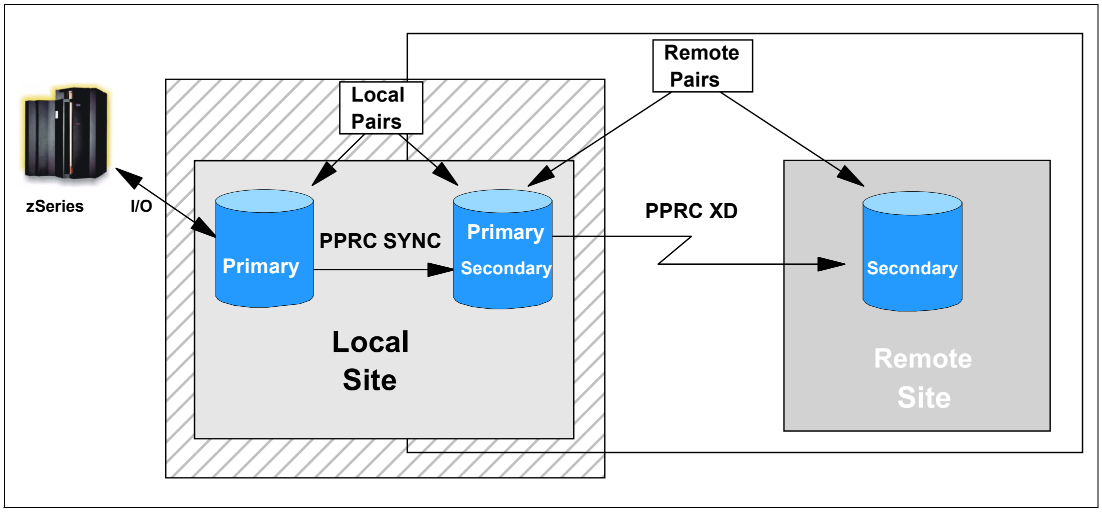



*Figure 8-19 Two site Asynchronous Cascading PPRC configuration*

The local PPRC pairs can be located in the same ESS, or in separate ESSs, however, at the same site location. This is a more economical solution in case you cannot afford an intermediate site or even additional ESS disk subsystems. Still, in the case of a complete disaster at the local site that also affects the intermediate site, the RPO will depend on how recent a consistent point-in-time copy was taken at the remote site.

## 8.5.3 Failover/Failback for Asynchronous Cascading PPRC

The exploitation of PPRC Failover and Failback modes helps reduce the time required to synchronize PPRC volumes after switching between sites during a planned or unplanned outage. Starting with LIC level 2.3.0, the PPRC Failover and Failback processing has been enhanced to support Asynchronous Cascading PPRC configuration’s Failover and Failback.

> **Note:** The discussion in this section can be complemented with the example presented in 3.5.1, “PPRC Failover and Failback modes” on page 91, which shows the actual ICKDSF commands, parameters, and Web User Interface panels used to execute the failover and the failback.

In our discussion we show how PPRC Failover and Failback modes can be used to swap application I/O to site B after I/O to site A has ceased, for example, due to a planned outage for site A maintenance.

For both the zSeries and the open system environments, PPRC Failover and Failback modes for Asynchronous Cascading PPRC can be invoked using the ESS Copy Services Web User Interface (WUI) and — provided a supported server — the ESS Copy Services Command Line Interface (CLI) via a saved task. For the zSeries environment the ICKDSF utility can also be used.

The following sections describe the process of switching application I/O to site B and then, when site A is available again, switching application I/O back to site A while maintaining the PPRC-XD session between site B and site C.

## Switching to site B

We begin the discussion with the assumption that I/O to site A has ceased due to a planned outage, and we start by showing how Failover can be used to switch to site B.

### *PPRC Failover to site B volumes*

The first step in switching application I/O from site A to site B is to do a PPRC Failover at site B. This can be done via the ESS Copy Services WUI or, if it is a zSeries environment, using the ICKDSF PPRCOPY ESTPAIR command. This procedure is similar to the one described in 8.4.2, "Planned outage procedure using PPRC Failover and Failback mode" on page 428 for a non-cascading environment.

The ESS performs the following operations when specifying the PPRC Failover option on site B volumes:

1. Detects that the site B volume is a cascading volume.
2. Suspends the A → B session but does not reverse the direction.
3. Activates Change Recording (CR) for the site B volumes.
4. Enables a new cascading volume flag, "Enable Secondary Host writes."

This is different from the PPRC Failover processing in a non-cascading environment, which would make B a primary in a B → A PPRC session. This cannot be done for a cascading PPRC volume because the B volume cannot be a primary for B → A and B → C at the same time. The B volume's Out-Of-Sync (OOS) information is required to maintain the PPRC-XD B → C sessions and Change Recording (CR) information is required to track updates to the B volumes for the imminent B → A sessions.

The state of the Asynchronous Cascading PPRC environment after the PPRC Failover processing has completed is shown in Figure 8-20.

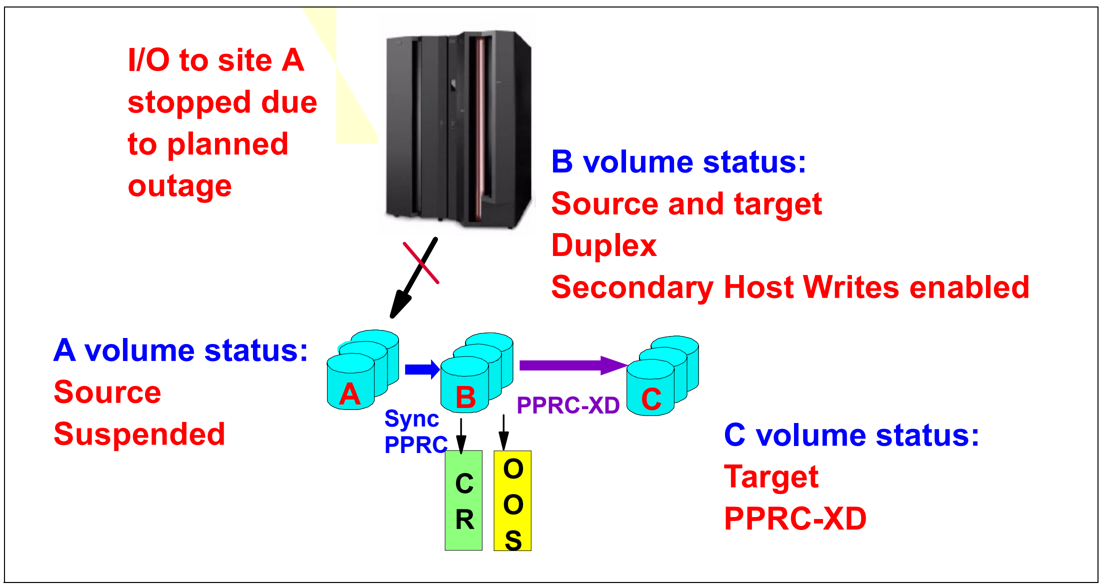



*Figure 8-20 Volume states after the PPRC Failover to site B volumes*

### *Swap I/O to site B volumes*

Host I/O can now be enabled to the site B volumes. The CR and OOS information is tracked for updates to site B volumes because the PPRC-XD session B → C is still active. The go-to-SYNC operation can still be used to synchronize the site C volumes and FlashCopy can be used to create consistent copies of the site C volumes during this phase. This can continue until site A is ready to become the primary site again for application I/O. Figure 8-21 shows the status after application I/O has been switched to site B.

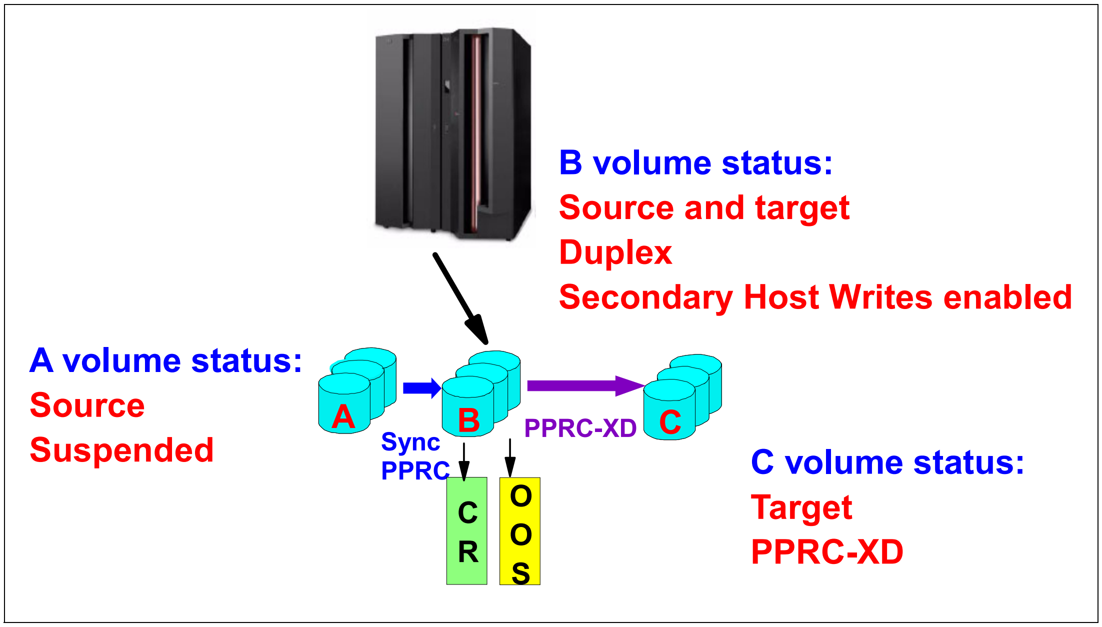



*Figure 8-21 I/O has been swapped to site B volumes*

## Switching back to site A

We assume now that the planned outage for site A has completed and we are ready to switch application I/O back to site A. This section describes the series of steps to achieve this.

### *Suspend the B → C PPRC-XD session*

After site A is repaired or ready again, the site A volumes need to be synchronized with the site B volumes before application I/O can be moved back to site A. In order to achieve this, a B → A PPRC session must be started. Since the site B volumes cannot be a primary for both B → A and B → C at the same time, the (PPRC-XD) B →C sessions must be suspended.

### *Reverse paths B → A and Failback to site B volumes*

With the B → C sessions already suspended, the PPRC Failback from B to A can now be done:

1. Establish B → A PPRC paths.
2. Issue a PPRC Failback command to the site B volumes. Failback processing does the following operations:
   a. Because the ESS is aware of the Asynchronous Cascading PPRC status, the OOS information will be used for the B → A PPRC session and the CR information will be used for the imminent (PPRC-XD) B → C sessions.
   b. Site A volumes are synchronized with site B.
   c. Although B is no longer primary for C, the *trusted resync* flag is set to allow a later re-synchronization.

### Quiesce application I/O to site B

The site A volumes are now in full duplex (fully synchronized) with the site B volumes, and so the site B volume's OOS information is no longer required. CR is still maintained to track updates to site B volumes since the updates will have to be copied to the site C volumes when the (PPRC-XD) B → C sessions are re-established.

Application I/O to the site B volumes can now be quiesced in preparation for switching back to site A. Figure 8-22 shows the status after the B → A sessions have been established (via PPRC Failback) and application I/O has been quiesced.

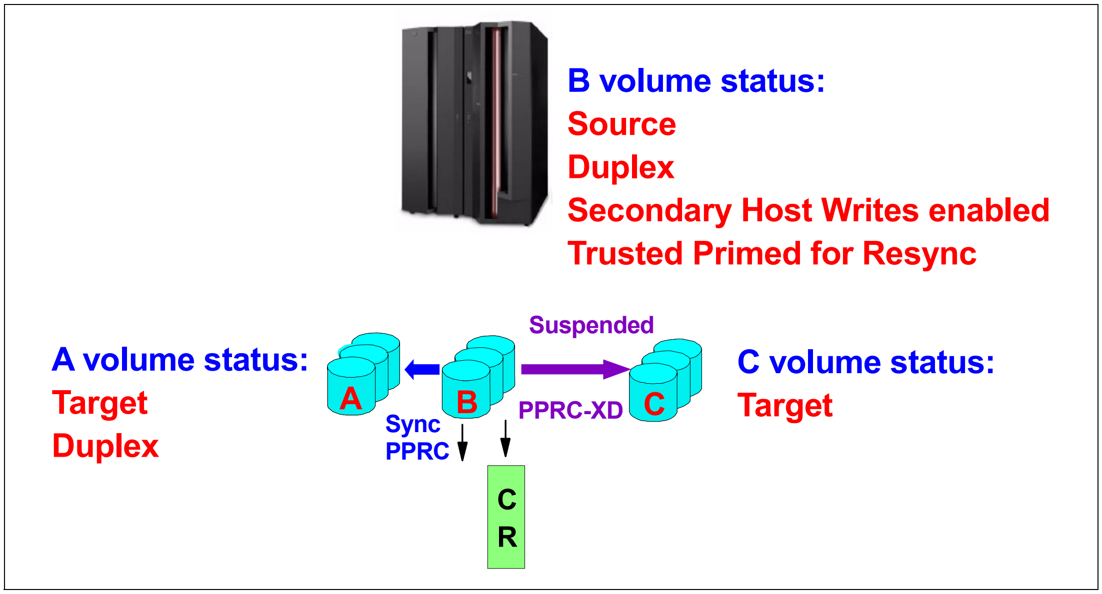



*Figure 8-22 Failback to site B volumes completed and application I/Os quiesced*

### Failover to site A volumes

Now site A volumes need to become primary volumes again before resuming application I/O updates. The following steps will accomplish this:

1. Establish the A → B PPRC paths.
2. Issue a PPRC Failover command to the site A volumes. This will:
   a. Terminate the B → A PPRC relationship.
   b. Establish the A → B PPRC relationship.
3. Issue a Failback command to the site A volumes:
   a. This will complete the restore of the A → B PPRC relationship. No re-synchronization is necessary since the volumes were in full duplex state.
   b. The cascading volume flag *Enable Secondary Host writes* is reset because host I/O will be directed to the site A volumes.

Figure 8-23 shows the volume states after Failover to site A.

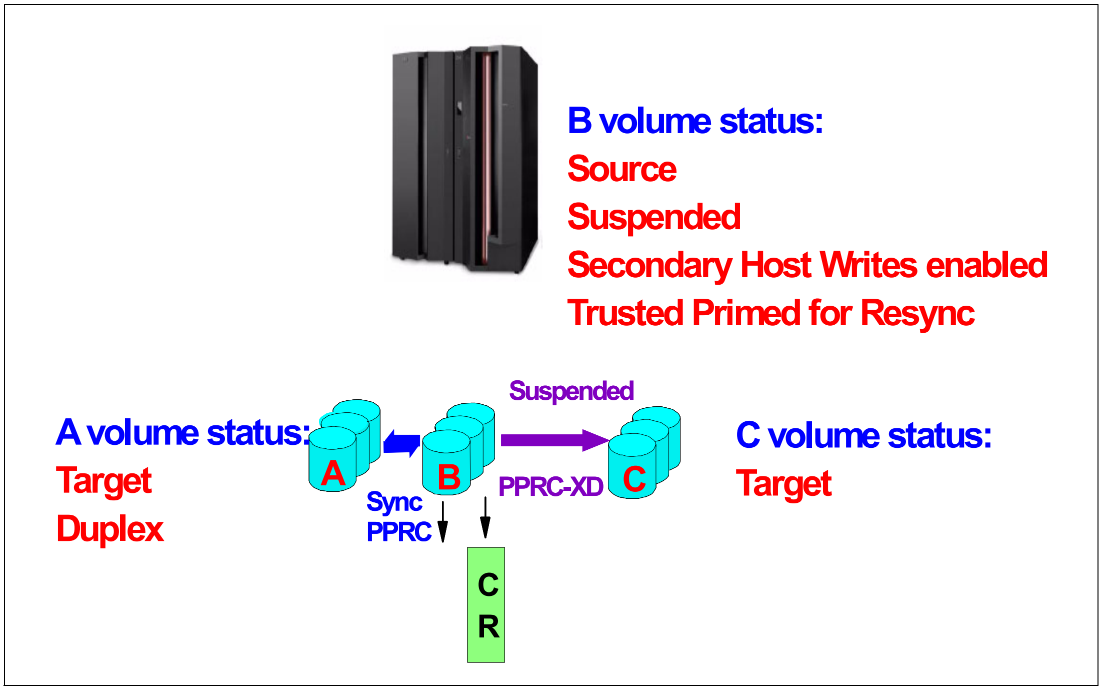



*Figure 8-23 Volume states after PPRC Failover to site A*

Figure 8-24 shows the volume states after the PPRC Failback to site A.

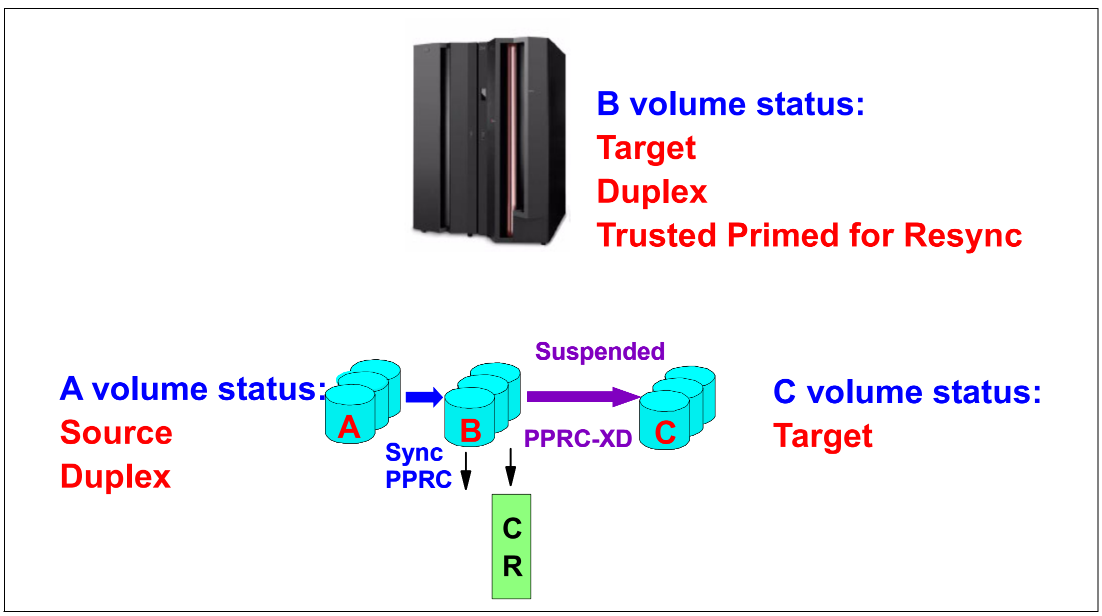



*Figure 8-24 Volume states after PPRC Failback to site A*

### *Return to normal Asynchronous Cascading PPRC sessions*

Now we can re-establish the normal Asynchronous Cascading PPRC environment, with sites A, B, and C fully functional.

1. Restart application host I/O to the site A volumes. Application writes are synchronously PPRC mirrored to site B. CR information has been maintained for the site B volumes.

2. Re-establish the (PPRC-XD) B → C pairs with *resync*. The *resync* option is allowed even though the site B volumes are SIMPLEX to the site C volumes. This is because the *trusted resync* flag was set during the Failback processing. The flag is reset during this processing. The CR information is transformed to OOS information, which is used to re-establish the B → C PPRC-XD relationship.

Figure 8-25 shows the status after returning to full Asynchronous Cascading PPRC.

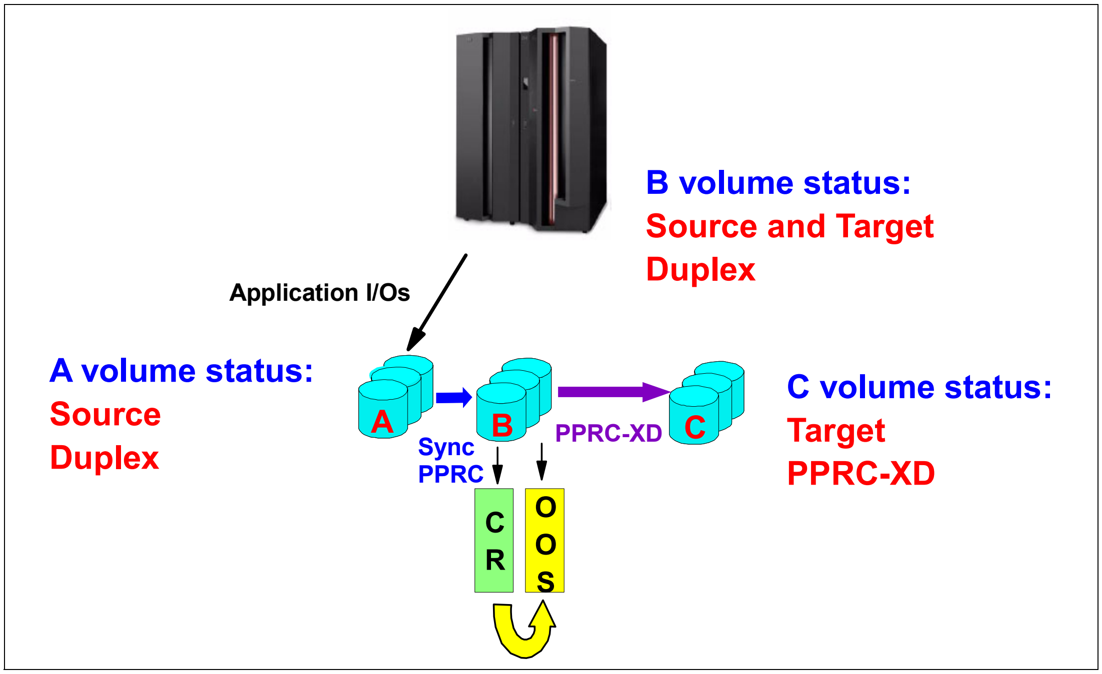



*Figure 8-25 Return to full Asynchronous Cascading PPRC relationship*

Normal operation for Asynchronous Cascading PPRC has now been re-established:

- Application I/O is directed to the site A volumes.
- Synchronous PPRC relationships A → B have been re-established.
- Non-synchronous PPRC-XD relationships B → C have been re-established. OOS information is maintained which can be used to synchronize the B → C sessions.

## 8.5.4 Failover/Failback for Asynchronous Cascading example

This is an example of setting up an Asynchronous Cascading PPRC environment and how to use PPRC Failover/Failback to switch application I/O. This example is a step-by-step procedure that follows the discussion in 8.5.3, “Failover/Failback for Asynchronous Cascading PPRC” on page 434.

In this example we use TSO and ICKDSF commands to:

1. Set up an Asynchronous Cascading PPRC environment using Fibre Channel links.
2. Set PPRC Failover to move the workload to site B.
3. Set PPRC Failover/Failback to move the workload back to site A.

Although the Web User Interface could be used, in our example the entire procedure is performed using TSO and ICKDSF commands, while the Web User Interface is only used to display the status of the configuration at each stage.

## Configuration setup

In this example we are using a two-site implementation of the Asynchronous Cascading PPRC configuration as discussed in "Two-site implementation" on page 433. So, the local site A and the intermediate site B are in fact within the same ESS, ESS A/B. The remote site C is in a different ESS, ESS C.

The following is the configuration used in our example (refer to Figure 8-26):

- ESS A/B - Serial 24663, WWNN - 5005076300C09DEF
- ESS C - Serial 23953, WWNN - 5005076300C09B29

We have a Fibre Channel link from SAID 000C on ESS A/B to SAID 002C on the same ESS A/B. This link is used to establish a path from LSS0 to LSS1 — both in ESS A/B, and this path is used for our Synchronous PPRC volume pairs (one pair in our example).

We also have a Fibre Channel link from SAID 0008 on ESS A/B to SAID 008C on ESS C. This link is used to establish a PPRC Fibre Channel link from LSS1 on ESS A/B to LSS0 on ESS C. This is used for our non-synchronous PPRC-XD volumes pairs (one pair in our example).

The volumes in LSS0 on ESS A/B are referred to as the site A volumes, the volumes in LSS1 on ESS A/B are referred to as the site B volumes, and the volumes in LSS0 on ESS C are referred to as the site C volume.

The SSIDs, device numbers and CCAs used are:

- LSS0 on ESS A/B has SSID X'1070' and we will use device number X'3010', CCA X'10'
- LSS1 on ESS A/B has SSID X'1071' and we will use device number X'3110', CCA X'10'
- LSS0 on ESS C has SSID X'1340' and we will use device number X'B010', CCA X'10'

Figure 8-26 illustrates the configuration used for this example.

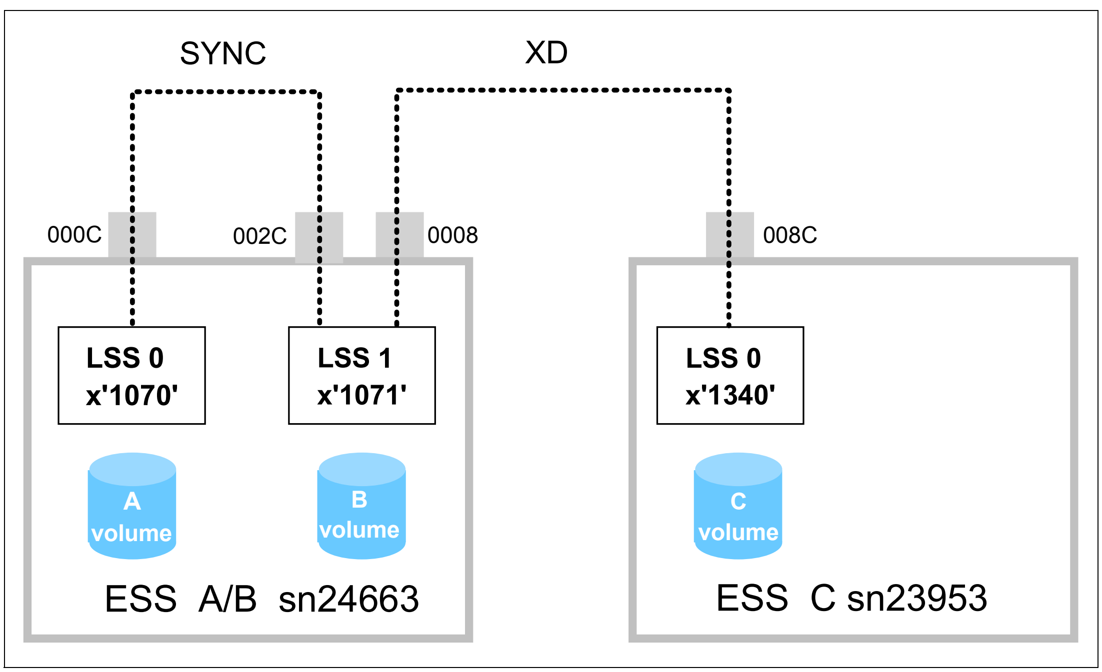



*Figure 8-26 Asynchronous Cascading PPRC configuration used in this example*

### Establishing the paths

The first step is to establish the paths from LSS0 to LSS1 in ESS A/B and from LSS1 in ESS A/B to LSS0 in ESS C. Example 8-1 shows the TSO commands to do this.

*Example 8-1 Establishing PPRC paths using Fibre Channel links*

```
CESTPATH  -
   DEVN(X'3010')         -
   PRIM(X'1070' 5005076300C09DEF X'00') -
   SEC(X'1071' 5005076300C09DEF X'01') -
   LINK(X'000C002C') -
   CGROUP(YES)

CESTPATH  -
   DEVN(X'3110')         -
   PRIM(X'1071' 5005076300C09DEF X'01') -
   SEC(X'1340' 5005076300C09B29 X'00') -
   LINK(X'0008008C') -
   CGROUP(NO)
```

### Establishing the pairs

First we establish one pair from LSS1 on ESS A/B to LSS0 on ESS C. This is our PPRC-XD pair. Example 8-2 shows the ICKDSF command to do this.

*Example 8-2 Establishing the PPRC-XD remote pair*

```
PPRCOPY ESTPAIR  -
   UNIT(3110)  -
   PRI(X'1071' 24663 X'10' )  -
   SEC(X'1340' 23953 X'10' )  -
   LSS(X'01' X'00') -
   MODE(NOCOPY) -
   OPTION(XD) -
   CASCADE
```

We then establish one pair from LSS0 on ESS A/B to LSS1 on ESS A/B. This is our Synchronous PPRC pair. Example 8-3 shows the ICKDSF command to do this.

*Example 8-3 Establishing the Synchronous PPRC local pair*

```
PPRCOPY ESTPAIR  -
   UNIT(3010)  -
   PRI(X'1070' 24663 X'10' )  -
   SEC(X'1071' 24663 X'10' )  -
   OPTION(SYNC) -
   MODE(COPY) -
   ONLINSEC(YES) -
   LSS(X'00' X'01')
```

**Note:** In this example, the secondary volume was ONLINE and so the ONLINSEC(YES) parameter was specified. Failure to do this would result in error message: `ICK34057I SECONDARY DEVICE FOUND TO BE GROUPED`.

The following figures show the status of the volumes at this stage. Figure 8-27 shows the status of the Synchronous PPRC pair. The screen capture was taken before the copy was complete (duplex pending state), so the disk icon is half shaded.

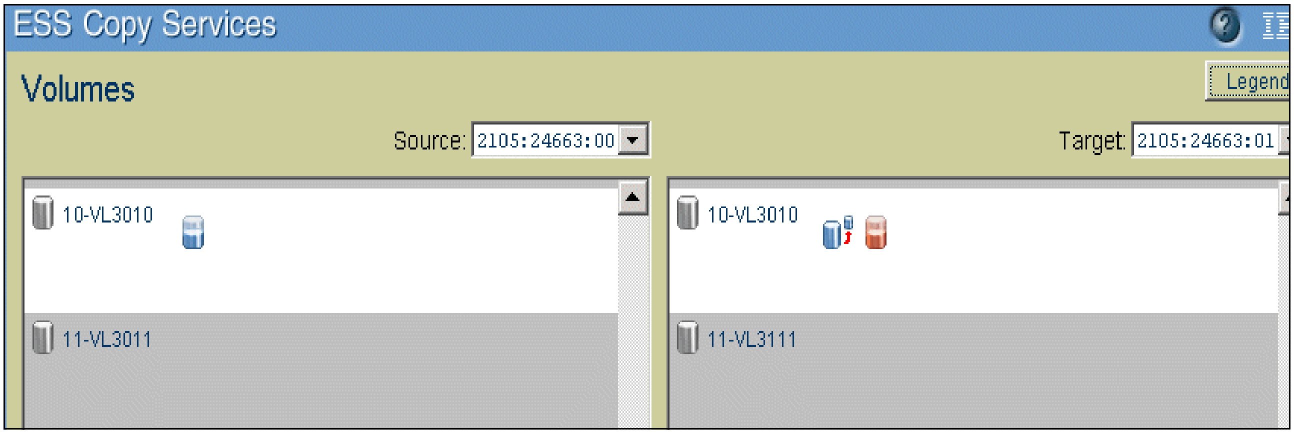



*Figure 8-27 Volumes in LSS0 and LSS1 in ESS A/B immediately after establishing pairs*

Figure 8-28 shows the status of the PPRC-XD pair.

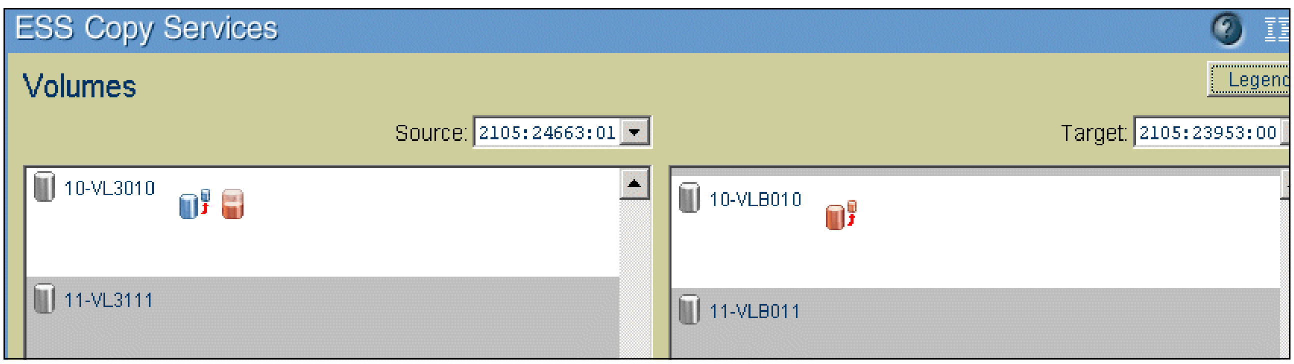



*Figure 8-28 Volumes in LSS1 in ESS A/B and LSS0 in ESS C after establishing pairs*

## Failover to site B volume

We now issue a PPRCOPY ESTPAIR command to the site B volume using the FAILOVER parameter. The ICKDSF command syntax is shown in Example 8-4.

*Example 8-4 PPRCOPY ESTPAIR command with FAILOVER parameter*

```
PPRCOPY ESTPAIR  -
    UNIT(3110)  -
    PRI(X'1071' 24663 X'10' )  -
    SEC(X'1070' 24663 X'10' )  -
    LSS(X'01' X'00') -
    ONLINSEC(YES) -
    FAILOVER
```

Figure 8-29 shows the status of the Synchronous PPRC volume pair at this stage. The status of the PPRC-XD target in LSS0 on ESS C has not changed.

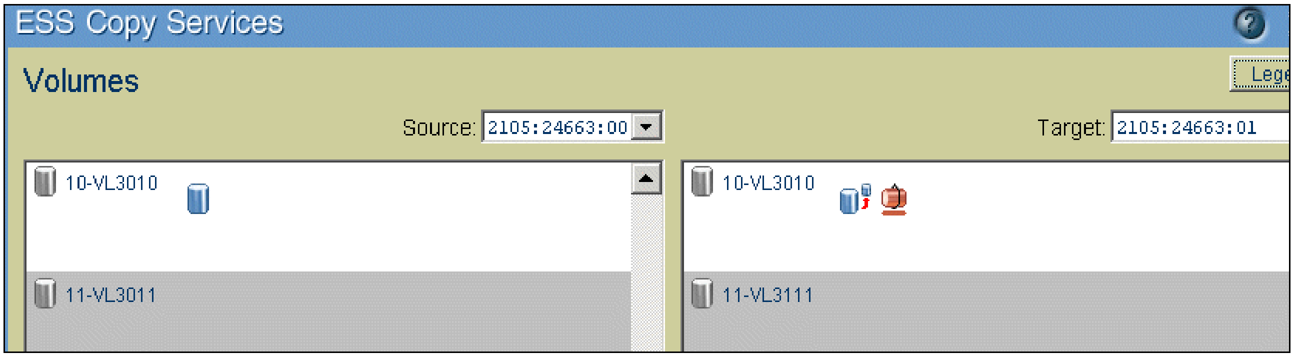



*Figure 8-29 Volumes in LSS0 and LSS1 in ESS A/B after Failover to B volume*

At this point application I/O can be swapped to the site B volumes. Figure 8-30 shows the Information Panel for the volume in LSS1 on ESS A/B, the intermediate site volume. Note the "Secondary Host Writes Enabled" status. This allows a volume which is a PPRC target to accept host I/O.

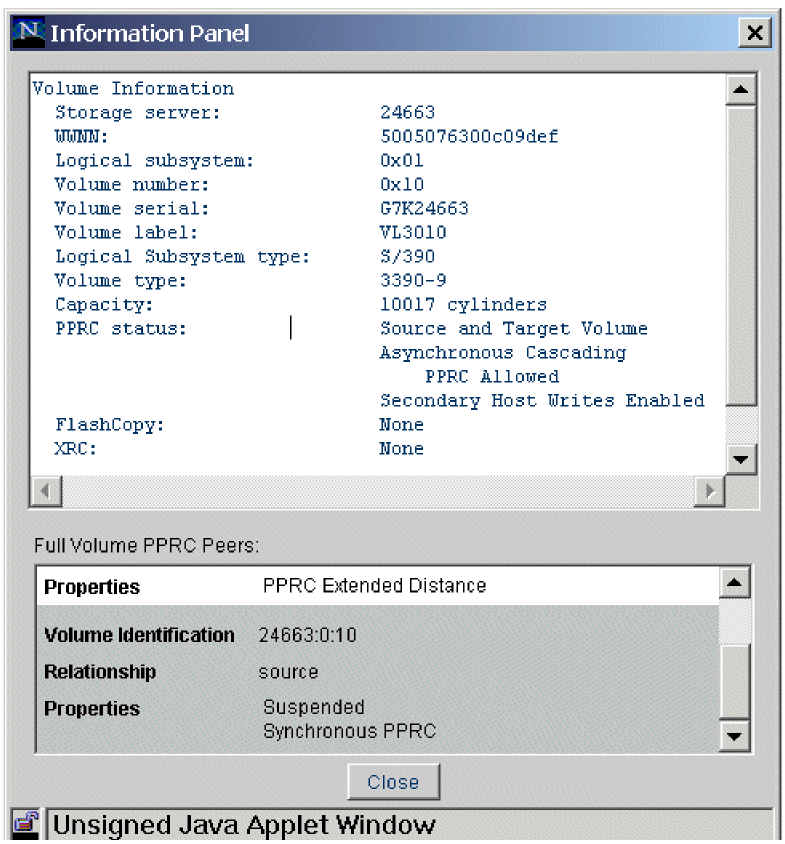



*Figure 8-30 Information display for the B volume after Failover*

The following steps follow the procedure detailed in "Switching back to site A" on page 436.

## Suspend the B → C sessions

This is the first step in moving application I/O back to site A. The PPRC-XD B → C sessions must be suspended because we need to establish a B → A session, and B cannot be the source for both at the same time. Example 8-5 shows the ICKDSF command to suspend the B → C PPRC-XD session.

*Example 8-5 ICKDSF PPRCOPY SUSPEND command*

```
PPRCOPY SUSPEND  -
    UNIT(3110)  -
    PRI(X'1071' 24663 X'10' )  -
    SEC(X'1340' 23953 X'10' )  -
    LSS(X'01' X'00')
```

Figure 8-31 shows the status of the PPRC-XD remote volume pairs at this stage. The status of the Synchronous PPRC source volume in LSS0 on ESS A/B has not changed.

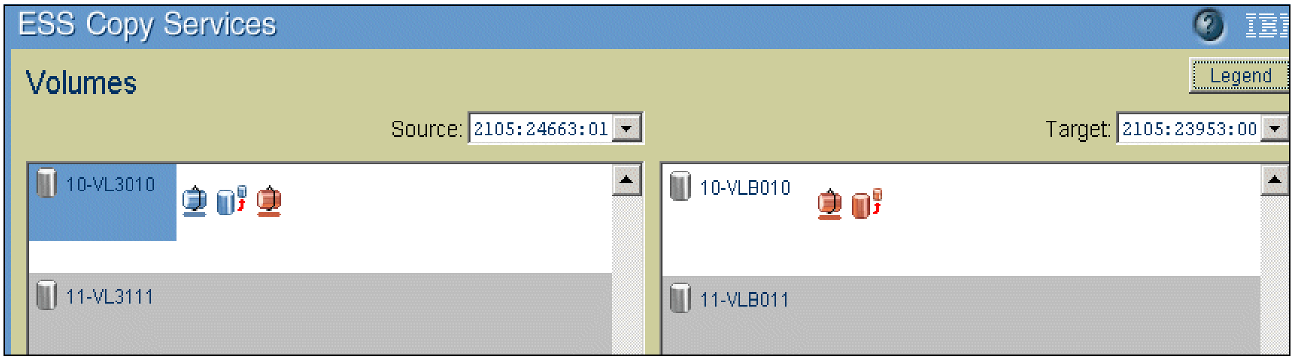



*Figure 8-31 Volumes in LSS1 in ESS A/B and LSS0 in ESS C after suspending B → C*

## Establish B → A paths

We can now establish the B → A path to allow the site A volume to be synchronized with the site B volume. Example 8-6 shows the ICKDSF command to establish the B → A path.

*Example 8-6* TSO CESTPATH command to establish B → A

```
CESTPATH  -
   DEVN(X'3110')          -
   PRI(X'1071' 5005076300C09DEF X'01') -
   SEC(X'1070' 5005076300C09DEF X'00') -
   LINK(X'002C000C') -
   CGROUP(YES)
```

## PPRC Failback to site B volume

The PPRC Failback processing will synchronize the site A volume with the site B volume and set the *trusted resync* flag to allow a later re-synchronization of the C volume with the B volume. Example 8-7 shows the ICKDSF command to Failback to the B volume.

*Example 8-7 ICKDSF PPRCOPY ESTPAIR command to Failback site B volume*

```
PPRCOPY ESTPAIR  -
    UNIT(3110)  -
    PRI(X'1071' 24663 X'10' )  -
    SEC(X'1070' 24663 X'10' )  -
    LSS(X'01' X'00') -
    ONLINSEC(YES) -
    FAILBACK
```

Figure 8-32 shows the status of the PPRC local volume pair at this stage. The status of the PPRC-XD target volume on LSS0 in ESS C has not changed. Note the *Trusted Primed for Resynchronization* icon.

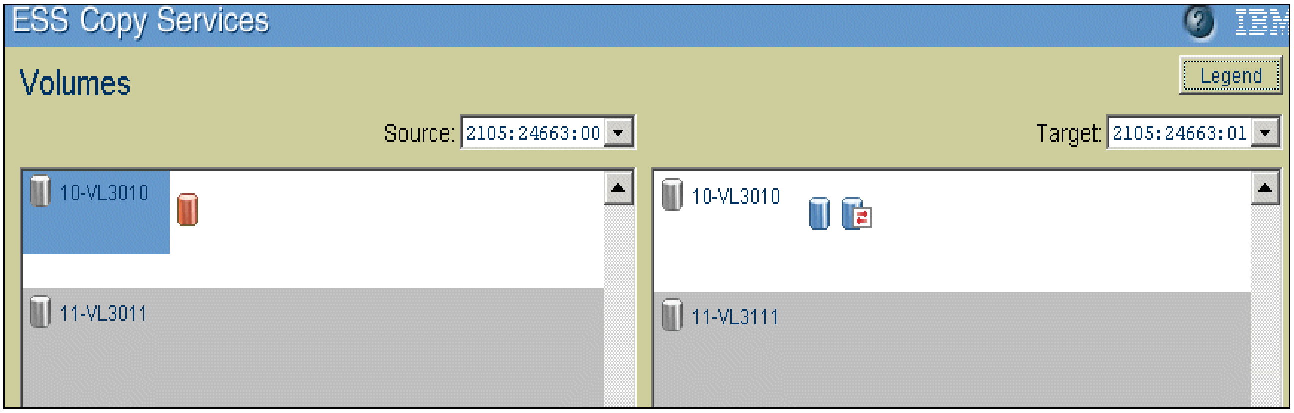



*Figure 8-32 Volumes in LSS0 and LSS1 on ESS A/B after Failback to B volume*

Figure 8-33 shows the information panel for the volume in LSS1 on ESS A/B, the intermediate site volume. Note the *Trusted Primed for Resynchronization* flag. This will allow a later re-synchronization of the B volume with the C volume.

The site A volume is now synchronized with the site B volume. We can now stop application I/O on the site B volume and begin the process of switching back to site A.

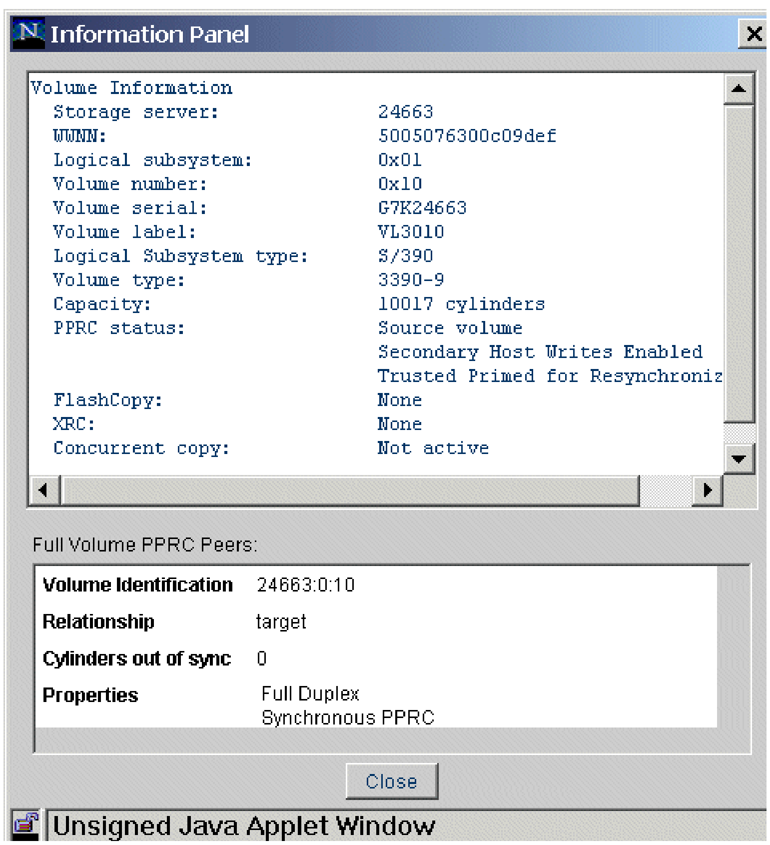



*Figure 8-33 Information Panel for the B volume after Failback to site B*

## Establish A → B path

We can now establish the A → B path for PPRC. Example 8-8 shows the TSO command to establish the A → B path.

*Example 8-8 TSO CESTPATH command to establish the A → B path*

```
CESTPATH  -
  DEVN(X'3010')          -
  PRIM(X'1070' 5005076300C09DEF X'00') -
  SEC(X'1071' 5005076300C09DEF X'01') -
  LINK(X'000C002C') -
  CGROUP(YES)
```

## PPRC Failover to site A volume

We can now issue the PPRC Failover to the site A volume. This will terminate the B → A PPRC relationship and establish the A → B PPRC relationship. Example 8-9 shows the ICKDSF command to Failover to site A.

*Example 8-9 ICKDSF PPRCOPY ESTPAIR command to Failover to site A*

```
PPRCOPY ESTPAIR  -
    UNIT(3010)  -
    PRI(X'1070' 24663 X'10' )  -
    SEC(X'1071' 24663 X'10' )  -
    LSS(X'00' X'01') -
    ONLINSEC(YES) -
    FAILOVER
```

Figure 8-34 shows the status of the PPRC volumes at this stage. The status of the PPRC-XD target volume in LSS0 on ESS C has not changed.

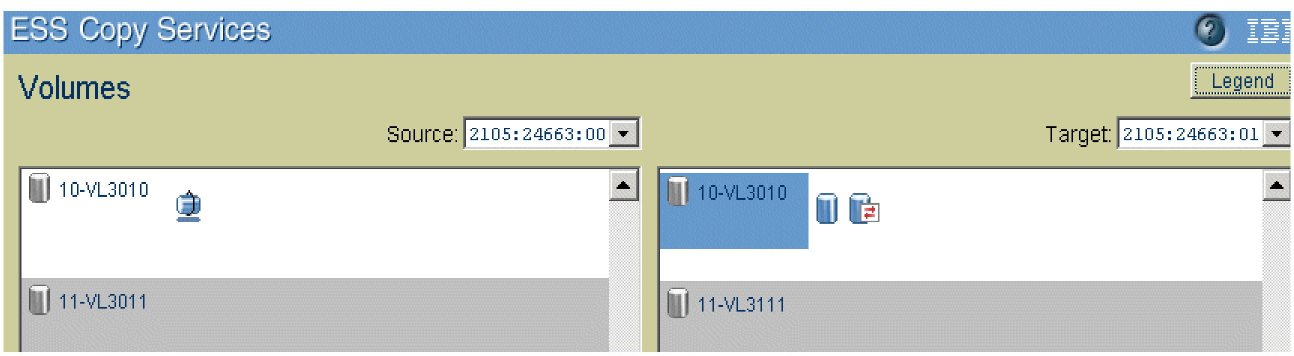



*Figure 8-34 Volumes in LSS0 and LSS1 on ESS A/B after Failover to A volume*

## PPRC Failback to site A volume

This will complete the restore of the A → B PPRC session and reset the *Enable Secondary Host Writes* flag on the B volume. Example 8-10 shows the ICKDSF command to Failback to site A.

*Example 8-10 ICKDSF PPRCOPY ESTPAIR command to Failback site A volume*

```
PPRCOPY ESTPAIR  -
    UNIT(3010)  - PRI(X'1070' 24663 X'10' )  -
    SEC(X'1071' 24663 X'10' )  -
    LSS(X'00' X'01') -
    ONLINSEC(YES) -
    FAILBACK
```

Figure 8-35 shows the status of the PPRC volumes at this stage. The status of the PPRC-XD target volume in LSS0 on ESS C has not changed. The screen shot shows the site A volume with a half shaded icon which normally means re-synchronization is in progress. There are no tracks being copied from volume A to volume B. It is only the PPRC status which is being updated from PENDING to DUPLEX.

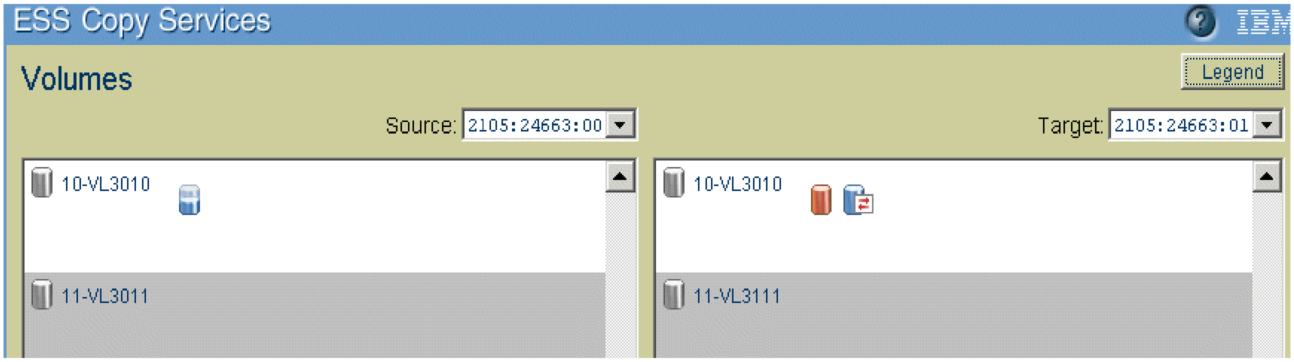



*Figure 8-35 Volumes in LSS0 and LSS1 on ESS A/B after Failback to A volume*

Figure 8-36 shows the Information Panel for the volume in LSS1 on ESS A/B, the intermediate site volume. Note the *Trusted Primed for Resynchronization* flag is still set but the *Enable Secondary Host Writes* has been reset.

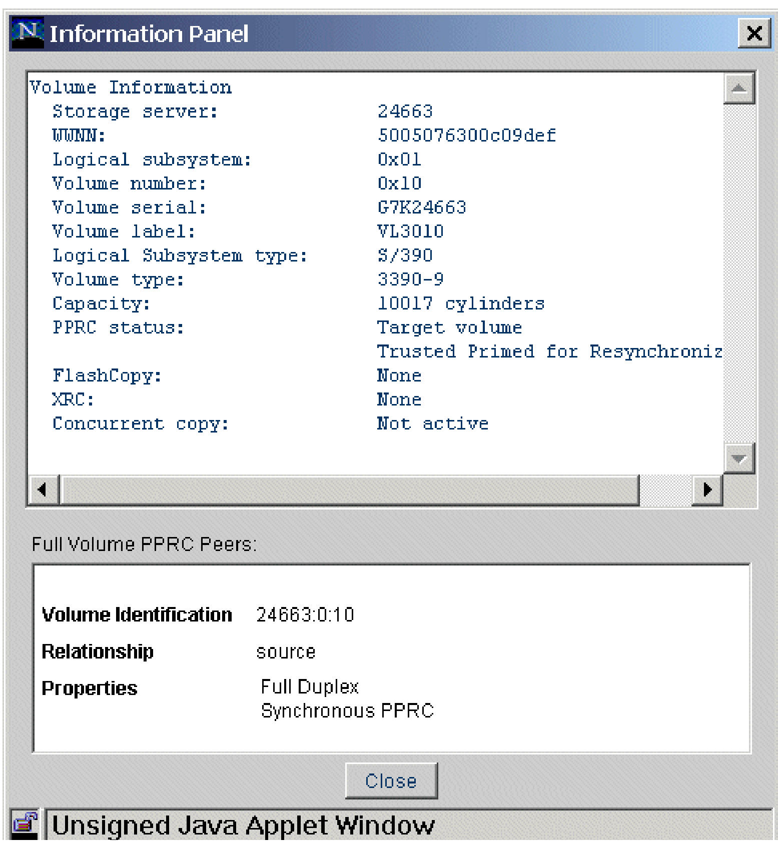



*Figure 8-36 Information Panel for the B volume after Failback to site A*

### Re-establish B → C sessions

Now that the A → B PPRC session has been restored, we can re-establish the B → C PPRC-XD pairs with *resync*. This is allowed because the B volume has the *Trusted Primed for Resynchronization* flag set. Example 8-11 shows the ICKDSF command to re-establish the B → C PPRC-XD session.

*Example 8-11 ICKDSF PPRCOPY ESTPAIR command to resynchronize B → C*

```
PPRCOPY ESTPAIR  -
   UNIT(3110)  -
   PRI(X'1071' 24663 X'10' )  -
   SEC(X'1340' 23953 X'10' )  -
   LSS(X'01' X'00') -
   MODE(RESYNC) -
   OPTION(XD) -
   CASCADE
```

This resets the *Trusted Primed for Resynchonization* flag. Our original configuration has now been restored.

# 8.6 PPRC and FlashCopy

This section illustrates some examples of the combined use of PPRC and FlashCopy.

PPRC and FlashCopy functions can be combined in order to obtain a tertiary copy. This third copy can be used for purposes like:

- Testing the disaster recovery configuration and procedures
- Creating a point-in-time consistent backup copy of a database
- Cloning application data for tests
- Creating data for business intelligence applications

Both PPRC and FlashCopy must be enabled on the ESS. If using the ESS Copy Services WUI then the maximum number of volume pairs that can be managed per Copy Services server is 2048 (or 4096 individual volumes). This number includes all the primary and secondary PPRC, PPRC-XD volumes *and* all the source and target FlashCopy volumes. When using TSO or DFSMSdss this consideration does not apply.

## 8.6.1 Duplicating data on a remote ESS

A procedure similar to the following could be used to clone data on a remote ESS for application testing purposes, the description of this example assumes that PPRC is already set up and running. These are the steps:

1. Verify that the volumes that are going to be duplicated are in full duplex state.
2. Suspend the PPRC pairs — PPRC starts change recording for later re-synchronization. At this moment the secondary volumes hold a consistent point-in-time copy of the primary volumes.
3. Invoke FlashCopy through DFSMSdss to make a tertiary copy of the secondary volumes.

   If you are executing FlashCopy on the same system image where the primary volumes are online, then the volume label must be changed.
4. After the short moment needed for the FlashCopy relationship to be established, re-synchronize the primary and secondary volumes —copying only the changed tracks to the secondary volume.

Figure 8-37 and Figure 8-38 illustrate the steps for this basic split mirroring procedure for duplicating production volumes, and where the application has not been interrupted at all.

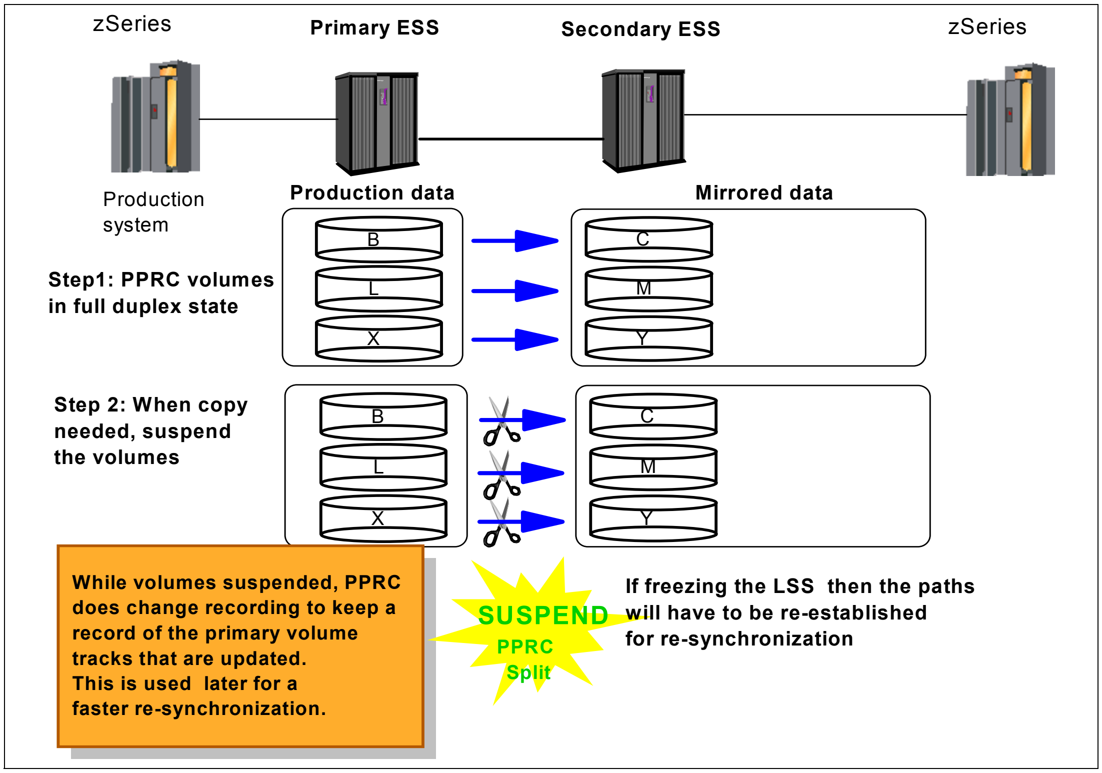



*Figure 8-37 Basic split mirror - suspending the pairs*

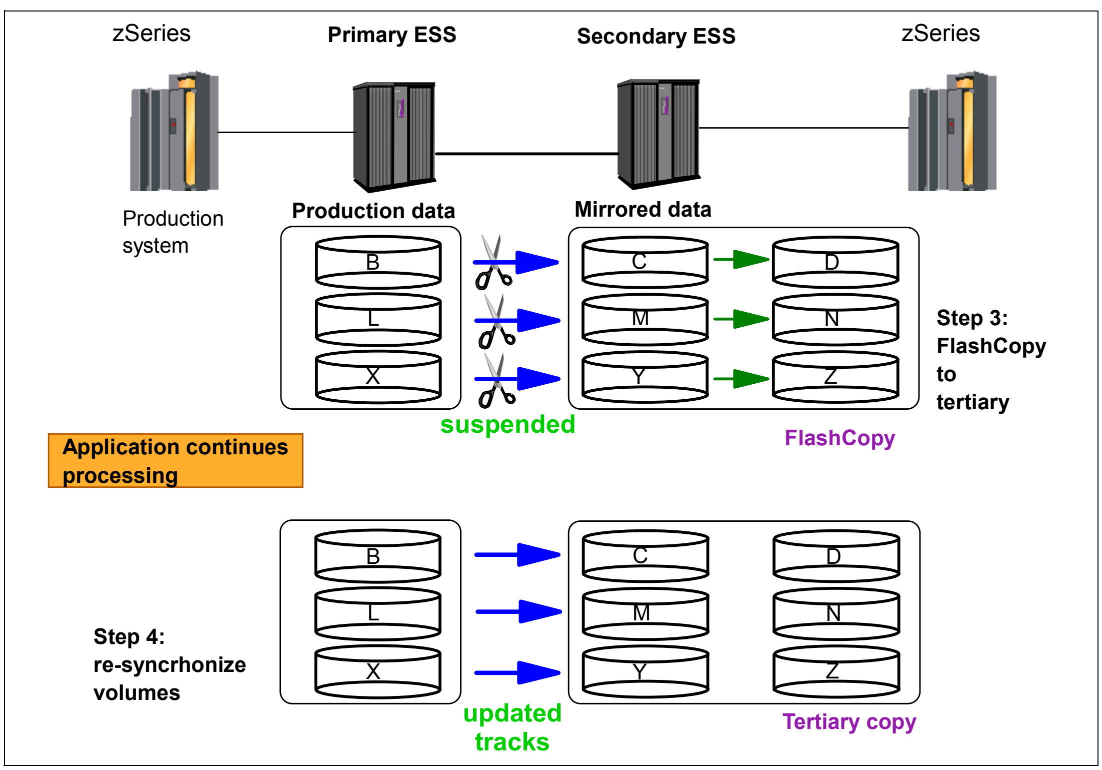



*Figure 8-38 Basic split mirror - creating tertiary copy and re-synch*

## 8.6.2 Split mirror backup/recovery for a database

PPRC by itself is a solution for protecting data against hardware outages or environmental disasters, but it does not protect data against user or application logical errors. For example, in the case of a user or application error, a hot-standby database would be in the same inconsistent state as the production database.

A split mirror backup recovery procedure can be implemented for a high availability backup recovery solution, where a tertiary point-in-time copy is taken on a ESS that is PPRC connected to the application site ESS. The PPRC relationship will be suspended (thus splitting the mirror) and will then be resumed for the re-synchronization of the secondary volumes. There is no need to stop the application while the database backup is being generated.

In the case of a user or application error, the database at the application site is available for analysis, while the secondary database can be recovered to a consistent point-in-time from its backup. The following publication can be referred to for further information on this implementation: *IBM TotalStorage Solutions for Disaster Recovery*, SG24-6547.

> **Note:** The following example is illustrated using TSO commands for managing the procedure. It can similarly be managed implementing tasks created with the ESS Copy Services WUI.

### Initial setup

The PPRC environment should be established for the required volumes. That is:

- All the hardware and software requirements for the PPRC environment are in place, including the ESCON links.
- Identifying all data sets and volumes that need to be in the PPRC configuration.
- Estimating the capacity requirements for the application site and the recovery site ESSs.
- Reviewing all operational considerations for the application and recovery systems.
- Building a disaster recovery plan.
- Testing the implemented solution.

The application site and the recovery site ESSs need to have the PPRC optional feature enabled, while the recovery site ESS needs the FlashCopy feature enabled in addition.

### Initial synchronization

The following steps describe the synchronization procedure for the initial setup for the DBt0 and DBt1 sets of volumes:

1. Establish the PPRC paths between the primary and the secondary ESSs, using the **CESTPATH** command.
2. Establish the PPRC pairs using the **CESTPAIR** command with **MODE(COPY)** parameter and **MSGREQ(YES)**, in order to be notified when the pair is in duplex state.
3. Issue **CQUERY** commands with the **VOLUME** parameter, to monitor all PPRC volumes for the duplex state.
4. Suspend all DB2 write I/O activity to database volumes and force DB2 to flush the log buffers, using the following DB2 command (refer to “DB2 suspend/resume log write activity” on page 414):

   ```
   SET LOG SUSPEND
   ```

5. Suspend all PPRC pairs using the CSUSPEND command. Now the mirror has been split and a consistent database image is available at the recovery site ESS on the secondary volumes.
6. The DB2 write I/O activity to the primary volumes can be resumed using the following DB2 command:

   ```
   SET LOG RESUME
   ```

7. At this time, the secondary volumes are off-line and suspended. We have to bring them to simplex state, and also put them online. The ID parameter on the CRECOVER command needs to be used in order to change their volser (see Figure 8-39).

   At this point, the DBt0 volume's VTOCs and VVDSs are out-of-sync: VOLC now has a VVDS with the name SYS1.VVOLA.VVDS. The CRECOVER command with ID parameter changed only the volser information, without altering the volume content.

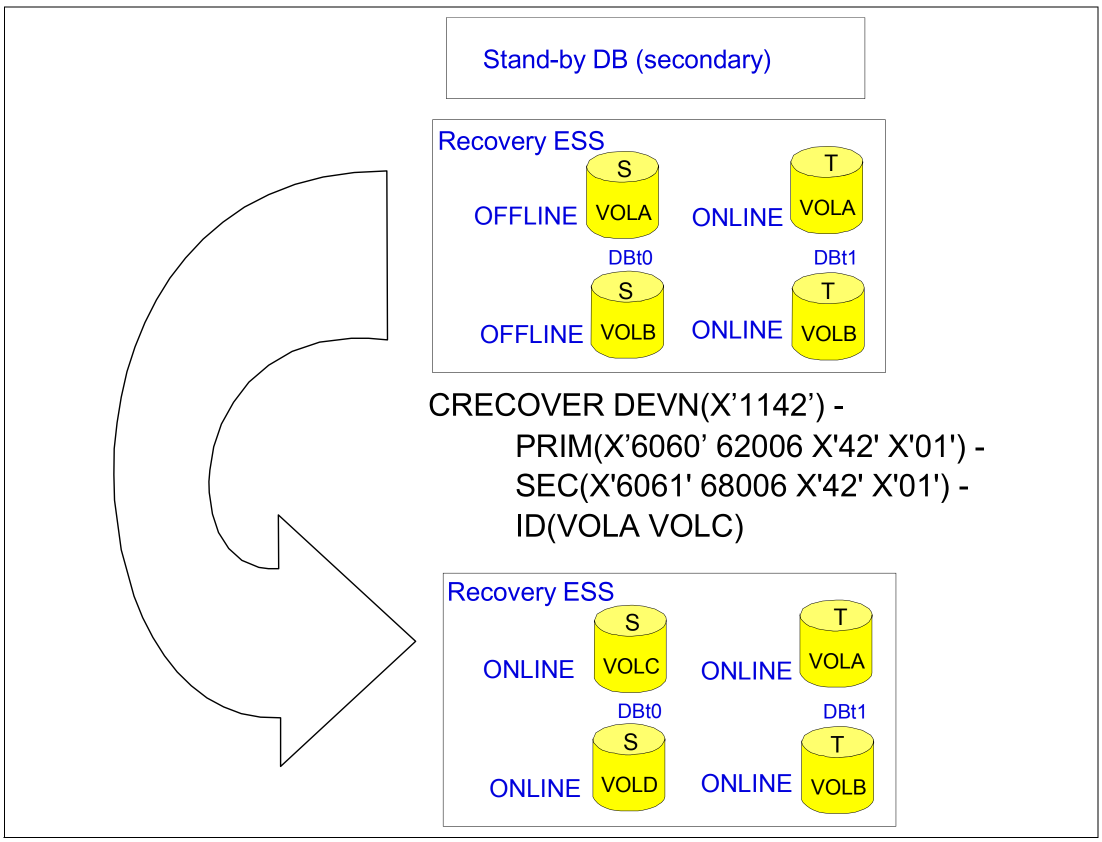
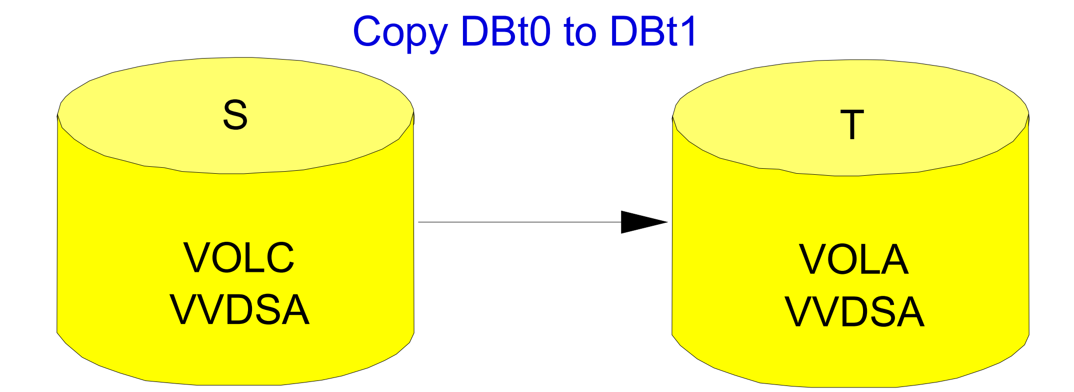



*Figure 8-39 Recovery the secondary volumes for system use*

8. Vary online the CRECOVERed set of volumes to the secondary system.
9. Now the DBt1 copies can be created by invoking FlashCopy with DFSMSdss; see Figure 8-40.




```
//COPYFULL JOB .....
//*
//INSTIMG  EXEC PGM=ADRDSSU
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=V,OUTLIM=3000
//SYSIN    DD *
  COPY FULL INDYNAM ((VOLC)) OUTDYNAM ((VOLA))
/*
```

*Figure 8-40 Invoking FlashCopy*

10. Re-synchronize the secondary volumes where the DB2 BSDS and the active and archive volume logs are being mirrored, by issuing the CESTPAIR command with MODE(RESYNC). This command is issued on the primary system, and a clip of the labels back to their original volumes is not needed, because the CESTPAIR RESYNCH and the subsequent write mirroring process will not modify the label. In other words, there will be a PPRC pairing between VOLA on the primary ESS and VOLC on the secondary ESS, with equal contents except the label.
11. Dump DBt1 volumes (the tertiary volumes) to tape, using DFSMSdss full volume dump.

### Generating the backup without application disruption

During normal processing only logs, bootstrap data sets (BSDS), and ICF catalog volume pairs are in duplex state. During the window used to produce a new point-in-time copy, the *data* volumes are suspended — PPRC will start change recording for these volumes. Starting from this point, the process of the split mirror backup for a disaster recovery solution is as follows:

1. Suspend the logs, BSDS, ICFCAT, and user catalog volumes by using the **CSUSPEND** command on the primary system.
2. On the secondary system, issue the **CRECOVER** command with the **ID** parameter to the secondary volumes. This command will bring the secondary volumes to simplex state.
3. Vary the secondary set of volumes online.
4. Invoke FlashCopy to do a full volume copy from the DBt0 volumes (the secondaries) to the DBt1 volumes (the tertiaries) for the logs, BSDS, and catalogs volumes.
5. Vary the secondary devices offline again.
6. Re-synchronize the data volumes plus the logs and the BSDS, and catalogs volumes issuing a **CESTPAIR** command with **MODE(RESYNC)** and **MSGREQ(YES)**. Only the updated tracks are copied to the secondary volumes.

7. Suspend the write activity to the DB2 database by using the following command:

```
SET LOG SUSPEND
```

8. When the volumes are in full duplex and DB2 writes suspended, issue a CSUSPEND command for all PPRC pairs — data volumes, logs, BSDS and catalogs — to split the mirror.
9. Resume the DB2 write I/Os by issuing the following DB2 command:

```
SET LOG RESUME
```

10. Vary the secondary data volumes online.
11. Issue the CRECOVER command against the secondary data volumes to bring them to simplex state:

```
CRECOVER DEVN(X'1142') -
PRIM(X'6060' 62006 X'42' X'01') -
SEC(X'6061'68006 X'42' X'01')
```

12. Vary the secondary data volumes offline again.
13. Invoke FlashCopy to obtain a full volume copy from the DBt0 data volumes (secondaries) to the DBt1 volumes (tertiaries).
14. Re-synchronize the log, BSDS, and catalog volumes issuing a CESTPAIR command with the RESYNC parameter option, for the volume pairings where the logs, BSDS, and catalogs reside.
15. Dump DBt1 volumes (tertiary copy) to tape (full volume dump).

This process can be rerun at any time to increment the point-in-time. The time it takes to dump to the tape is the limiting factor for the frequency.

### Recovery

The recovery process depends on the event that caused the application outage. If there was an outage at the application site, the following steps should be taken:

1. Issue CRECOVER commands against all database volumes that still are in duplex state.
2. Follow normal DB2 recovery procedures to apply the DB2 logs on the database — DBt0 volumes.
3. Restart DB2 on the recovery site, using the recovered database on DBt0 volumes.

If the outage is due to a user or application logical error, both DBt0 (primary) and DBt2 (secondary) sets of volumes are destroyed or corrupted. The recovery must be done with the following procedure:

1. Issue CDELPAIR and CDELPATH commands to all the defined volume pairings and paths in the PPRC configuration, in order to remove all the PPRC definitions.
2. Invoke FlashCopy to copy the DBt1 volumes (tertiary point-in-times) to the DBt0 (secondary) volumes.
3. Create a new DBt2 (primary) set of volumes.
4. Establish a PPRC configuration where the primary volumes are the DBt0 volumes, and the secondary are the DBt2 volumes. Establish the paths in the opposite direction as when they are in normal PPRC configuration. Issue CESTPAIR commands to copy all the DBt0 volumes to DBt2 volumes.
5. When the copy is complete, issue CSUSPEND against all PPRC pairings.
6. Issue CRECOVER against all DBt2 volumes.

7. Apply the DB2 logs on the database on the DBt2 volumes and restart production DB2 using this database.
8. Reestablish the normal PPRC configuration for the on going split/mirror backup.

Database recovery procedures are discussed in more detail in the IBM publication, *IBM TotalStorage Solutions for Disaster Recovery,* SG24-6547.

# 8.7 Asynchronous PPRC solutions

In this section we describe how you can manage a planned or unplanned outage with Asynchronous PPRC for ESSs at LIC 2.4.x or higher.

## 8.7.1 Unplanned outage and switch to the remote site

Asynchronous PPRC is designed to provide application consistent data at a remote site at a global distance even in cases of a rolling disaster at the local site. For the following steps we assume, that you have an established Asynchronous PPRC session continuously creating Consistency Groups at the local site and sending them to the remote site; see Figure 8-41. If you have a disaster at the primary site and therefore have to start the production at the remote site, you will need to ensure that you use the data from the most valid Consistency Group at the secondary site.

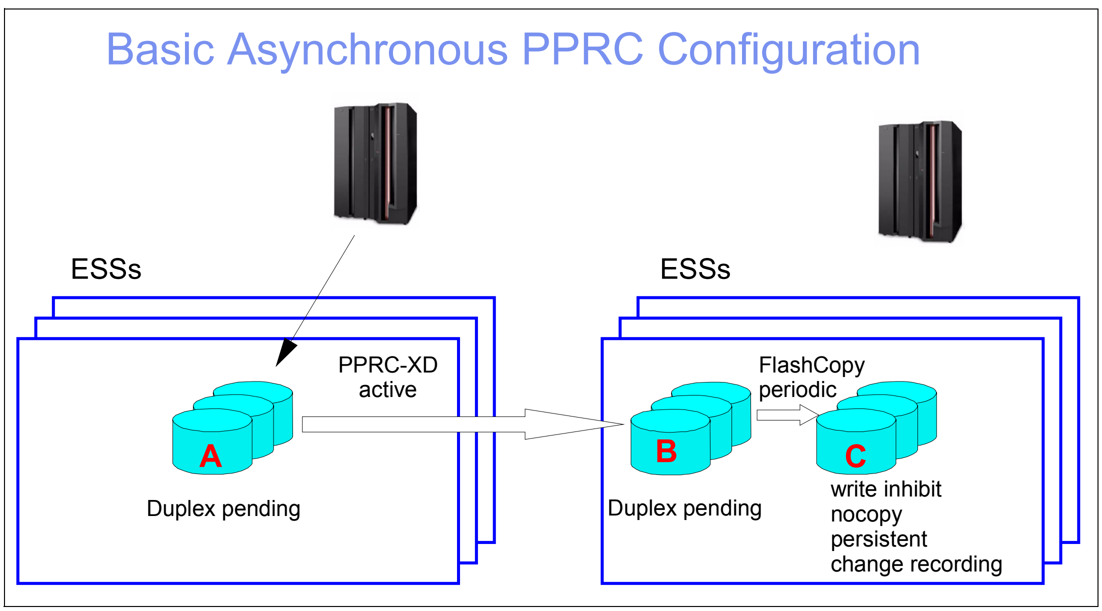



*Figure 8-41 Basic Asynchronous PPRC configuration*

You will later want to return production to the local site. If you have a running system at the remote site that is independent of the local site, you can use the ICKDSF commands to perform the failover. You should have jobs and/or automation procedures prepared to do the steps below. Otherwise you have to use the Web User Interface with prepared tasks to perform the following steps for a disaster at the primary site where all or a part of the local site is lost:

1. If the Master of the Asynchronous PPRC session is still active, issue the TERMINATE SESSION command to the Master, otherwise go to step two. This will stop the Master trying to create Consistency Groups if this process is still running. Address this command to the same LSS you used to start the session.

2. Failover to the PPRC secondary volumes using the ESTABLISH PAIR command with the FAILOVER option at the remote site to the PPRC secondary volumes. These volumes become *primary suspended*.
3. It is required to check the consistency status of the FlashCopy pairs even if it is unlikely to have inconsistent data at the FlashCopy target volumes. This can be done by issuing the FLASHCPY QUERY RELATIONS command to each of the FlashCopy primary volumes. The queries provide a FlashCopy sequence number for each volume and whether or not the volume is revertible to the previous FlashCopy relationship.

   In most of the cases, the volumes are consistent and you can continue with step 6. See the table in 4.3.2, "ICKDSF commands for Asynchronous PPRC" on page 165 for more details on the consistency and the necessary actions. If the indication directs you to revert the FlashCopy pairs back to the previous FlashCopy relationship, submit the FLASHCPY WITHDRAW command with the REVERT option. If you are directed to commit all of the FlashCopy pairs to the new FlashCopy relationship submit the FLASHCPY WITHDRAW command with the COMMIT option.

**Note:** These commands do not withdraw the relationships.

4. Issue a FLASHCPY ESTABLISH PAIR command with FASTREVERSERESTORE and TGTOKASPPRCPRIM(YES) parameter to the PPRC secondary volumes. This will reverse the direction of the FlashCopy relationship and copy the consistent data to the suspended PPRC secondary volumes at the remote site.

**Note:** The FlashCopy relationship will end after the background process has finished. The former FlashCopy target volumes, the **C** volumes, (see Figure 8-41) are no longer usable.

5. Before you continue the background copy process should be finished. This should be very fast, as only small data differences have to be copied. Before you start the applications at the PPRC secondary volumes (now suspended primary) you should establish a FlashCopy to the original FlashCopy target volumes with the specific options as the previous inband command for establishing the original FlashCopy. Then set the former PPRC secondary volumes online to the remote systems, test the data and start the applications.
6. If your applications at the remote site are running and the local site has recovered from the disaster you may want to revert the production workload to the local site again. For this reason you should establish PPRC paths to the local site ESS to enable the PPRC mirror to the local site. As Fibre Channel PPRC links are bidirectional links these paths can be established prior to the procedure.
7. Issue the PPRCOPY ESTPAIR command with the mode XD and FAILBACK option to the remote site PPRC volumes. Now the local site volumes change from *primary suspended* to *PPRC pending* copy state. The changed data from the remote site will be drained now to the local site.
8. Wait until the first pass of the PPRC-XD copy process of all volumes has finished, then shutdown the applications. Use the PPRCOPY QUERY command to check the volumes for out-of-sync cylinders. Alternatively you can catch-up the volumes to the duplex state with the PPRC ESTABLISH PAIR command with synchronous mode. When the number of out-of-sync cylinders is zero, proceed to the next step.
9. Submit the PPRCOPY ESTPAIR command with the FAILOVER option to the local volumes that will become *primary suspended*.

10. Reestablish the paths from the local to the recovery site again. Now you can start the applications at the local site, check the data and if the systems are running issue the PPRCOPY ESTPAIR command with FAILBACK and mode XD to the local volumes.
11. Finally Asynchronous PPRC has to be restarted again by submitting the PPRCOPY STARTASYNCOPY command to the local volumes.

> **Note:** If the disaster destroys the primary devices and you need to replace them, previously defined jobs or tasks can not be used because the serial numbers of the new ESSs are different.

## 8.7.2 Planned outage

When you have implemented Asynchronous PPRC you may need to have planned outages to enable hardware maintenance at the local or the remote site or to generate another copy of the data at the remote site for disaster recovery tests or production test purposes. Usually the hardware maintenance of the Enterprise Storage Server is non-disruptive but if you run Asynchronous PPRC you have many components in the chain of data flow that have to be considered.

The following steps can be done by the Web User Interface or ICKDSF for the z/OS volumes.

### Planned outage at the remote site

This can be done the following steps:

1. Issue the PPRCOPY TERMASYNCCOPY command to pause the session. This will stop the Asynchronous PPRC session but finish the creation of the latest Consistency Group at the FlashCopy target volumes at the secondary site. Address this command to the same LSS you used to start the session. You should check the status of the session using the PPRC QUERY ASYNCCOPY command before you continue.
2. Suspend the PPRC-XD relationships. Issue the suspend command to all of the primary volumes. The PPRC secondary volumes become suspended secondary volumes and contain fuzzy copies of the data.
3. Now you can perform the actions you have planned for this outage at the remote site. Usually this is for maintenance or reconfiguration.
4. If the secondary site is ready again you can reestablish the PPRC-XD copy pairs with the RESYNC parameter, to update the remote site volumes with the data that has changed at the local site.
5. Restart the Asynchronous PPRC session with the same options as before and check with the PPRC QUERY ASYNCCOPY command if the formation of Consistency Groups continues.

### Planned outage at the local site and switch to the remote site

We assume you have an established Asynchronous PPRC session continuously creating Consistency Groups at the remote site. If you have to shut down the primary site for any reason and run the application at the other site for some time and you want to return production at the local site later again, you can use the following failover and failback procedure:

1. Shut down the applications at the local site.
2. Wait until all PPRC-XD pairs do not have any out-of-sync cylinders. They should take a short time to catch up and the primary and secondary volumes will have the same data.

You can get this information easily with QUERY OOSYNCSTATE of ICKDSF or the Asynchronous PPRC Information panel of the Web User Interface for every primary ESS.

3. Issue the TERMASYNCCOPY command with the PAUSE option to pause the session. This will stop the Asynchronous PPRC session but finish the creation of the latest Consistency Group at the FlashCopy target volumes at the secondary site. Address this command to the same LSS you used to start the session. You should check the status of the session using the PPRC QUERY ASYNCCOPY command before you continue.

**Tip:** Issue the pause first and ensure success before terminating to ensure a current consistent copy has been taken.

4. Next issue the TERMASYNCCOPY command with the TERMINATE option to terminate the session. You should again check the status of the session using the PPRC QUERY ASYNCCOPY command before you continue.
5. Suspend PPRC for all the volumes using the PPRCOPY SUSPEND command.
6. Failover to the PPRC secondary volumes with PPRCOPY ESTPAIR command at the remote site to the PPRC secondary volumes. These volumes become *primary suspended*. Now the maintenance at the primary site can be done.
7. You can start the applications at the remote site now. The FlashCopy target volumes still contain the data from the time of failover. Be aware, that you are not protected from disasters at the remote site until you switch back to the local site and re-establish Asynchronous PPRC.
8. If the local site is ready for production again, you should establish PPRC paths to the local site ESS to enable the reverse PPRC mirror to the local site. As Fibre Channel PPRC links are bidirectional links these paths can be established prior to the procedure.
9. Issue the PPRC ESTABLISH PAIR command with mode XD and FAILBACK option to the remote site PPRC volumes. Now the local site volumes change from *primary suspended* to *PPRC pending* copy state. The changed data from the remote site will be drained now to the local site.
10. Wait until the first pass of the PPRC-XD copy process of all volumes has finished. The PPRC QUERY command can be used to check the volumes for out-of-sync cylinders. Alternatively you can catch-up the volumes to the duplex state with the PPRC ESTABLISH PAIR command with OPTION(SYNC).
11. When the PPRC copy has completed first pass, shut down the applications at the remote site.
12. Submit the PPRCOPY ESTPAIR command with the FAILOVER option to the local volumes that will become primary suspended.
13. Now you can start the applications at the local site.

**Tip:** Ensure that applications are stable before copying data to the remote site. This will protect the data at the remote site which could be returned to if applications at the primary site failed to start.

14. Reestablish the paths from the local to the recovery site again. Issue the PPRCOPY ESTPAIR command with FAILBACK and Mode XD to the local volumes.
15. Finally Asynchronous PPRC has to be restarted again by submitting the PPRCOPY STARTASYNCCOPY command to the local volumes.

## 8.7.3 Create a consistent set of test data with a planned outage

For disaster recovery testing, application testing or application development, you may need a consistent copy of your production data at the remote site.

The following steps describe a similar procedure to create this set of data with minimal impact at the primary site. See Figure 8-42.

1. Quiesce the applications at the primary site.
2. Wait until the PPRC-XD copy process of all volumes has finished and there are no out-of-sync cylinders. The PPRC QUERY command can be used to check the volumes for out-of-sync cylinders.
3. Pause the Asynchronous PPRC session with the PPRCOPY TERMASYNCCOPY command.
4. To allow Asynchronous PPRC to be resumed, optionally a separate set of volumes can be created using FlashCopy. Issue the FlashCopy inband command with mode COPY to the PPRC primary volumes by addressing a second set of FlashCopy target volumes, D, at the remote site. Refer to Figure 8-42.
5. The remote D volumes can now be used at the remote site, and the Asynchronous PPRC restarted.
6. When FlashCopy is initiated, resume the applications.
7. Resume the paused Asynchronous PPRC session using the PPRCOPY STARTASYCCOPY command.

**Attention:** If the FlashCopy to the D volumes is persistent, then before any recovery can be done from the C volumes in a disaster recovery situation, the D volume FlashCopies must be withdrawn.

The PPRC target volumes now have two FlashCopy targets. Only the first relationship belongs to Asynchronous PPRC. The second relationship is not persistent and can not have change recording on. It can now be used for testing purposes.

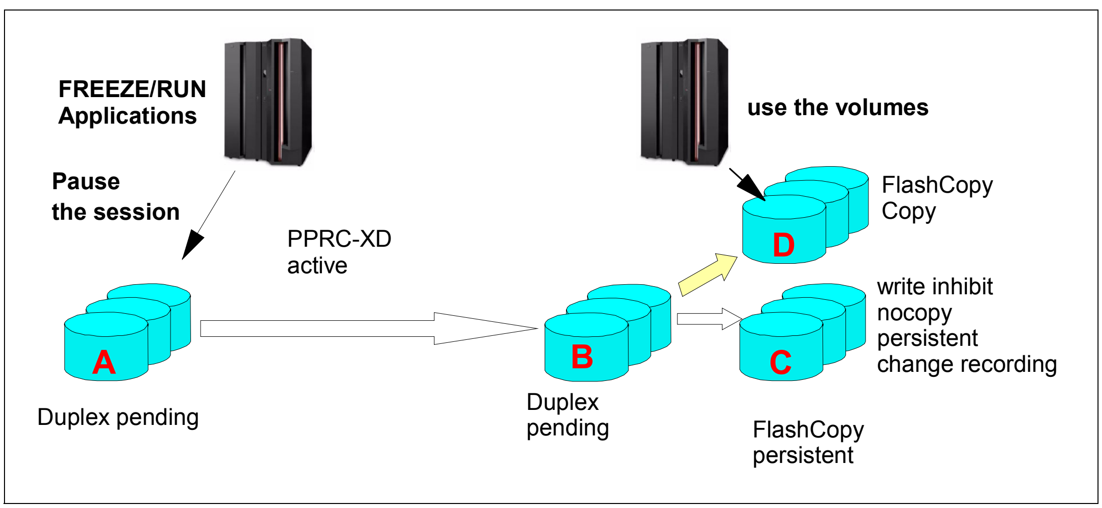



*Figure 8-42 Create a set of test volumes at the remote site*

## 8.8 FlashCopy solutions

This section contains several examples of FlashCopy solutions.

### 8.8.1 Establishing a FlashCopy relationship

To establish FlashCopy relationship, follow these steps:

1. In the ESS Copy Services Volumes panel, click to select the FlashCopy source volume. A blue rectangle will enclose the selected source volume (see Figure 8-43).
2. Then right-click the desired target volume. A red rectangle will enclose the selected target volume (see Figure 8-43). With FlashCopy V1, the target volume must be on the same LSS as the source volume.

> **Tip:** If you are going to select several volume pairs, you have the alternative of initially clicking the **Enter Multiple Selection Mode** button, and then iterate through steps 1 and 2 until you select all the volumes. Once all the volumes have been selected, you then proceed with steps 3 through 6.

3. Right-click the target volume again to bring up the Select Task Type wizard panel (also shown in Figure 8-43).

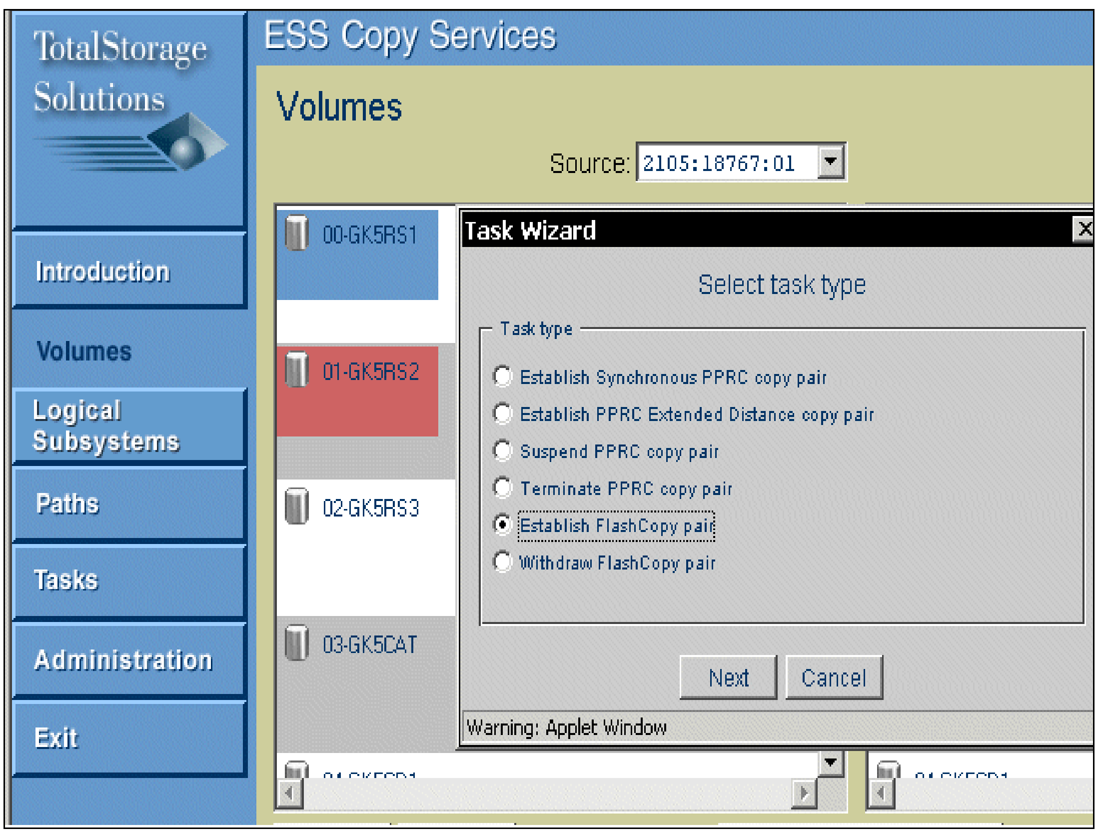



*Figure 8-43 Establish FlashCopy relationship*

4. Select **Establish FlashCopy pair** and click **Next**.

5. At the Select Copy Options wizard panel, select the desired copy options and click **Next**. The Select Copy Options panel is shown in Figure 8-44, and in our example, no options were selected, so the FlashCopy will be established requesting a background copy (default).

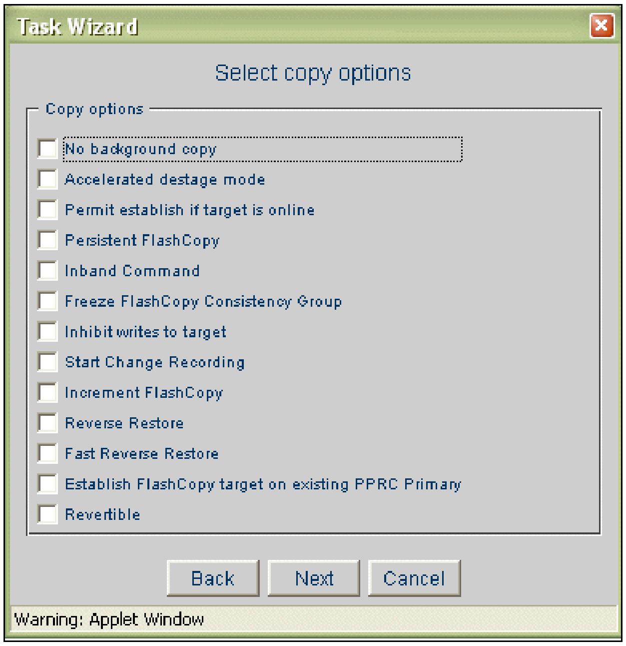



*Figure 8-44 Select copy options*

6. Enter the name and description of the task at the Define Task panel, and either click **Save** to save the defined task, or click **Run** to execute the task (see Figure 8-45).

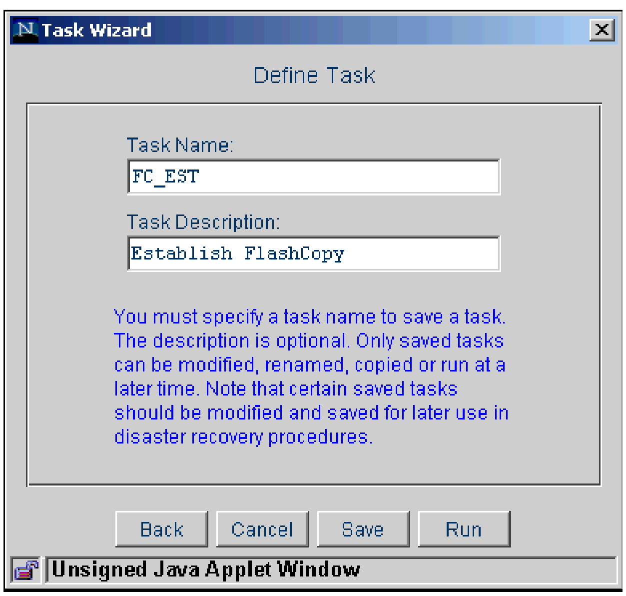



*Figure 8-45 Define task*

After the FlashCopy pair has been established, the volumes will display the FlashCopy icons as shown in Figure 8-46.

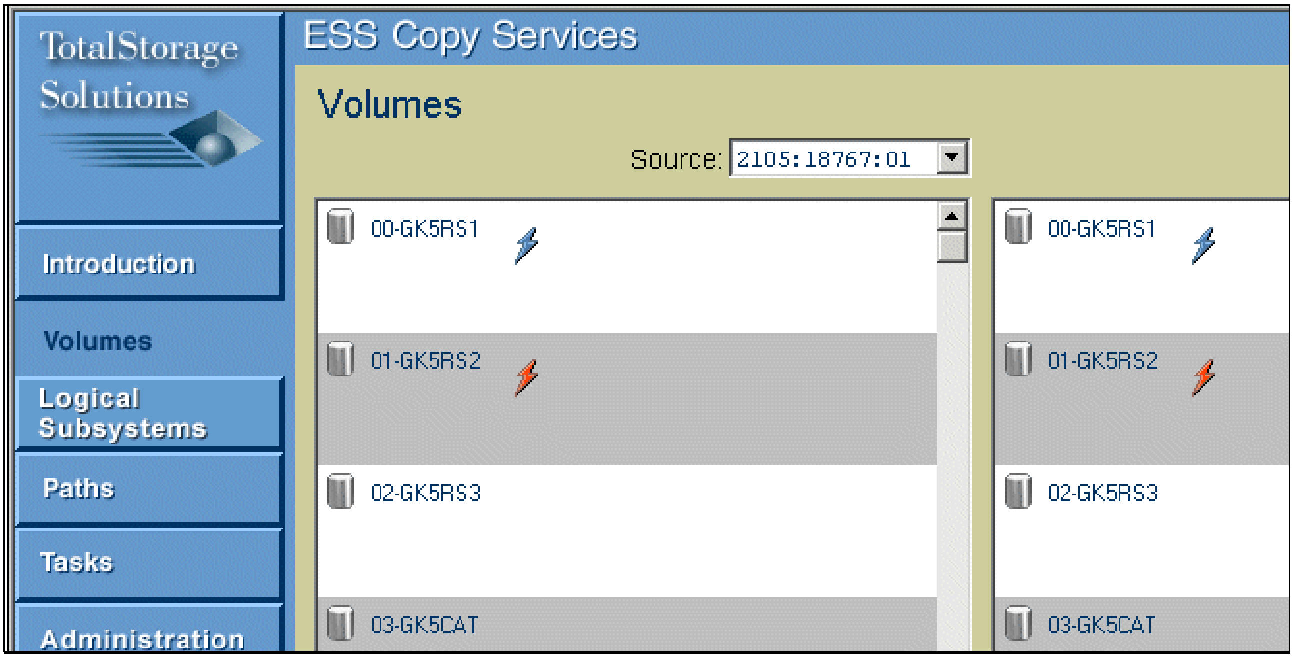



*Figure 8-46 Source and target volumes in FlashCopy relationship*

## 8.8.2 Withdrawing a FlashCopy relationship

To withdraw a FlashCopy relationship:

1. In the ESS Copy Services Volumes panel (see Figure 8-43), select the FlashCopy source volume.
2. Then right-click the target volume.
3. Right-click the target volume again to bring up the Select Task Type wizard panel.
4. Select **Withdraw FlashCopy pair** and click **Next**.
5. At the Select Withdraw Options panel click **Next;** we did not select any options here.
6. Enter the name and description of the task at the Define Task panel, and either click **Save** to save the defined task, or click **Run** to execute the task.

Note that the procedure to withdraw the FlashCopy pair can be done differently, by right-clicking again on the source volume (in step 2).

## 8.8.3 Persistent FlashCopy relationship

The Persistent FlashCopy option (described in “Persistent FlashCopy relationship” on page 16) is invoked only through the ESS Copy Services Web User Interface by selecting the **Persistent FlashCopy** option on the Select Copy Options panel (see Figure 8-47) when establishing the FlashCopy relationship.

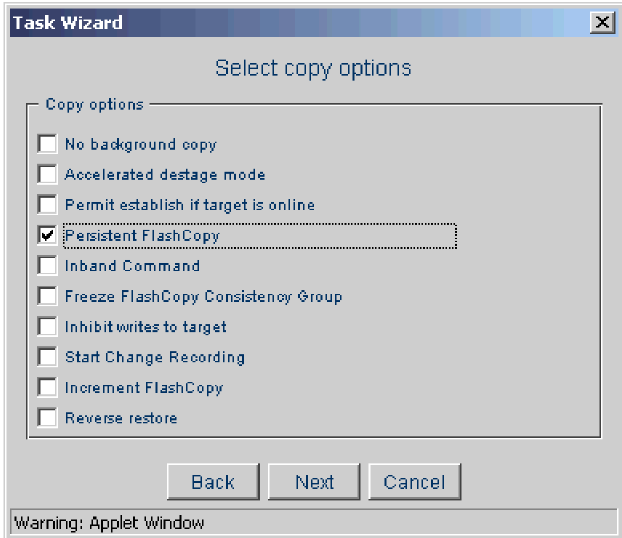



*Figure 8-47 Specifying Persistent FlashCopy option*

### 8.8.4 FlashCopy no-background copy to background copy

Changing a FlashCopy relationship from no-background copy to background copy (explained in, "FlashCopy no-background copy to background copy" on page 17) can be done only using the ESS Copy Services WUI. This is done in the following way:

1. From the ESS Copy Services Volumes panel (shown in Figure 8-43), select the FlashCopy source volume.
2. Right-click the source volume (do not select target volume).
3. Right-click again to bring up the Select Task Type panel (Figure 8-43).
4. Select **Withdraw FlashCopy pair** and click **Next**.
5. At the Select Withdraw Options panel, select **FlashCopy Start Background Copy** as shown in Figure 8-48, and then click **Next**.
6. Enter the name and description of the task at the Define Task panel, and either click **Save** to save the defined task, or click **Run** to execute the task.

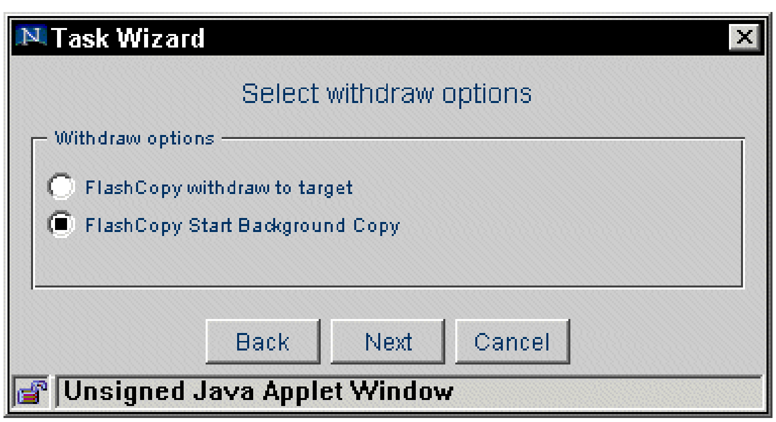



*Figure 8-48 Initiating FlashCopy no-background copy to background copy*

## 8.8.5 FlashCopy solutions

This section briefly describes how the FlashCopy V1 function may be used to enhance database backup. For more information and examples of solutions using FlashCopy, refer to the redbook, *IBM TotalStorage Solutions for Disaster Recovery,* SG24-6547.

### SAP R/3 split mirror backup/recovery

SAP's split mirror backup integrates database, hardware, and R/3 functionality in a single comprehensive solution: the duplication of source files by means of mirrored disks. Besides the already known benefits, this Split Mirror Backup/Recovery solution delivers *serverless* backup and recovery capability using the ESS advanced functions. After the copy is finished, the connection between source and mirror is split off, the mirror is mounted to another host and can be used for backup and recovery purposes.

During the backup process, the R/3 production system never stops — it continues its task of servicing user requests. This *zero* downtime for the R/3 production server means that SAP users do not miss a beat while the backup takes place. No transactions are canceled during the copy or backup processes, which takes only a few minutes. The *instant* availability of consistent copies of the database provides the ability to recover rapidly in the event of a disaster. Additional *near instant* and consistent copies of the production database provide important additional uses for BW Info Cube (an Info Cube is the primary data structure used in the implementation of a data warehouse), training, testing, and remote data vaulting purposes. This is a unique strength of this solution.

At its core is a special DB2 function, created by IBM, that enables a controlled suspension or *freeze* of the DB2 update activity — instead of having to stop the entire DB2 system — while copies are being made. It enables the DB2 subsystem to recover to a point-in-time without having to experience an extended recovery outage, or without having to stop or quiesce the primary system. The *writes* are throttled down for a few seconds while the FlashCopy function provides near-instant copies of the database. The physical copy is created under-the-covers without any effect on the user, the primary production server, or the primary ESS subsystem. For further details, see “Using FlashCopy with DB2” on page 463. Depending upon scalability and availability requirements, this solution can easily be implemented using two ESS subsystems providing data mirroring, data vaulting, and Disaster Recovery/business continuance capabilities for mission critical environments.

### Using FlashCopy with DB2

Most 24x7 DB operation and Very Large DB (VLDBs) require a mirroring solution. The SAP solution requires the ability to provide a full implementation for online split mirror handling. It requires the ability to split off a consistent mirror of a database, while OLTP keeps on running on the primary (live) database. Only a limited performance hit is acceptable. Recovery on the split off database is acceptable for consistency. In fact, nobody can afford doing offline or online backups on a 1 TB live database. Backups and system copies can be done from a mirror image.

So, an option has been added to DB2 which will temporarily *freeze* updates to a DB2 subsystem, while the logs and database can be copied using FlashCopy. The copies can be sent — using PPRC for example — to a remote site backup system. The remote site could then be restarted quickly in the event of a disaster at the primary site, and DB2 restart would rollback any in-flight units of recovery at the time the copies were made, bringing the subsystem to a point of consistency.

> **Note:** This option is available in DB2 V6.1 (with PTFs) and subsequent releases.

This option allows you to be able to temporarily *freeze* or *suspend* DB2 update activity, instead of having to stop the DB2 subsystem, while the copies are being made. This will allow you to recover the DB2 subsystem to a point-in-time without having to experience an extended recovery outage, or without having to stop or quiesce the primary system.

We suggest that you avoid using this function while long running units of recovery are active. DB2 restart time is lengthened by long running updates.

The new options are added to the SET LOG command to be able to suspend and resume logging for a DB2 subsystem. When a suspend request is issued:

- A system checkpoint will be taken (in a non-data sharing environment).
- Any unwritten log buffers will be written to DASD.
- The BSDS will be updated with the high- written RBA.
- And the log-write latch is obtained to prevent any further log records from being created.

This will prevent any further updates to the database until update activity is resumed. The latch will be held until a SET LOG command is issued to resume logging, or until a STOP DB2 command is issued. The scope for these commands is single-subsystem only, so the commands will have to be entered for each member when running in a data sharing environment.

A highlighted message will be issued showing that logging has been suspended. The message will be deleted when logging has been resumed. The output from a DISPLAY LOG command will also indicate that logging is suspended when a SET LOG SUSPEND command is active. Read-only activity should be able to continue while update activity is suspended provided there is no other resource contention.

You can think of the SET LOG SUSPEND command as a PAUSE button for DB2. It causes DB2 to stop processing, so that you can backup everything, but it does *not* flush out the changed data pages from the virtual buffer pools to DASD.

The implementation of DB2 suspend/resume would be a fuzzy backup, in that the header page could be written to disk, but the data page writes might still be in progress at the time of the copy. This function would allow you to restart DB2 normally at the recovery site, but you could not use it with log-only recovery (you could have a step to copy the logs and BSDS and do a RECOVER LOGONLY). This process is intended to work as follows:

1. Suspend, FlashCopy the entire system, resume.
2. At the recovery site, restore everything from step 1 and restart.

## 8.8.6 SnapShot to FlashCopy migration

SnapShot is the virtual duplication function of the IBM RAMAC® Virtual Array (RVA) that enables the user to make almost immediate copies of data. Current SnapShot users planning to migrate to ESS FlashCopy can reference Appendix C, “Migrating from RVA to ESS” on page 367 for detailed information and examples for doing the migration.

## 8.8.7 Geographically Dispersed Parallel Sysplex (GDPS)

How would an outage of the system affect the business? Are system maintenance and upgrades being put off to avoid system downtime? What about a site disaster? Is the business-critical processing and data protected from a site disaster? In today’s highly competitive e-business world, outages will have a devastating impact on a business; they could mean its demise.

Many companies have inadequate *business continuance* plans developed on the premise that back office, and manual processes will keep the business running until computer systems are available. Characteristics of these recovery models allow critical applications to recover within 24 to 48 hours, with data loss potentially exceeding 24 hours, and full business recovery taking days or weeks. As companies transform their business to compete in the e-marketplace, business continuance strategies and availability requirements must be re-evaluated to ensure that they are based on today's business requirements.

In e-business, two of the most stringent demands are *high-availability/continuous-operation* and near *transparent Disaster Recovery* (D/R or *business continuity*). Continuously available systems combine the characteristics of *high availability* and *continuous operations* to always deliver the highest levels of service (24x7x365). High availability is the attribute of a system to provide service at agreed-upon levels and mask *unplanned outages* from end users. Continuous operation, on the other hand, is the attribute of a system to continuously operate and mask *planned outages* from end users. To attain the highest levels of high-availability/continuous-operation and near-transparent D/R, the solutions must be automated and based on geographical clusters and data mirroring. These technologies are the backbone of the GDPS solution.

## GDPS in a PPRC environment

One of the main concerns in any synchronous hardware based data mirroring implementation, like PPRC, is how to maintain a *time-consistent* image of all data in the secondary site, particularly in case of unplanned outages. Time-consistent means that the secondary disks contain all updates until a specific point-in-time, without anything missing, and no updates beyond that point.

In order to maintain data consistency on the secondary site during an unplanned outage, it is necessary to *freeze* data mirroring by means of some automated solution as soon as the very first error is detected. GDPS is an integrated solution for server, workload, and network management that provides an automated solution for remote data management. It will actually prevent loss of logical integrity of the data at the recovery site, avoiding all the errors that are expected during a rolling disaster to be propagated to the recovery site. Such errors would prevent a fast and efficient recovery.

The fact that the data image at the recovery site remains time-consistent (not corrupted logically by the errors that occurred at the primary site), allows the applications to be emergency restarted at the recovery site, without the need to go through a lengthy data recovery process.

GDPS provides an automated solution for preserving consistency at the recovery site. IBM provides GDPS as a service offering. It is discussed in more detail in Appendix B, "Geographically Dispersed Parallel Sysplex (GDPS)" on page 359.

## Error recovery in a GDPS environment

GDPS monitors many types of error events, the most significant category being changes in the remote copy configuration. This category is so important because this type of error may affect the time-consistency of the recovery data. Figure 8-49 shows a simplified example of how GDPS automation could handle an error situation.

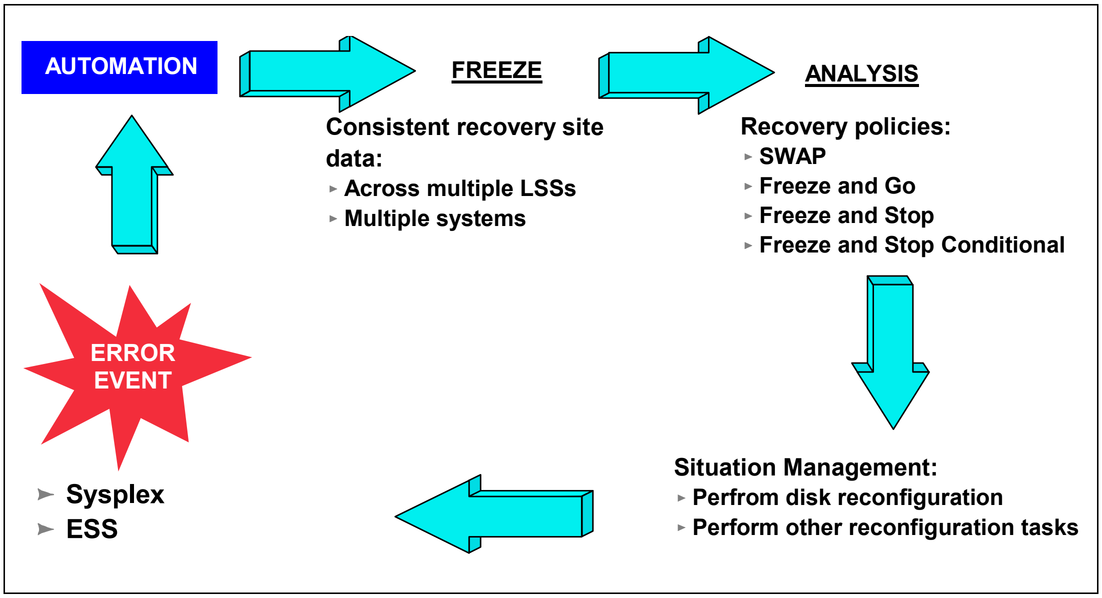



*Figure 8-49 Automated procedure for keeping consistency at the recovery site*

When a volume pair is suspended because of an error condition, the automation routines will immediately freeze the image of all the required secondary volumes by freezing all the application-related LSSs that are known to GDPS. This will preserve the consistency of the recovery data at that specified point-in-time. This action is followed by an analysis step. The analysis is driven by a pre-determined policy, which can be either of the following:

- SWAP
- FREEZE and GO
- FREEZE and STOP
- FREEZE and STOP CONDITIONAL.

### *SWAP*

With this policy, automation routines swap primary DASD transparently from one site to another without requiring an application or system outage, since swapping is executed before an error condition is reported to the application. GDPS HyperSwap™ function is described in 8.8.8, “GDPS/PPRC HyperSwap” on page 467.

### *FREEZE and GO*

With this policy, secondary PPRC volumes are suspended but automation routines allow applications to continue, thus updates of the primary volumes proceed. However, these updates will not be reflected on the secondary PPRC volumes, and this is an exposure in case there is a subsequent application site failure. But, the availability of the application is favored, as it is not stopped by this error event.

### FREEZE and STOP

With this policy, the automation routines stop all applications right where they are at the time of freezing the volumes pairs. Application availability will be impacted, but the currency of the recovery site data is preserved. This policy guarantees no data loss and complete data integrity on both primary and secondary PPRC volumes.

### FREEZE and STOP CONDITIONAL

With this policy, the automation routines freeze the image of the secondary volumes, and then determine the reason for the suspend condition. If the freeze was caused because of hardware problems on the secondary site equipment, then the FREEZE and GO policy is applied, and thus applications will be allowed to continue. Otherwise, the automation routines apply the FREEZE and STOP policy, thus stopping all the required applications.

## 8.8.8 GDPS/PPRC HyperSwap

The GDPS/PPRC HyperSwap function is designed to broaden the continuous availability attributes of GDPS/PPRC. This function can help significantly increase the speed of switching sites and switching disks between sites. The HyperSwap function is designed to be controlled by complete automation, allowing all aspects of the site switch to be controlled through GDPS.

The HyperSwap function provides the ability to transparently switch the applications I/O activity to the secondary PPRC volumes for both planned and unplanned reconfiguration. Large configurations can be supported as HyperSwap has been designed to provide the swap of large number of volumes very quickly. The important ability to re-synchronize incremental disk data changes in both directions between primary and secondary PPRC disks is provided as part of this function. Figure 8-50 summarizes some other characteristics of GDPS/HyperSwap.

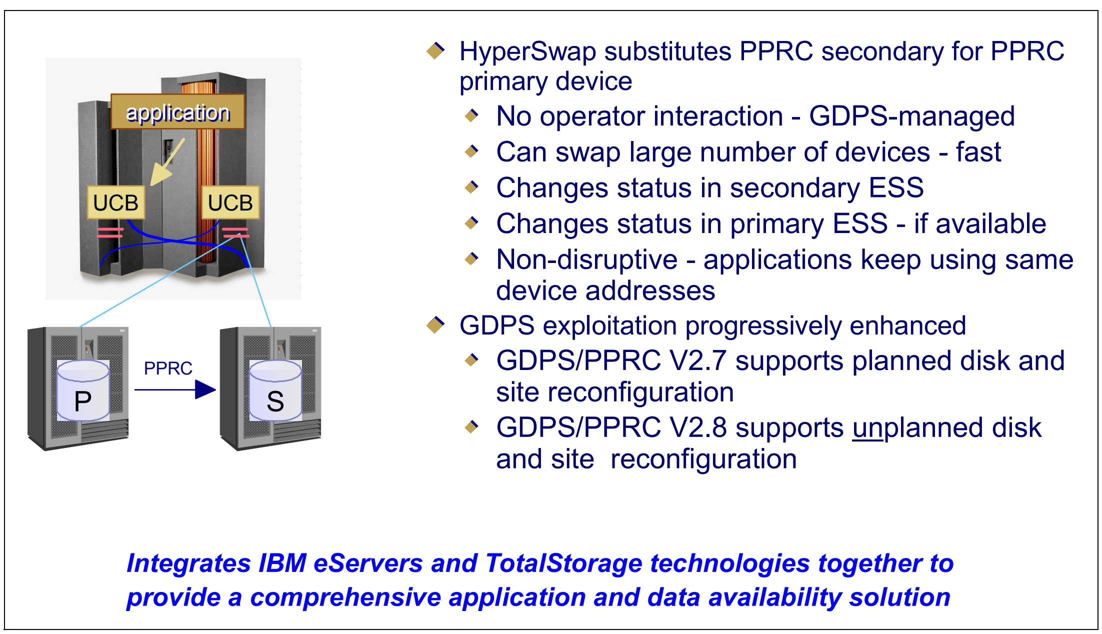



*Figure 8-50 GDPS/PPRC HyperSwap*

### 8.8.9 Remote Copy Management Facility: RCMF

The Remote Copy Management Facility (RCMF) is designed to simplify the storage administrator's remote copy management functions by managing the remote copy configuration rather than individual remote copy pairs. Storage management tasks include the initialization and monitoring of the PPRC volume pairs on the basis of policy and performing routine operations on installed storage subsystems. RCMF does not manage secondary consistency.

RCMF can be useful in managing the ESS PPRC environment because it:

- Provides a central point of control through full screen, tree-structured panels
- Provides a vehicle for initializing and maintaining the remote copy configuration
- Provides status monitoring and exception reporting
- Drives P/DAS for device migration or planned outages

RCMF is part of the GDPS service offering and is provided by IBM Global Service for both PPRC and XRC forms of remote copy. For more information related to RCMF contact your IBM representative.

IBM Redbooks

# IBM TotalStorage Enterprise Storage Server: Implementing ESS Copy Services with IBM eServer zSeries

(1.0” spine)
0.875”<->1.498”
460 <-> 788 pages

# IBM TotalStorage Enterprise Storage Server: Implementing ESS Copy Services with IBM eServer zSeries

**Understand the characteristics of the powerful ESS Copy Services functions**

**Learn how PPRC, FlashCopy, XRC, and Concurrent Copy work**

**Useful information for effective implementation**

This IBM Redbook describes the copy functions available with the IBM TotalStorage Enterprise Storage Server (ESS). The powerful ESS Copy Services functions are explained in detail, and their characteristics are thoroughly covered. This redbook also provides information on how to manage the various ESS Copy Services functions, and discusses their implementation.

This redbook applies to the ESS Models when used in the IBM eServer zSeries environments. Because this book provides a broad understanding of the ESS Copy Services functions, as well as presenting details about the management interfaces and the implementation considerations, it is a recommended manual for IT professionals who are planning the implementation of any of the ESS Copy Services functions in a zSeries environment, as well as for those who will manage these environments.

This fifth edition of the redbook has been updated with the latest ESS Copy Services announcements: Asynchronous PPRC can be used for a two site Disaster Recovery solution, a FlashCopy target can now be a PPRC primary, and there are several other updates associated with ESS LIC 2.4.0. These include the ESS Model 750 overview, minor additions for ESS API support, Multiple Device Manager Replication Manager, Asynchronous PPRC Utilities for Open System Environment, and Asynchronous PPRC Utilities for ICKDSF Users.

**INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION**

**BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE**

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

**For more information:**
**ibm.com**/redbooks

SG24-5680-04 ISBN 0738490296