Dell EMC™ Mainframe Enablers
TimeFinder™/Mirror for z/OS

Version 8.0 and higher

Product Guide

REV 06
CONTENTS

Preface

Chapter 1  Introduction
Mainframe Enablers and TimeFinder/Mirror ............................................. 16
Licensing ........................................................................................................ 16
Introduction to TimeFinder/Mirror ............................................................... 17
Advantages ..................................................................................................... 17

Chapter 2  Getting Started
Enabling TF/Clone and TF/Consistency Group ........................................... 20
Starting ResourcePak Base ........................................................................ 20
Setting up security ...................................................................................... 21
Running TF/Mirror ..................................................................................... 21
Running TF/Mirror recovery jobs ............................................................... 23
Configuring TF/Mirror ............................................................................... 23
Setting defaults with site options ................................................................. 24
Site option overview .................................................................................. 24
Default site options .................................................................................... 25
Customizing site options ........................................................................... 26
Locking site options .................................................................................... 27
Viewing site options and overrides ........................................................... 28

Chapter 3  Replicating Data
Replication environment ............................................................................. 34
Basic configuration ..................................................................................... 34
SRDF configurations ................................................................................ 34
Replication process .................................................................................... 35
Provisioning BCVs .................................................................................... 36
Setting global parameters .......................................................................... 37
Synchronizing from STD to BCV: ESTABLISH ......................................... 37
Stopping synchronization: SPLIT .............................................................. 42
Re-synchronizing from STD to BCV: RE-ESTABLISH .............................. 47
Synchronizing from BCV to STD: RESTORE ........................................... 48
Deleting the STD/BCV pair ....................................................................... 51
Replication technology ............................................................................... 52
Clone emulation ........................................................................................ 52
Supported device types ............................................................................. 54
GCM support ............................................................................................. 54
Online/offline status check ....................................................................... 55
Device state change ................................................................................... 55
Device locks ............................................................................................... 55
Multi-attach ............................................................................................... 56
Multitasking ............................................................................................... 56
TF/Mirror with SRDF ................................................................................ 58
R1 is STD ................................................................................................ 58
R1 is BCV ................................................................................................. 59
R2 is STD ................................................................................................ 60
Contents

R11 is STD ................................................................. 61
R21 is STD ................................................................. 62
R22 is STD ................................................................. 63
TF/Mirror operations ..................................................... 64
  Basic operations ...................................................... 64
  Viewing BCVs ............................................................ 64
  Managing device status ............................................... 64
  Managing device holds ................................................ 64
  Setting up clone emulation ........................................... 65
  Setting up Multi-Attach ............................................... 65
  Setting up Concurrent BCVs ......................................... 65
  Setting up Protected BCVs ........................................... 66
  Setting up a SPLIT ..................................................... 66
  Setting up Consistent SPLITs ........................................ 67
  Synchronizing R1/R2 ................................................... 68
  Updating volser information ........................................ 68
  Waiting for synchronization completion and timeouts ......... 69
Miscellaneous .......................................................... 70
TF/Mirror queries ....................................................... 71

Chapter 4  Automating Replication with SRDF/AR

Introduction to SRDF/AR .................................................. 74
SRDF/AR environment ..................................................... 75
  Single-hop configuration ............................................. 75
  Multihop configuration ............................................... 76
  Initial configuration requirements ................................. 77
Creating an SRDF/AR environment .................................. 78
SRDF/AR storage estimates ............................................... 79
SRDF/AR processes and cycles ......................................... 80
  Step 0: Health check ................................................... 81
  Step 1A: Ensure R1 BCVs are offline ............................... 82
  Step 1B: Ensure BCVs paired to R2 STDs are offline .......... 82
  Step 1C: Check R1-R2 invalid tracks and status ................ 82
  Step 1D: Check R2 STD local mirrors ............................. 82
  Step 1E: Check R1 BCV status ....................................... 82
  Step 2: Consistent SPLIT of R1 BCVs ............................... 83
  Step 3: Wait for SRDF sync of R1s with R2s ..................... 84
  Step 4: Check if R1 BCVs are Ready ............................... 85
  Step 5: RE-ESTABLISH R1 BCVs .................................... 85
  Step 6: Wait for mirror sync on BCVs paired to R2 STDs ...... 85
  Step 7: RE-ESTABLISH BCVs paired to R2 STDs .............. 86
  Step 8: Wait for TF/Mirror sync of R2 STDs ..................... 86
  Step 9: SPLIT BCVs paired to R2 STDs ........................... 87
  Step 10: Wait for mirror sync on R1 BCVs ....................... 87
  Step 10b: Wait for mirror sync on BCVs paired to R2 STDs .. 87
  Step 11: Cycle-end health check ..................................... 87
  Step 11b: Wait for mirror sync on BCVs paired to R2 STDs .. 87
SRDF/AR operations ..................................................... 88
  Managing SRDF/AR processes ..................................... 88
  Specifying devices for SRDF/AR process ......................... 88
  Setting up SRDF/AR environment .................................. 88
  Setting up cycle ....................................................... 89
  Setting up cycle steps ................................................. 89

TF/Mirror for z/OS 8.0 and Higher Product Guide
FIGURES

1 Customized site options ........................................................................................................... 27
2 Site Options for TF/Mirror report—Basic format .................................................................. 29
3 Site Options for TF/Mirror report—Extended format ............................................................. 31
4 Basic TF/Mirror configuration ............................................................................................. 34
5 TF/Mirror process flow .......................................................................................................... 35
6 Initial configuration under Enginuity 5773 ......................................................................... 37
7 Assigning the BCV as a mirror of the STD ........................................................................... 38
8 Establishing an STD/BCV pair .............................................................................................. 39
9 Concurrent BCV ..................................................................................................................... 40
10 Restoring an STD/BCV pair .................................................................................................. 48
11 R1 is STD ............................................................................................................................. 58
12 R1 is BCV ............................................................................................................................ 59
13 R2 is STD ............................................................................................................................ 60
14 R11 is STD .......................................................................................................................... 61
15 R21 is STD .......................................................................................................................... 62
16 R22 is STD .......................................................................................................................... 63
17 Single-hop SRDF/AR configuration ...................................................................................... 75
18 Multihop SRDF/AR configuration ...................................................................................... 76
19 Initial single-hop SRDF/AR configuration .......................................................................... 77
20 SRDF/AR process life cycle ................................................................................................ 80
21 Consistent SPLIT of R1 BCVs ............................................................................................ 83
22 Waiting for R1 BCVs to synchronize with R2 STDs .......................................................... 84
23 R1 BCVs synchronized with R2 STDs ................................................................................ 84
24 RE-ESTABLISH of R1 BCVs ............................................................................................... 85
25 RE-ESTABLISH of BCVs paired to R2 STDs ..................................................................... 86
26 SPLIT of BCVs paired to R2 STDs ...................................................................................... 87
27 Device configuration for SRDF/AR example 1 ................................................................. 91
28 Device configuration for SRDF/AR example 6 ................................................................. 93
29 Dell EMC SRDF/AR Monitor panel ................................................................................... 107
30 Executing SARTOOL REXX ............................................................................................. 109
31 Specify the Session Parameters panel ............................................................................... 110
32 Dell EMC SRDF/AR Function Controls panel .................................................................... 111
33 DOIO error code format ...................................................................................................... 193
<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Default site options</td>
<td>25</td>
</tr>
<tr>
<td>2</td>
<td>Command mapping in clone emulation</td>
<td>52</td>
</tr>
<tr>
<td>3</td>
<td>Online/offline status check</td>
<td>55</td>
</tr>
<tr>
<td>4</td>
<td>Basic TF/Mirror operations</td>
<td>64</td>
</tr>
<tr>
<td>5</td>
<td>Viewing BCVs</td>
<td>64</td>
</tr>
<tr>
<td>6</td>
<td>Managing device status</td>
<td>64</td>
</tr>
<tr>
<td>7</td>
<td>Managing device holds</td>
<td>65</td>
</tr>
<tr>
<td>8</td>
<td>Setting up clone emulation</td>
<td>65</td>
</tr>
<tr>
<td>9</td>
<td>Setting up the Multi-Attach feature</td>
<td>66</td>
</tr>
<tr>
<td>10</td>
<td>Managing Concurrent BCVs</td>
<td>66</td>
</tr>
<tr>
<td>11</td>
<td>Setting up Protected BCVs</td>
<td>67</td>
</tr>
<tr>
<td>12</td>
<td>Setting up a SPLIT</td>
<td>67</td>
</tr>
<tr>
<td>13</td>
<td>Setting up Consistent SPLITs</td>
<td>68</td>
</tr>
<tr>
<td>14</td>
<td>Synchronizing R1/R2</td>
<td>68</td>
</tr>
<tr>
<td>15</td>
<td>Updating volser information</td>
<td>68</td>
</tr>
<tr>
<td>16</td>
<td>Setting up TF/Mirror</td>
<td>69</td>
</tr>
<tr>
<td>17</td>
<td>Waiting for completion and timeouts</td>
<td>69</td>
</tr>
<tr>
<td>18</td>
<td>Miscellaneous parameters</td>
<td>70</td>
</tr>
<tr>
<td>19</td>
<td>SRDF/AR control block defaults</td>
<td>79</td>
</tr>
<tr>
<td>20</td>
<td>Managing SRDF/AR processes</td>
<td>88</td>
</tr>
<tr>
<td>21</td>
<td>Specifying devices for SRDF/AR process</td>
<td>88</td>
</tr>
<tr>
<td>22</td>
<td>Setting up SRDF/AR environment</td>
<td>88</td>
</tr>
<tr>
<td>23</td>
<td>Setting up process cycle</td>
<td>89</td>
</tr>
<tr>
<td>24</td>
<td>Setting up cycle steps</td>
<td>89</td>
</tr>
<tr>
<td>25</td>
<td>Drive failure recovery procedures</td>
<td>98</td>
</tr>
<tr>
<td>26</td>
<td>Step-dependent information</td>
<td>101</td>
</tr>
<tr>
<td>27</td>
<td>SARTOOL key controls</td>
<td>109</td>
</tr>
<tr>
<td>28</td>
<td>SRDF/AR Process Information primary commands</td>
<td>111</td>
</tr>
<tr>
<td>29</td>
<td>SRDF/AR Process Information line commands</td>
<td>112</td>
</tr>
<tr>
<td>30</td>
<td>JCL to run reports in batch mode</td>
<td>113</td>
</tr>
<tr>
<td>31</td>
<td>Valid SRDF/AR process batch report options</td>
<td>113</td>
</tr>
<tr>
<td>32</td>
<td>Message prefixes</td>
<td>192</td>
</tr>
<tr>
<td>33</td>
<td>Abend codes</td>
<td>192</td>
</tr>
<tr>
<td>34</td>
<td>TF/Mirror reason codes</td>
<td>193</td>
</tr>
<tr>
<td>35</td>
<td>EXTENTS reason codes</td>
<td>198</td>
</tr>
</tbody>
</table>
PREFACE

As part of an effort to improve its product lines, Dell EMC periodically releases revisions of its software and hardware. Therefore, some functions described in this document might not be supported by all versions of the software or hardware currently in use. The product release notes provide the most up-to-date information about product features.

Contact your Dell EMC representative if a product does not function properly or does not function as described in this document.

Note: This document was accurate at publication time. New versions of this document might be released on the Dell EMC online support website. Check the Dell EMC online support website to ensure that you are using the latest version of this document.

Purpose

This document describes how to configure and use Dell EMC TimeFinder/Mirror for z/OS.

Coverage

This document describes TimeFinder/Mirror for z/OS in the following operating environments supported by Mainframe Enablers 8.0 and later:

- PowerMaxOS 5978
- HYPERMAX OS 5977
- Enginuity™ 5876
- Enginuity 5773

Note: See prior versions of the TimeFinder/Mirror for z/OS Product Guide for information pertaining to other Enginuity levels.

Audience

This document is intended for the host system administrator, system programmer, or operator who is involved in managing or operating TimeFinder/Mirror for z/OS.

Related documentation

To access related documentation, go to the PowerMax™ and VMAX™ All Flash Technical Documentation webpage at:


Note

Enginuity 5773 is not supported with Mainframe Enablers 8.3. For Mainframe Enablers 8.0, 8.1, and 8.2, Enginuity 5773 is not supported in SRDF configurations that include a storage system running PowerMaxOS 5978 or HYPERMAX OS 5977.
The following documents provide information about Mainframe Enablers:

- Mainframe Enablers Release Notes
- Mainframe Enablers Installation and Customization Guide
- Mainframe Enablers Messages Guide
- ResourcePak™ Base for z/OS Product Guide
- SRDF™ Host Component for z/OS Product Guide
- AutoSwap™ for z/OS Product Guide
- Consistency Groups for z/OS Product Guide
- TimeFinder™ SnapVX and zDP™ Product Guide
- TimeFinder/Clone Mainframe Snap Facility Product Guide
- TimeFinder/Mirror for z/OS Product Guide
- TimeFinder Utility for z/OS Product Guide

The following documents provide additional information:

- PowerMax Family Product Guide—Documents the features and functions of the PowerMax storage systems.
- PowerMaxOS for PowerMax and VMAX All Flash Release Notes—Describes new features and any known limitations.
- VMAX All Flash Product Guide—Documents the features and functions of the VMAX All Flash storage systems.
- HYPERMAX OS for VMAX All Flash and VMAX3 Family Release Notes—Describes new features and any known limitations.
- VMAX3 Family Product Guide—Documents the features and functions of the VMAX3 100K, 200K, and 400K storage systems.
- VMAX Family Product Guide—Documents the features and functions of the VMAX 10K, 20K, and 40K storage systems.
- E-Lab™ Interoperability Navigator (ELN)—Provides a web-based interoperability and solution search portal. You can find the ELN at elabnavigator.EMC.com.

Conventions used in this document

Dell EMC uses the following conventions for special notices:

⚠️ **CAUTION**

CAUTION, used with the safety alert symbol, indicates a hazardous situation which, if not avoided, could result in minor or moderate injury.

---

**Note:** A note presents information that is important, but not hazard-related.

---

**IMPORTANT**

An important notice contains information essential to software or hardware operation.
Typographical conventions

Dell EMC uses the following type style conventions in this document:

Normal
- Used in running (nonprocedural) text for:
  - Names of interface elements, such as names of windows, dialog boxes, buttons, fields, and menus
  - Names of resources, attributes, pools, Boolean expressions, buttons, DQL statements, keywords, clauses, environment variables, functions, and utilities
  - URLs, pathnames, filenames, directory names, computer names, links, groups, service keys, file systems, and notifications

Bold
- Used in running (nonprocedural) text for names of commands, daemons, options, programs, processes, services, applications, utilities, kernels, notifications, system calls, and man pages
- Used in procedures for:
  - Names of interface elements, such as names of windows, dialog boxes, buttons, fields, and menus
  - What the user specifically selects, clicks, presses, or types

Italic
- Used in all text (including procedures) for:
  - Full titles of publications referenced in text
  - Emphasis, for example, a new term
  - Variables

Courier
- Used for:
  - System output, such as an error message or script
  - URLs, complete paths, filenames, prompts, and syntax when shown outside of running text

Courier bold
- Used for specific user input, such as commands

Courier italic
- Used in procedures for:
  - Variables on the command line
  - User input variables

< >
- Angle brackets enclose parameter or variable values specified by the user

[]
- Square brackets enclose optional values

|
- Vertical bar indicates alternate selections—the bar means “or”

{}
- Braces enclose content that the user must specify, such as x or y or z

...
- Ellipses indicate nonessential information omitted from the example

Where to get help

Dell EMC support, product, and licensing information can be obtained on the Dell EMC Online Support site as described next.

Note: To open a service request through the Dell EMC Online Support site, you must have a valid support agreement. Contact the Dell EMC sales representative for details about obtaining a valid support agreement or to answer any questions about your account.
Product information

For documentation, release notes, software updates, or for information about Dell EMC products, licensing, and service, go to the Dell EMC Online Support site (registration required) at:

support.EMC.com

Technical support

Dell EMC offers a variety of support options.

Support by Product—Dell EMC offers consolidated, product-specific information at:

support.EMC.com/products

The Support by Product web pages offer quick links to Documentation, White Papers, Advisories (such as frequently used Knowledgebase articles), and Downloads, as well as more dynamic content, such as presentations, discussion, relevant Customer Support Forum entries, and a link to Dell EMC Live Chat.

Dell EMC Live Chat—Open a Chat or instant message session with a Dell EMC Support Engineer.

eLicensing support

To activate your entitlements and obtain your license files, visit the Service Center on support.EMC.com, as directed on your License Authorization Code (LAC) letter emailed to you.

For help with missing or incorrect entitlements after activation (that is, expected functionality remains unavailable because it is not licensed), contact your Dell EMC Account Representative or Authorized Reseller.

For help with any errors applying license files through Solutions Enabler, contact Dell EMC Customer Support.

If you are missing a LAC letter, or require further instructions on activating your licenses through the Online Support site, contact Dell EMC's worldwide Licensing team at licensing@emc.com or call:

- North America, Latin America, APJK, Australia, New Zealand: SVC4EMC (800-782-4362) and follow the voice prompts.
- EMEA: +353 (0) 21 4879862 and follow the voice prompts.

Your comments

Your suggestions will help us continue to improve the accuracy, organization, and overall quality of the user publications. Send your opinions of this document to:

VMAXContentFeedback@emc.com
CHAPTER 1
Introduction

This chapter covers the following topics:
◆ Mainframe Enablers and TimeFinder/Mirror......................................................... 16
◆ Introduction to TimeFinder/Mirror ........................................................................ 17
Mainframe Enablers and TimeFinder/Mirror

Dell EMC TimeFinder™/Mirror is one of the Dell EMC Mainframe Enablers. The Dell EMC Mainframe Enablers include the following components that you can use to monitor and manage your storage:

- ResourcePak™ Base for z/OS
- SRDF™ Host Component for z/OS
- AutoSwap™ for z/OS
- Consistency Groups for z/OS
- TimeFinder SnapVX
- Data Protector for z Systems (zDP™)¹
- TimeFinder/Clone Mainframe Snap Facility
- TimeFinder/Mirror for z/OS
- TimeFinder Utility

When you install the Mainframe Enablers, you install the software for all of these components.

Licensing

See the following documents for information about licensing:

- Mainframe Enablers Installation and Customization Guide

**Note:** Since Mainframe Enablers 8.3 do not support Enginuity 5773, for information about licenses required under Enginuity 5773 see the Mainframe Enablers Installation and Customization Guide for Mainframe Enablers 8.0, 8.1, or 8.2.

- PowerMax Family Product Guide
- VMAX All Flash Product Guide
- VMAX3 Family Product Guide
- VMAX Family Product Guide

¹. zDP requires TimeFinder SnapVX but is a separately licensed product.
Introduction to TimeFinder/Mirror

TimeFinder/Mirror is a local replication solution designed to ensure business continuance. TF/Mirror enables you to make full-volume copies of production data from a storage system device to a Business Continuance Volume (BCV). The BCV can then be separated from the standard device and used for backup, restore, decision support, or applications testing. Afterwards, the BCV can again be mirrored to the same or a different device.

TF/Mirror can be used in SRDF configurations to provide replication across SRDF links on remote storage systems.

With SRDF/Automated Replication (SRDF/AR), you can automate SRDF replication to provide a logically consistent, restartable image of the data at a remote (recovery) site in the event of a disaster at the production site.

Advantages

◆ You can create dependent write consistent copies of data locally or remotely without the need to interrupt production jobs.

◆ You can use BCVs as the source for the following:
  ■ Backup operations
  ■ Data warehousing applications
    Because BCVs are a point-in-time mirror image of the production data, they can be used as “gold” copies of data to be written and rewritten repeatedly.
  ■ Application testing based on real data
    The speed with which a BCV can be reconstituted means that multiple test cycles can occur rapidly and sequentially. Applications can be staged using BCVs before committing them to the next application refresh cycle.

◆ Compatible with mainframe security mechanisms such as RACF

◆ Integration with database management system (DBMS) utilities available from several Independent Software Vendors (ISVs) and their products

◆ Integration with many mainframe-specific ISVs and their products
Introduction
CHAPTER 2
Getting Started

This chapter covers the following topics:

- Enabling TF/Clone and TF/Consistency Group .................................................... 20
- Starting ResourcePak Base .................................................................................. 20
- Setting up security ............................................................................................... 21
- Running TF/Mirror .............................................................................................. 21
- Configuring TF/Mirror ......................................................................................... 23
- Setting defaults with site options ........................................................................ 24
Enabling TF(Clone) and TF/Consistency Group

To use TF/Mirror with PowerMaxOS 5978, HYPERMAX OS 5977 or Enginuity 5876, you need to install Mainframe Enablers and enable the TF(Clone) component of TF(Clone) Mainframe Snap Facility. For Consistent SPLIT operations, you have to enable the TF/Consistency Group component as well.

Under Enginuity 5773, TF/Mirror and TF/Consistency Group are required when you use the clone emulation mode.

License requirements

With Enginuity 5773, TF/Mirror issues error message BCVM142E (as well as possible operation-specific messages) and does not execute if the License Feature Code keys for TF/Mirror and TF/Consistency Group (in case of a Consistent SPLIT) have not been installed in the initialization parameters file of ResourcePak Base.

With PowerMaxOS 5978, HYPERMAX OS 5977 and Enginuity 5876, the eLicense entitlement for TF(Clone) only needs to be checked to ensure that it is enabled. The license for TF/Consistency Group is still required if you are performing Consistent SPLIT operations.


Security

TF(Clone) Mainframe Snap Facility can provide enhanced SAF security checking using the XFACILIT resource class. This means that TF/Mirror operations that invoke clone emulation may be affected.

Note: The Mainframe Enablers Installation and Customization Guide provides more information about enhanced security checking.

Starting ResourcePak Base

TF/Mirror requires that ResourcePak Base (SCF) is running. Before using TF/Mirror, start ResourcePak Base as described in the ResourcePak Base for z/OS Product Guide.

By default, TF/Mirror expects that ResourcePak Base runs as a task named EMC$SCF. If your instance of ResourcePak Base has another name, specify it in the SCF$nnnn DD statement of the TF/Mirror job.

If ResourcePak Base is not running, issuing any TF/Mirror command results in an error message stating that EMCSCF is not available.
Setting up security

TF/Mirror uses SAF calls to validate access to resources. This feature is turned on by default. The source code for the TF/Mirror SAF interface routine is provided in the SAMPLIB library of ResourcePak Base, so that you can tailor it to your specific needs.

Note: The Mainframe Enablers Installation and Customization Guide describes the Security Interface and the class and resource names used.

Running TF/Mirror

You normally run TF/Mirror as a batch job. The JCL is as follows:

```plaintext
//EMCTF EXEC PGM=EMCTF,REGION=0M,PARM=parameters
//SYSOUT DD SYSOUT=*  
//SCF$nnnnn DD DUMMY  
//EMCQCAPI DD SYSOUT=*  
//SYSIN DD *  
```

TF/Mirror commands

**EXEC parameters**

**PGM**

The main program of TF/Mirror is EMCTF.

**REGION**

The region size.

**PARM**

See “PARM parameters” on page 21.

**PARM parameters**

**ECACLEAR**

Clears ECA 1 on all STDs in a Consistent SPLIT referenced in the input stream.

Note: “Replication process” on page 35 describes STDs. “Consistent SPLIT” on page 43 describes Consistent SPLITs.

This parameter can be abbreviated as ECACL.

**MAXREQ=nnnnn**

Specifies the maximum number of TF/Mirror commands in SYSIN. The default value is 16384. The maximum allowable value is 65536.

**RELEASEDEVICELOCK**

Releases device locks for all BCVs referenced in the input stream. This parameter is not allowed for an active SRDF/AR process.

Note: “Automating Replication with SRDF/AR” on page 73 describes SRDF/AR.

---

1. ECA stands for Enginuity Consistency Assist, a Dell EMC technology used to control I/O. See Consistency Groups for z/OS Product Guide for more information about ECA.
This parameter can be abbreviated as REDLOCK.

RESETIOSLEVEL

Resets the IOS level for all STDs in a Consistent SPLIT referenced in the input stream, when ECA is not used.

Note: “Replication process” on page 35 describes STDs. “Consistent SPLIT” on page 43 describes Consistent SPLITs.

This parameter can be abbreviated as RIOSL.

SAR_PAUSED(process-name)

Bypasses device locks when the SRDF/AR process is paused, allowing TF/Mirror operations to be run on the devices.

Note: “Automating Replication with SRDF/AR” on page 73 describes SRDF/AR.

The SAR_PAUSED parameter is used in a recovery situation, for example, after a drive failure, to restore devices to their proper state before continuing the SRDF/AR process.

DD statements

SCF$nnnn

Identifies the ResourcePak Base task that the TF/Mirror job runs against.

The SCF$nnnn statement is optional. If you do not specify it, the default ResourcePak Base task name SCF$EMC is used.

EMCQCAPI

Collects TF/Clone Mainframe Snap interface (EMCQCAPI) messages when the debugging is enabled clone operations using the DEBUG(CLONE) option of the GLOBAL command, as described in “GLOBAL” on page 130, and the operation is processed through the clone emulation mode.

SYSIN

Contains the TF/Mirror commands. The SYSIN DD statement can reference a dataset: DISP=SHR, DSN=dataset. The maximum number of TF/Mirror commands is defined by the MAXREQ statement, as described in “MAXREQ=nnnnn” on page 21. You can use up to 128 different sequence levels.

TFSITEO

(Optional) When specified, the Site Options for TF/Mirror extended report lines are redirected to this DD.

Note: “Viewing site options and overrides” on page 28 discusses the Site Options for TF/Mirror report.
Running TF/Mirror recovery jobs

When recovering from a job or system failure, you can use certain TF/Mirror PARM parameters, such as ECACLEAR, RELEASEDEVICELOCK, and RESETIOSLEVEL, to clear or reset the devices to their original (normal) state.

The input command stream must be the same as in the failed job. However, the statements in TF/Mirror recovery jobs are not executed as regular TF/Mirror commands. They are used solely to determine the devices which need to be reset or cleared.

After the recovery action completes, you can resume normal operation by submitting the job again without the recovery parameters.

For example, you can use the following sample JCL to release all SRDF/AR device locks associated with the process `process-name`:

```plaintext
//RELEASE EXEC PGM=EMCTF,REGION=0M,PARM='RELEASEDEVICELOCK'
//STEPLIB DD DSN=TF/MirrorLibrary,DISP=SHR
//SYSOUT DD SYSOUT=*  
//SYSABEND DD SYSOUT=*  
//SYSIN DD *
GLOBAL WAIT,MAXRC=4
**  
** RELEASE LOCKS FROM YOUR EXISTING SRDF/AR PROCESS.  
**  
MODIFY 03,process-name,START
/*
```

Executing a job step with `PARM='RELEASEDEVICELOCK'` results only in clearing the locks for the devices in the process `process-name` specified in the MODIFY START statement. After the locks are cleared, the SRDF/AR process ends.

Configuring TF/Mirror

TF/Mirror has three configuration layers, each of which is able to override the one above it:

1. TF/Mirror site options

   Site options predefine default values of many TF/Mirror and SRDF/AR commands. You can use default site option values or customize site options according to your requirements. “Setting defaults with site options” on page 24 discusses site options in detail.

2. Settings made with TF/Mirror GLOBAL command parameters

   You can override site option values by using the matching parameters of the GLOBAL command described in “GLOBAL” on page 130.

   To change the value of a site option that has a matching GLOBAL command parameter, specify the GLOBAL parameter you want to use. This value will be used for current batch job step. After processing of that job step is finished, TF/Mirror uses the site option value again.

   **Note:** Not all GLOBAL command parameters have a matching site option. Some of the site options can be protected from overriding.
Getting Started

3. Parameters set on other TF/Mirror operator commands

Many of the TF/Mirror operator commands can take parameters available on the GLOBAL command statement. If you need to override a site option or a GLOBAL parameter, you can set the equivalent parameter with that command.

The value you set with an operator command (other than GLOBAL) is only in force for the duration of the operation of the command. After the command is finished, TF/Mirror uses the corresponding GLOBAL parameter value (if set) or the site option value.

Setting defaults with site options

Site option overview

Site options predefine default values of many TF/Mirror and SRDF/AR commands. Site options provide the ability to fine-tune TimeFinder default behavior for the site, minimizing the amount of settings made with the TimeFinder JCL batch job or operator commands.

The default values set by the site options apply to commands issued using TF/Mirror and SRDF Host Component (the TF command). For example, if GLOBAL CLONE(ALL) is defined as a site option, it becomes a new default value for the CLONE parameter of the GLOBAL command. For a complete list of commands and parameters which default values can be set using site options, see “Default site options” on page 25.

You can use the default site option values or customize site options according to your requirements, as described in “Customizing site options” on page 26. To ensure that a customized value is not overridden by an operator command, lock the customized site option as described in “Locking site options” on page 27.

To get a complete picture of what command parameter values are currently used across all TF/Mirror configuration layers, produce the Site Options for TF/Mirror report described in “Viewing site options and overrides” on page 28.
**Default site options**

Table 1 lists TF/Mirror and SRDF/AR commands and parameters with their corresponding values that are set as default site option values.

**Note:** To change the values for your environment, follow instructions in “Customizing site options” on page 26. To view current settings, see “Viewing site options and overrides” on page 28.

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter and value</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESTABLISH</td>
<td>WAIT(120)</td>
<td>“WAIT[(nnnn)][NOWAIT]” on page 127</td>
</tr>
<tr>
<td>GLOBAL</td>
<td>ATTach(8,1)</td>
<td>“ATTach[(nn,wait_int)]” on page 131</td>
</tr>
<tr>
<td></td>
<td>AUTORELease*:</td>
<td>“AUTORELease((Yes</td>
</tr>
<tr>
<td></td>
<td>No for TF/Mirror, Yes for SRDF/AR</td>
<td></td>
</tr>
<tr>
<td></td>
<td>BYPASSPARSEERROR(No)</td>
<td>“BYPASSPARSEERROR((Yes</td>
</tr>
<tr>
<td></td>
<td>CLONEemulation(RAID5BCV)</td>
<td>“CLONEemulation((RAID5BCV</td>
</tr>
<tr>
<td></td>
<td>CONSistent(MODE(ECA), TIMEOUT(15),SYNConly, ECACLEAR(SEQLEVEL))</td>
<td>“CONSistent([MODE({IOSLEVEL</td>
</tr>
<tr>
<td></td>
<td>CONVertFullEstablish(No)</td>
<td>“CONVertFullEstablish((Yes</td>
</tr>
<tr>
<td></td>
<td>FASTESTablish(No)</td>
<td>“FASTESTablish(Yes</td>
</tr>
<tr>
<td></td>
<td>FBAMETA(COMPLETEonly)</td>
<td>“FBAMETA((ALLOWINCOMPLETE</td>
</tr>
<tr>
<td></td>
<td>MAXGrp(64)</td>
<td>“MAXGrp(nnnn)” on page 135</td>
</tr>
<tr>
<td></td>
<td>MAXRC(0,SETZERO)</td>
<td>“MAXRC(n[, [SETMAX</td>
</tr>
<tr>
<td></td>
<td>MULTBCV(REject)</td>
<td>“MULTBCV((NEW</td>
</tr>
<tr>
<td></td>
<td>MULTIAATTach(ALL)</td>
<td>“MULTIAATTach((NONE</td>
</tr>
<tr>
<td></td>
<td>PROTECTEDRESTORE(REqest)</td>
<td>“PROTECTEDRESTORE((REQUEST</td>
</tr>
<tr>
<td></td>
<td>TerminateExistingRelationship(No)</td>
<td>“TerminateExistingRelationship((Yes</td>
</tr>
<tr>
<td></td>
<td>TolerateDesiredState(NONE)</td>
<td>“TolerateDesiredState((NONE</td>
</tr>
<tr>
<td></td>
<td>VXWAIT(Yes)</td>
<td>“VXWAIT(Yes</td>
</tr>
<tr>
<td></td>
<td>WAIT</td>
<td>“WAIT</td>
</tr>
</tbody>
</table>
Getting Started

Customizing site options

You can customize the default site option values according to your requirements.

**Note:** Only the site administrator should customize site options to avoid unforeseen complications and problems with TF/Mirror processing and expected results.

To customize the default site option values for the installation, edit the JCL contained in the #94TFMJB member of the EMC.SMFEvrm.RIMLIB library and run #94TFMJB to apply the changes.

**Note:** The Mainframe Enablers Installation and Customization Guide discusses EMC.SMFEvrm.RIMLIB.

Include the site option values in the DEFINE step of #94TFMJB:

```
//DEFINE...
...
//SYSIN DD *

site options values
...
*/
```

---

**Table 1 Default site options**

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter and value</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>MODIFY DEFINE</td>
<td>ITRKSYNC(60,2)</td>
<td>“ITRKSYNC([nnn],[mmm])” on page 179</td>
</tr>
<tr>
<td></td>
<td>POLICY(P1,P3,P5,P9,P13)</td>
<td>“POLICY(Pn,[...])” on page 180</td>
</tr>
<tr>
<td></td>
<td>REEST(8,5,1)</td>
<td>“REEST(count,initial_wait,interval)” on page 181</td>
</tr>
<tr>
<td></td>
<td>REEST_TGT(8,5,1)</td>
<td>“REEST_TGT(count,initial_wait,interval)” on page 181</td>
</tr>
<tr>
<td></td>
<td>RESUME(0,30,1)</td>
<td>“RESUME(count,initial_wait,interval)” on page 181</td>
</tr>
<tr>
<td></td>
<td>TIMEOUT(15,TERMinate(IMMEDiate))</td>
<td>“TIMEOUT(time,[CONTinue</td>
</tr>
<tr>
<td>QUERY</td>
<td>EXtended(No)</td>
<td>“EXtended(Yes,[SBCV],[SNAPHold])” on page 141</td>
</tr>
<tr>
<td>RE-ESTABLISH</td>
<td>WAIT(120)</td>
<td>“WAIT([nnnn])NOWAIT” on page 150</td>
</tr>
<tr>
<td>RESTORE</td>
<td>WAIT(120)</td>
<td>“WAIT([nnnn])NOWAIT” on page 158</td>
</tr>
<tr>
<td>SPLIT</td>
<td>BCVRefresh(No)</td>
<td>“BCVRefresh([Yes][No])” on page 162</td>
</tr>
<tr>
<td></td>
<td>INSTant(Yes)</td>
<td>“INSTant([Yes][No])” on page 164</td>
</tr>
<tr>
<td></td>
<td>R2SYNC(Yes)</td>
<td>“R2Sync([Yes][No])” on page 165</td>
</tr>
<tr>
<td></td>
<td>WAIT(120)</td>
<td>“WAIT([nnnn])NOWAIT” on page 166</td>
</tr>
</tbody>
</table>

---

a. With default site option values, TF/Mirror and SRDF/AR have a different default value of this parameter. However, when a value is specified in site options or in GLOBAL, the specified value is used for both TF/Mirror and SRDF/AR.
Specify the site option values in the following format:

```
command_name command_parameters
```

**Note:** Commands and parameters that can be specified as site options are listed in Table 1 on page 25. For complete information about the syntax of each command, see “Command Reference” on page 115.

Figure 1 shows an example of customized site options:

```
GLOBAL ATTACH(7,5)
GLOBAL AUTORELEASE(YES),LOCK
GLOBAL BYPPERR(N),LOCK
GLOBAL CLONE(ALLBCV)
GLOBAL CONS(MODE(ECA),
        TIMEOUT(25),
        SYNONLY,
        ECACLEAR(CONTROLLER)),
        LOCK
GLOBAL CONVFE(N),FASTEST(Y),FBAMETA(COMplete),LOCK
GLOBAL MAXGRP(76),MAXRC(4,SETMAX)
GLOBAL MULTBCV(OLD),MULTIATT(ALL),PROTRESTORE(ALLBCV)
GLOBAL TER(NO),TDS(NONE),WAIT,LOCK
QUERY EX(Y,BCV,SNAP)
ESTABLISH WAIT(200)
RE-ESTABLISH WAIT(160)
RESTORE WAIT(140)
SPLIT BCVR(NO),INS(YES),R2SYNC(YES),WAIT(25),LOCK
MODIFY DEFINE,
        ITRKSNC(70,4),
        POLICY(P1,P2,P3,P5,P6,P9,P10,P11,P13),
        REEST(8,5,1),
        REEST_TGT(12,6,2),
        RESUME(0,40,2),
        LOCK
MODIFY DEFINE,
        TIMEOUT(25,CONTINUE)
```

**Figure 1 Customized site options**

**Locking site options**

Locking protects customized site options from unauthorized override by a user issuing a TF/Mirror operator command or the SRDF Host Component TF command.

The following commands support locking: GLOBAL, SPLIT, MODIFY DEFINE.

To lock a customized site option, specify the LOCK parameter with the site option statement, for example:

```
SPLIT BCVR(NO), INS(YES), R2SYNC(YES), WAIT(25), LOCK
```

LOCK applies to all command parameters listed in the site option statement. If you need only a part of command parameters to be locked, add separate statements containing locked and non-locked command parameters; for example:

```
SPLIT BCVR(NO)
SPLIT INS(YES), R2SYNC(YES), LOCK
SPLIT WAIT(25)
```
Getting Started

When a user attempts to override a locked site option but has insufficient rights, TF/Mirror acts depending on the current MAXRC value:

- If MAXRC is less than 8, TF/Mirror issues error message BCVI166E and stop command processing. RC 8 can be generated based on the SETMAX/SETZERO setting.

- If MAXRC equals to 8, TF/Mirror proceeds with command processing; however, the locked parameters are not overridden. Values specified as customized site options in #94TFMJB are used. RC 4 can be generated based on the SETMAX/SETZERO setting.

The following resources are used to implement site options security:

- XFACILIT class: EMC.ADMIN.CMD.TF.SITE-OPTIONS-OVERRIDE
- TF#BASE class: SITE-OPTIONS-OVERRIDE

Note: The Mainframe Enablers Installation and Customization Guide provides detailed information about the EMCSAFI security interface.

Viewing site options and overrides

To view site option values together with any overrides, produce the Site Options for TF/Mirror report as follows:

- Issue GLOBAL DEBUG(SITEOptions) to produce the Basic format report described in “Basic format” on page 28.

- Issue GLOBAL DEBUG(SITEOptions(ALL)) to produce the Extended format report described in “Extended format” on page 30.

If the TFSITEO DD statement is specified in the TF/Mirror batch job, TF/Mirror produces the Extended format report and redirects it to the TFSITEO DD.

Basic format

The Basic format report displays current site option values (default or customized) and their GLOBAL overrides.

The output is similar to Figure 2.

Note: This example displays site options defined as shown in Figure 1 on page 27, with the MAXRC value overridden with the following operator command:

GLOBAL MAXRC(8,SETMAX)
### Site Options for TF/Mirror report—Basic format

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameter</th>
<th>Site options</th>
<th>Current value</th>
</tr>
</thead>
<tbody>
<tr>
<td>MODIFY DEFINE</td>
<td># ITRKSYNC</td>
<td># BCVRefresh</td>
<td>No</td>
</tr>
<tr>
<td>QUERY</td>
<td># INSTANT</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>RE-ESTABLISH</td>
<td># WAIT</td>
<td>*140</td>
<td></td>
</tr>
<tr>
<td>SPLIT</td>
<td># BCVRefresh</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>ESTABLISHzzzzzzzzzzz</td>
<td>WAIT</td>
<td>*200</td>
<td></td>
</tr>
<tr>
<td>QUERY</td>
<td>EXtended</td>
<td>(Y, SBCV, SNAPH)</td>
<td></td>
</tr>
<tr>
<td>RE-ESTABLISHzzzzzzzzzzz</td>
<td>WAIT</td>
<td>*160zzzzzzzzzzzz</td>
<td></td>
</tr>
<tr>
<td>SPLIT</td>
<td># BCVRefresh</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>TERMINATE</td>
<td>CONTInue</td>
<td>CONTInue</td>
<td></td>
</tr>
</tbody>
</table>

---

**Report fields**

- **Command**: The TF/Mirror or SRDF/AR command used to define the site option.
- **Parameter**: A TF/Mirror or SRDF/AR command parameter used to define the site option.
- **Site options**: The TF/Mirror or SRDF/AR command parameter value set as the site option (default or customized).
- **Current value**: The actual value applied on the operator command.

Throughout the report, an asterisk (*) means that the specified value takes effect when the corresponding command parameter is issued with no sub-parameters. The hash (#) indicates that the site option is locked and override is allowed for authorized users only.
**Getting Started**

### Extended format

The Extended format report appends listing of overriding operator commands to the basic format. It contains additional entries for every ESTABLISH, RE-ESTABLISH, RESTORE, SPLIT or QUERY command found in user input. The exact command referred to by the Site Options for TF/Mirror report can be found using the statement number shown right next to the command name.

The output is similar to Figure 3.

**Note:** This example displays site options defined as shown in Figure 1 on page 27 and operator commands issued as follows:

(0001) GLOBAL MAXRC(8,SETMAX)
(0002) QUERY 01,51D4,1,EX(N)
(0003) SPLIT 02,51D4,NOWAIT

---

<table>
<thead>
<tr>
<th>BCVI168I</th>
<th>Command</th>
<th>Parameter</th>
<th>Site options</th>
<th>Current value</th>
</tr>
</thead>
<tbody>
<tr>
<td>BCVI170I</td>
<td>GLOBAL</td>
<td>*7(5)</td>
<td>*(7,5)</td>
<td>*(7,5)</td>
</tr>
<tr>
<td>BCVI170I</td>
<td>AUTORElease (TF/M,SRDF/AR)</td>
<td>(Yes,Yes)</td>
<td>(Yes,Yes)</td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td># BYPASSPARSEERROR</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td>CLONEemulation</td>
<td>ALL</td>
<td>ALL</td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td># CONSistent</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td># MODE</td>
<td>ECA</td>
<td>ECA</td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td>TIMEOUT</td>
<td>25</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td># SYNC</td>
<td>SYNConly</td>
<td>SYNConly</td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td># ECACLR</td>
<td>CONTROLLER</td>
<td>CONTROLLER</td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td># CONVERTFullEstablish</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td># FASTESTablish</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td># FBAMETA</td>
<td>COMPLETEonly</td>
<td>COMPLETEonly</td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td>MAXGRP</td>
<td>76</td>
<td>76</td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td>MAXRC</td>
<td>4,SETMAX</td>
<td>8,SETMAX</td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td>MULTBCV</td>
<td>OLDest</td>
<td>OLDest</td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td>MULTIATTach</td>
<td>ALL</td>
<td>ALL</td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td>PROTECTEDRESTORE</td>
<td>ALL</td>
<td>ALL</td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td># TerminateExistingRelationship</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td>TolerateDesiredState</td>
<td>NONE</td>
<td>NONE</td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td>VXWAIT</td>
<td>Yes</td>
<td>YES</td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td>WAIT</td>
<td>WAIT</td>
<td>WAIT</td>
<td></td>
</tr>
<tr>
<td>BCVI168I</td>
<td>QUERY</td>
<td>EXTended</td>
<td>(Y, SBCV, SNAPH)</td>
<td></td>
</tr>
<tr>
<td>BCVI168I</td>
<td>ESTABLISH</td>
<td>WAIT</td>
<td>*200</td>
<td></td>
</tr>
<tr>
<td>BCVI168I</td>
<td>RE-ESTABLISH</td>
<td>WAIT</td>
<td>*160</td>
<td></td>
</tr>
<tr>
<td>BCVI168I</td>
<td>RESTORE</td>
<td>WAIT</td>
<td>*140</td>
<td></td>
</tr>
<tr>
<td>BCVI168I</td>
<td>SPLIT</td>
<td># BCVRefresh</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td># INStant</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td># R2SYNC</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td># WAIT</td>
<td></td>
<td>*25</td>
<td></td>
</tr>
<tr>
<td>BCVI168I</td>
<td>MODIFY DEFINE</td>
<td># ITRKSYNC</td>
<td>70,4</td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td># POLICY</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td>P1</td>
<td>On</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td>P2</td>
<td>On</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td>P3</td>
<td>On</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td>P4</td>
<td>Off</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td>P5</td>
<td>On</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td>P6</td>
<td>On</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td>P7</td>
<td>Off</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td>P8</td>
<td>Off</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td>P9</td>
<td>On</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td>P10</td>
<td>On</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td>P11</td>
<td>On</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td>P12</td>
<td>Off</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td>P13</td>
<td>On</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td>P14</td>
<td>Off</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td>P15</td>
<td>Off</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td>P16</td>
<td>Off</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td># REEST</td>
<td>8,5,1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td># REEST_TGT</td>
<td>12,6,2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td># RESUME</td>
<td>0,40,2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td>TIMEOUT</td>
<td>25</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td>TERMINATE</td>
<td>CONTINUE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCVI168I</td>
<td>(0002) QUERY</td>
<td>EXtended</td>
<td>(Y,SBCV,SNAPH)</td>
<td>(N)</td>
</tr>
<tr>
<td>BCVI168I</td>
<td>(0003) SPLIT</td>
<td># BCVRefresh</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>BCVI170I</td>
<td># INSTant</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td># R2SYNC</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>BCVI170I</td>
<td># WAIT</td>
<td>*25</td>
<td>NOWAIT</td>
<td></td>
</tr>
</tbody>
</table>

**Figure 3** Site Options for TF/Mirror report—Extended format

For field explanation, see “Report fields” on page 29.
CHAPTER 3
Replicating Data

This chapter covers the following topics:

- Replication environment ................................................................. 34
- Replication process ........................................................................ 35
- Replication technology .................................................................. 52
- TF/Mirror with SRDF ................................................................. 58
- TF/Mirror operations .................................................................... 64
- TF/Mirror queries ......................................................................... 71
Replication environment

TF/Mirror replicates data between production devices called standard devices or STDs and backup devices called business continuance volumes or BCVs. STDs and BCVs have unique addresses that make them accessible by the host.

When one or more BCVs are assigned to an STD, the devices form an STD-to-BCV relationship or an STD/BCV pair. An STD/BCV pair includes one STD and one or more BCVs.

Basic configuration

Basic TF/Mirror configuration is illustrated in Figure 4.

![Figure 4 Basic TF/Mirror configuration](image)

SRDF configurations

See “TF/Mirror with SRDF” on page 58.
Replication process

Figure 5 provides an overview of TF/Mirror process flow.

The TF/Mirror process includes the following steps:

- Provisioning BCVs
- Setting global parameters
- Synchronizing from STD to BCV: ESTABLISH
- Stopping synchronization: SPLIT
- Re-synchronizing from STD to BCV: RE-ESTABLISH
- Synchronizing from BCV to STD: RESTORE
- Deleting the STD/BCV pair

Note: Under PowerMaxOS 5978, HYPERMAX OS 5977 and Enginuity 5876, all TF/Mirror operations are performed in the clone emulation mode. This means that an internal API automatically converts all TF/Mirror commands to corresponding TF/Clone Mainframe Snap Facility commands. For more information, see “Clone emulation” on page 52.
Provisioning BCVs

Configure devices on the storage system as BCVs.

PowerMaxOS 5978/HYPERMAX OS 5977/Enginuity 5876

A BCV is a clone copy\(^1\). The devices to be used as BCVs must be configured with the BCV attribute.

The BCV does not require PowerMax/VMAX mirror positions. However, the BCV can have any configuration except VDEV (virtual device); that is, the BCV can have any form of RAID protection.

**Note:** The *TimeFinder/Clone Mainframe Snap Facility Product Guide* provides information about clone copies and VDEVs.

Enginuity 5773

A BCV is a mirror. Your Dell EMC Customer Support Engineer configures storage system devices as BCVs during system installation or servicing.

A BCV can be configured as:

- A single mirror
- A locally mirrored device
- An SRDF source (R1) device

**Note:** The *SRDF Host Component for z/OS Product Guide* describes R1 devices.

- A RAID-protected device (including RAID5- and RAID6-protected devices)

The BCV can function either as:

- An additional mirror to a storage system logical volume
- An independent, host-addressable volume

**Mirror structure**

The STD mirrors each contain copies of the data stored in the STD. There can be up to four STD mirrors. The STD can have any local mirror structure (normal, RAID1, RAID S, RAID5, RAID6, RAID10).

A BCV mirror is an STD mirror that is assigned upon creation of the STD/BCV pair. In case of Concurrent BCVs, two mirror positions are required.

**Note:** “Concurrent BCV” on page 40 describes Concurrent BCVs.

Three dynamic mirrors are allowed. These can be any combination of BCVs, Dynamic SRDF, and Dynamic Spares. However, there is still a limit of two attached BCVs for each STD, regardless of dynamic mirror support.

---

1. See “Clone emulation” on page 52 for information about clone emulation and clone copies in TF/Mirror.
Figure 6 provides an example of the initial storage system configuration. The host accesses the M1 and M2 mirrors as a single device (Vol A). The host accesses the BCV as Vol B.

![Initial configuration under Enginuity 5773](image)

**Setting global parameters**

Global parameters influence behavior of TF/Mirror and provide default values for execution of TF/Mirror commands. A global default value can be overridden by a parameter of a particular TF/Mirror command.

You can set global parameters using the GLOBAL command, as described in “GLOBAL” on page 130.

**Synchronizing from STD to BCV: ESTABLISH**

You start replication from STDs to BCVs using the ESTABLISH command, as described in “ESTABLISH” on page 123.

The ESTABLISH command pairs STDs and BCVs specified as the command parameters, copies the contents of each STD to its paired BCV, and then keeps synchronizing all changes made on the STD to the BCV. Any new data written to the STD is copied to the BCV. During synchronization, the BCV is inaccessible by the host.

**Note:** Under PowerMaxOS 5978, HYPERMAX OS 5977 or Enginuity 5876, the TF/Mirror ESTABLISH command is mapped to the SNAP VOLUME PRECOPY(YES) DIFFERENTIAL(YES) command of TF/Clone Mainframe Snap Facility, as explained in “Clone emulation” on page 52. See the TimeFinder/Clone Mainframe Snap Facility Product Guide for information about the SNAP VOLUME command.
Replicating Data

ESTABLISH steps

The following actions are performed when the storage system receives the ESTABLISH command from the host:

- Validity checks, such as:
  - If STD and BCV are of the same size
    
    Note: CKD STD volumes can only be established to CKD BCV volumes of the same size or larger. FBA STD volumes can only be established to FBA volumes of the same size.
  - If the device specified as the BCV has the BCV attribute
  - If the STD has a BCV assigned to it
- Making the BCV Not Ready and offline to the host
- If clone emulation is not used: Assigning the BCV as the next available (second, third, or forth) mirror of the STD, as illustrated in Figure 7:

![Figure 7 Assigning the BCV as a mirror of the STD](image-url)
Replicating Data

- Copying the data from the STD to one or more BCVs

  If clone emulation is not used, the BCV receives the data from both M1 and M2, as shown in Figure 8. Any data existing on the BCV is overwritten.

- Synchronizing any changes made to the data on the STD to the BCV

Multi-BCV

The Multi-BCV feature allows an STD to have more than one paired BCV.

- In the clone emulation mode:

  All BCVs paired with the STD can be active at the same time; that is, data can be copied from the STD to all BCVs simultaneously.

  The maximum number of paired BCVs is 6.

- When clone emulation is not used:

  An STD can have up to 8 paired BCVs. Only one STD/BCV pair can be active at a time. For each of other BCVs, an SDDF session is opened to log changes in data on the STD. Prior to ESTABLISHing another STD/BCV pair, you must stop synchronization for the current pair using the SPLIT command.

  You can determine the action the system takes when the maximum number of multiple BCVs for an STD is exceeded using the MultBCV parameter of the GLOBAL command, as described in “GLOBAL” on page 130.
Concurrent BCV

**IMPORTANT**
The Concurrent BCV feature is not needed in the clone emulation mode.

The Concurrent BCV feature enables you to synchronize the STD with two independent BCVs, as shown in Figure 9:

![Concurrent BCV Diagram]

**Figure 9** Concurrent BCV

The STD can have any local mirror structure. For Concurrent BCVs, two available mirror positions are required.

With Enginuity 5773, there can be three dynamic mirrors for each storage system device (STD or BCV). A dynamic mirror can be:

- A BCV
- A dynamic SRDF mirror
- A dynamic spare mirror
- A dynamic relocation volume (DRV)

However, you can have at most two BCVs attached to an STD at the same time (when the Multi-BCV feature and clone emulation are not used).

Other restrictions include:

- RESTOREs cannot be issued if the standard device has an active BCV paired.

**Note:** “Synchronizing from BCV to STD: RESTORE” on page 48 describes the RESTORE action.
A second Concurrent BCV pair cannot be ESTABLISHed or RESTOREd if the first
STD/BCV pair was a Protected RESTORE.

Note: “Protected RESTORE” on page 49 describes Protected RESTOREs.

Protected BCV ESTABLISH is mutually exclusive with Concurrent BCV because all
the available mirror positions are used.

Note: “Protected BCV” on page 41 describes the Protected BCV ESTABLISH
action.

A Concurrent BCV is not allowed if the STD is a Dynamic SRDF device.

Note: The SRDF Host Component for z/OS Product Guide describes dynamic
SRDF devices.

To enable the Concurrent BCV feature, use the CBCV parameter of the ESTABLISH
command, as described in “ESTABLISH” on page 123.

Protected BCV

IMPORTANT
The Protected BCV feature applies to locally mirrored BCVs when no clone emulation
is used. In the clone emulation mode, all ESTABLISH operations are executed as
though the Protected BCV feature is enabled.

The Protected BCV feature adds both mirrors of the BCV as mirrors of the STD during
synchronization from the STD to the BCV. If there are no available mirror positions, this
operation fails. When synchronization to the BCV is stopped, both BCV mirrors are
synchronized with the STD from the beginning. This ensures that data on the BCV is
protected at any point of time.

To enable the Protected BCV feature, use the PROTECTEDBCVEstablish parameter of
the ESTABLISH command, as described in “ESTABLISH” on page 123.

Post-ESTABLISH operations

Normally, all devices being ESTABLISHed must be SPLIT. “Stopping synchronization:
SPLIT” on page 42 provides information about the SPLIT action.
Stopping synchronization: SPLIT

You stop replication using the SPLIT command, as described in “SPLIT” on page 161.

The SPLIT command applies both to STD-to-BCV and BCV-to-STD replication. When you stop replication, the STD and BCVs contain an exact copy of data, which is valid up to the point in time when the SPLIT command is issued.

Note: Under PowerMaxOS 5978, HYPERMAX OS 5977 and Enginuity 5876, the TF/Mirror SPLIT command is mapped to the ACTIVATE command of TF/Clone Mainframe Snap Facility, as explained in “Clone emulation” on page 52. See the TimeFinder/Clone Mainframe Snap Facility Product Guide for information about the ACTIVATE command.

After the SPLIT operation completes, each of the devices can be operated independently (including Concurrent BCVs). The host accesses the devices using their original device addresses. You can now use the BCV for business continuance purposes, such as backup.

Note: You can use the TimeFinder Utility after the SPLIT to process files on the BCV. The TimeFinder Utility for z/OS Product Guide describes the TimeFinder Utility.

SPLIT steps

IMPORTANT

Under PowerMaxOS 5978, HYPERMAX OS 5977 and Enginuity 5876, SPLIT uses clone emulation for all functions. SPLIT captures a point-in-time image of the source volume and makes the BCV Ready to the host as an instant point-in-time copy.

The following actions are performed when the storage system receives the SPLIT command from the host:

◆ Validity checks, such as:
  ■ If the STD has an active BCV mirror
  ■ If the devices specified as STD and BCVs comprise an STD/BCV pair
  ■ If the BCV is fully synchronized with the STD
  ■ For MSC: if the MSC session is globally consistent

◆ SRDF environment checks for Consistent SPLITs:
  ■ If a mixture of SRDF/A and non-SRDF/A devices exist
  ■ If multiple SRDF/A groups are in one MSC (Multi-Session Consistency) session

Note: “TF/Mirror with SRDF” on page 58 describes SRDF configurations for TF/Mirror. The SRDF Host Component for z/OS Product Guide provides information about SRDF/A and MSC. “Consistent SPLIT” on page 43 describes Consistent SPLITs.

◆ Suspending I/O to the STD until the SPLIT operation completes
◆ Destaging any pending writes to the STD and BCVs
Replicating Data

- Separating the BCV from the STD/BCV pair
- Opening an SDDF (Symmetrix Differential Data Facility) session to track changes on the STD
  
  The SDDF session begins logging changes in data on the STD after the STD and BCVs are established. Second and subsequent SPLITs benefit from the SDDF session because it enables only changed tracks to be copied.
- Making the BCV Ready for the host using its original address
- Resuming STD operations while flagging any new writes to the STD. Logging of changes is necessary for updating the BCV if it is RE-ESTABLISHED with the same STD at a later time.

**Note:** "Re-synchronizing from STD to BCV: RE-ESTABLISH" on page 47 describes the RE-ESTABLISH action.

**Instant SPLIT**

In an Instant SPLIT, the SPLIT operation is performed in the background after completion of the SPLIT I/O request.

**IMPORTANT**

Under currently supported levels of the operating environment, all splits are Instant SPLITs.

**Consistent SPLIT**

A Consistent SPLIT is a SPLIT operation that ensures consistency between the data copies on the STD and BCVs.

For a Consistent SPLIT, the STD must be online and available to the host.

To perform a Consistent SPLIT, use either of the options:

- Suspend all applications that are using the STD and ensure that all host buffering and intermediate caching is flushed to the appropriate logical device in the storage system before performing the SPLIT operation.
- Use the CONSistent parameter of the SPLIT command with the LOCAL (for a single-system SPLIT) or GLOBAL (for a multisystem SPLIT) parameter, as described in “SPLIT” on page 161.

**Consistent SPLIT with zBoost PAV Optimizer**

When processing Consistent SPLIT under PowerMaxOS 5978, HYPERMAX OS 5977 or Enginuity 5876, TF/Mirror suspends zBoost™ PAV Optimizer write processing before setting ECA window. It resumes the write processing before clearing the ECA window.

**Note:** The ResourcePak Base for z/OS Product Guide describes zBoost PAV Optimizer.
Remote Consistent SPLIT

You can perform a Consistent SPLIT of BCVs attached to any SRDF device in the SRDF/A or SRDF/S environment.

**Note:** “TF/Mirror with SRDF” on page 58 describes SRDF configurations for TF/Mirror.

The BCV can reside in a locally attached storage system or in a storage system up to 3 hops away.

In a multihop environment, you do not have to initiate a Remote Consistent SPLIT from the host attached to the production storage system. Instead, you can initiate a Consistent SPLIT from a recovery site. TF/Mirror dynamically validates the location of the R1 device.

- For SRDF/S:
  - If the R2 STD is active on the SRDF link, TF/Mirror automatically locates the storage system that contains the associated R1 (up to 3 hops away) and sets ECA₁ accordingly in that storage system before performing the SPLIT operation.
  - If the R2 STD is not active on the SRDF link, TF/Mirror sets ECA in the storage system containing the SRDF/S R2 device before performing the SPLIT operation.

For example, assume Site A is where the SRDF/S R1s are located and Site B is where the SRDF/S R2s and attached BCVs are located. A host at Site C running TF/Mirror can perform a Remote Consistent SPLIT of the BCVs at Site B (2 hops away). TF/Mirror locates the R1s at Site A (3 hops away) and sets ECA there before performing the SPLIT operation at Site B.

- For SRDF/A:
  - If the R2 STD is active on the SRDF link, TF/Mirror suspends the SRDF/A delta set switch process in the storage system where the R2 resides before performing the SPLIT operation. ECA is disabled for SRDF/A.

- For mixed SRDF/A and non-SRDF/A:
  - Consistency cannot be guaranteed. TF/Mirror performs a non-consistent SPLIT and issues the BCVI122W message (or BCVI122E, depending on the MAXRC parameter value of the GLOBAL command).

- For MSC:
  - TF/Mirror checks if the MSC session is globally consistent. When global consistency cannot be guaranteed, TF/Mirror issues the BCVM169W (or BCVI119E, depending on the MAXRC parameter value of the GLOBAL command described in “GLOBAL” on page 130).

---

1. For information about ECA (Enginuity Consistency Assist), see the Consistency Groups for z/OS Product Guide.
If the R2 STDs of the STD/BCV pair are in multiple SRDF/A groups, MSC for these groups is required. If these SRDF/A groups are not in one active MSC group, consistency cannot be guaranteed. TF/Mirror issues the BCVI150W message and performs a non-consistent SPLIT if MAXRC is >= 4 or issues the BCVI150E message and stops processing when MAXRC is less than 4 (the MAXRC value is a parameter of the GLOBAL command described in “GLOBAL” on page 130).

ECA I/O Control Mode

If the SRDF environment consists of mixed ECA and non-ECA protected volumes, TF/Mirror uses non-ECA protection. If all the volumes are using ECA, TF/Mirror uses ECA protection.

\[\text{Note: For information about ECA, see the Consistency Groups for z/OS Product Guide.}\]

You can set TF/Mirror to clear ECA for a storage system when polling has completed for all BCVs in that system. To do this, use the CONS(ECACLEAR(CONTROLLER)) parameter of the GLOBAL command, as described in “GLOBAL” on page 130.

Steps in Remote Consistent SPLIT

To perform a Consistent SPLIT on the remote storage system:

1. The storage system containing the R2 STDs locks the device from further updates.
2. If the R1/R2 device pair is not already suspended, the remotely mirrored pair is suspended for a few seconds to execute the SPLIT.
3. The storage system containing the R2 STDs executes the SPLIT command.
4. If the remotely mirrored pair was not previously suspended, the link is resumed. Any changed tracks from the R1 or R11 (because of updates while the pair was suspended) are propagated to the R2 STDs for synchronization purposes.
5. The BCV containing a copy of the R2 STD data is made available to its host.

Note the following when doing a point-in-time Consistent SPLIT on two different R2 devices off the legs of an R11 device:

- PowerMaxOS 5978, HYPERMAX OS 5977 or Enginuity 5876 is required.
- Each PowerMax/VMAX device must be at the same sequence level.
- The SPLIT occurs in both storage systems where R2s reside while SRDF/A cycle switching is suspended in both systems and a common split time is achieved.
- A Consistent SPLIT on only one leg and not the other can be performed.
- A Consistent SPLIT can be performed on each leg separately at different sequence levels. In this situation, a common split time is not achieved.

Reverse SPLIT

\[\text{IMPORTANT}\]

\[\text{The Reverse SPLIT feature cannot be used in the clone emulation mode.}\]
In a Reverse SPLIT, you reverse the synchronization direction for the device mirrors. The BCV (M1) is synchronized from the BCV mirror (M2), whereas the normal direction is from M1 to M2.

The prerequisites for a Reverse SPLIT are as follows:

- The BCV must be mirrored locally. To synchronize from a remote mirror, see “Remote Reverse SPLIT” on page 46.
- The BCV mirror (M2) must have been fully synchronized with the BCV before ESTABLISHing the BCV.

If the BCV does not have a local mirror or the mirror is not synchronized, you receive the BCVM148W message (or BCVM148E, depending on the MAXRC parameter value of the GLOBAL command).

To perform a Reverse SPLIT, use the BCVRefresh(Y) parameter of the SPLIT command, as described in “SPLIT” on page 161.

If you use the ChangedOnly(Y) parameter of the SPLIT command for a Reverse SPLIT, the BCV is updated from its mirror and only changed tracks on the BCV are replaced with the mirrored tracks (M2 tracks). This feature can make resynchronization complete faster.

Remote Reverse SPLIT

In a Remote Reverse SPLIT, you split R1 BCV from its STD for the purpose of restoring data from R2 to R1 BCV using SRDF recovery procedure 4 or 6.

**Note:** The SRDF Host Component for z/OS Product Guide describes the SRDF recovery procedures.

To perform a Remote Reverse SPLIT, use the R2SYNC(N) parameter of the SPLIT command, as described in “SPLIT” on page 161.

Post-SPLIT operations

While the devices are split, changes can be introduced to the data on the STD and BCVs. This means that the data stored on the devices is not identical anymore.

After you finish the business continuance activities on the BCVs, you can restart replication in either of the directions:

- From the STD to the BCV, thus discarding any changes made to the data on the BCV. You choose this option if the data yielded from running applications on the BCV is not needed, or if a fresh copy of current data is needed. See “Re-synchronizing from STD to BCV: RE-ESTABLISH” on page 47 for details.

**Note:** You can also create a new STD/BCV pair, consisting of the same BCV but with a different STD.
- From the BCV to the same or different STD. In this case, the copy stored on the BCV survives. See “Synchronizing from BCV to STD: RESTORE” on page 48 for details.

**Note:** The operations you perform on the BCVs may alter the change bits and the last reference date in the VTOC. This information is not reflected back to the STD unless you execute a RESTORE command.

**Re-synchronizing from STD to BCV: RE-ESTABLISH**

You can restart replication from the STD to the BCV using the RE-ESTABLISH command described in “RE-ESTABLISH” on page 147.

The RE-ESTABLISH functionality and steps are the same as for the ESTABLISH operation, as described in “Synchronizing from STD to BCV: ESTABLISH” on page 37, with the following exception.

Instead of copying all data, the system copies to the BCV only new data written to the STD since the last SPLIT of the STD/BCV pair. This is called *differential or incremental* synchronization. Any new data written to the BCV while the STD/BCV pair was SPLIT are overwritten by the data on the corresponding track on the STD.

**Note:** Under PowerMaxOS 5978, HYPERMAX OS 5977 and Enginuity 5876, the TF/Mirror RE-ESTABLISH command is mapped to the SNAP VOLUME DIFFERENTIAL(YES) command of TF/Clone Mainframe Snap Facility, as explained in “Clone emulation” on page 52. See the TimeFinder/Clone Mainframe Snap Facility Product Guide for more information about the SNAP VOLUME command.

The RE-ESTABLISH process may take time to complete if there was a large amount of updates to the STD and/or BCV.
Synchronizing from BCV to STD: RESTORE

If you want to keep the data residing on the BCV, you can copy the entire contents of the BCV to the STD. You can use the original STD specified in the STD/BCV pair (previously ESTABLISHed and SPLIT), or a different STD.

**Note:** Under PowerMaxOS 5978, HYPERMAX OS 5977 and Enginuity 5876, the TF/Mirror SPLIT command is mapped to the SNAP VOLUME DIFFERENTIAL(YES) command of TF/Clone Mainframe Snap Facility, as explained in “Clone emulation” on page 52. See the TimeFinder/Clone Mainframe Snap Facility Product Guide for more information about the SNAP VOLUME DIFFERENTIAL(YES) command.

You can perform additional RESTOREs from the protected image on the BCV. This is helpful in situations when the STD may need to be returned to the point of the previous SPLIT operation.

RESTORE steps

The following actions are performed when the storage system receives the RESTORE command from the host:

- Validity checks, such as if the STD and BCV are of the same size
- Making the BCV *Not Ready* to the host
- If clone emulation is not used: Assigning the BCV as the next available (second, third, or forth) mirror of the STD
- Copying the data from the BCV to the STD and all its mirrors (if clone emulation is not used), as shown in Figure 10. Existing data on the STD is overwritten.

![Figure 10 Restoring an STD/BCV pair](image)

- Synchronizing any changes made to the data on the BCV to the STD
### Full RESTORE

In a full RESTORE, the entire contents of the BCV is copied into the specified STD. The STD and BCV must be offline to all hosts to which they are attached.

In the clone emulation mode, if the devices in a pair do not have an existing relationship, a new session must be created between them. In this case, an ESTABLISH is performed instead of a RESTORE.

To perform a full RESTORE, use the “Full RESTORE” syntax listed in “RESTORE” on page 155.

### Incremental RESTORE

In an Incremental RESTORE, the storage system reassigns the BCV to the STD to which it was assigned before the SPLIT and copies from the BCV to the STD only changed data (tracks). Only data that was updated on the BCV while the STD/BCV pair was SPLIT is copied. Any changed tracks on the STD are overwritten by the data on the corresponding track on the BCV.

The BCV must be offline to all hosts to which it is attached.

If you perform an Incremental RESTORE with the STD online, you may need to bring the device offline and then online again, depending on the data that was updated.

To perform an Incremental RESTORE, omit the cuup parameter of the RESTORE command, as described in “RESTORE” on page 155.

### Protected RESTORE

**IMPORTANT**

In the clone emulation mode, the Protected RESTORE feature is enabled by default. All RESTOREs are Protected RESTOREs.

A Protected RESTORE is one in which the BCV is protected against host writes to the paired STD until the next SPLIT command is issued.

When clone emulation is not used, you use the PROTECTEDRESTORE parameter of the RESTORE command, as described in “RESTORE” on page 155, to perform a Protected RESTORE. In addition, you have to set the PROTECTEDRESTORE parameter of the next SPLIT command, as described in “SPLIT” on page 161.

### SPLIT after Protected RESTORE

When clone emulation is not used, the SPLIT command used next to a Protected RESTORE must be executed in a Protected RESTORE mode. To do this:

- Set the PROTECTEDRESTORE parameter of the SPLIT command to Yes, or
- Set the PROTECTEDRESTORE parameter of the GLOBAL command to ALLBCV or CLONE.

See “SPLIT” on page 161 and “GLOBAL” on page 130 for details on the commands and parameters.
Unprotected RESTORE

IMPORTANT

Unprotected RESTOREs can be performed under Enginuity 5773 when the clone emulation mode is not used and the Protected RESTORE feature is disabled.

An Unprotected RESTORE can be destructive to the BCV because any new writes made to the STD are also made to the BCV.

If you want to keep the original point-in-time copy after an Unprotected RESTORE, perform the following steps to preserve the BCV:

- Ensure that the STD is offline to all hosts to which it is attached.
- Perform a RESTORE to the STD.
- With the STD remaining offline, split the BCV using the SPLIT command.
- Bring the STD online only after the SPLIT is accomplished.

RESTORE confirmation

The RESTORE process may take time to complete if there was a large amount of updates to the STD and/or BCV. A WTOR is issued to the system console for operator confirmation.

You can use the following resources at the SAF Security Interface to bypass the WTOR:

- EMC.ADMIN.CMD.TF.FULL-RESTORE-BYPASS-WTOR for a full RESTORE
- EMC.ADMIN.CMD.TF.PARTIAL-RESTORE-BYPASS-WTOR for an incremental RESTORE


Restrictions

- In the clone emulation mode, TF/Mirror accepts a RESTORE only if all tracks had been copied from the STD before the RESTORE operation was initiated.

- In an SRDF environment, when using the clone emulation mode, all blocks that previously prevented concurrent RESTOREs and R2 to R1 synchronization are removed. You can perform R2 to R1 synchronization if a RESTORE operation to the R2 is in progress.

  Note: “TF/Mirror with SRDF” on page 58 describes SRDF configurations for TF/Mirror.

- If you issue multiple Incremental RESTORE commands to multiple storage systems simultaneously with the gatekeepers being the STDs for the RESTORE operations, the requests fail. To avoid the problem, change the RESTORE commands to specify a BCV as the gatekeeper.

- TF/Mirror denies RESTORE commands to active R2 STD devices.
Under Enginuity 5773, you cannot use SRDF recovery procedures 4 and 6 during clone emulation RESTOREs.

**Note:** The *SRDF Host Component for z/OS Product Guide* describes SRDF recovery procedures.

The information necessary for a RESTORE is lost (necessitating a full copy from the BCV mirror) if a spare is invoked against an STD and then removed while the STD is split from the BCV.

### Deleting the STD/BCV pair

When all replication operations are complete, you can delete the relationship between the STD and the BCV using the DELINC or MULTIDELINC parameter of the CONFIG command, as described in “CONFIG” on page 117.

**IMPORTANT**

Do not use the CONFIG DELINC or CONFIG MULTIDELINC command if the synchronization has not been completed because all the source tracks may not have been copied to the BCV.
Replicating Data

Replication technology

The replication technology used by TF/Mirror depends on the OS level:

- Under PowerMaxOS 5978, HYPERMAX OS 5977 and Enginuity 5876, an internal API automatically converts all TF/Mirror commands to corresponding TF/Clone Mainframe Snap Facility commands. This is called clone emulation. All TF/Mirror operations are automatically performed in the clone emulation mode.

For information about the clone emulation mode, see “Clone emulation” on page 52.

For information about TF/Clone Mainframe Snap Facility, see the TimeFinder/Clone Mainframe Snap Facility Product Guide.

- Under Enginuity 5773, clone emulation or native TF/Mirror processing can be used.

For RAID5 or RAID6 replication targets, clone emulation is used. For other devices, clone emulation is controlled by the CLONEemulation parameter of the GLOBAL command, as described in “GLOBAL” on page 130.

Under native TF/Mirror processing, the replication target is a specially tagged volume introduced when the storage system is configured. It functions as a mirror of the source device.

Clone emulation

With clone emulation, an internal API automatically converts all TF/Mirror commands to corresponding TF/Clone Mainframe Snap Facility commands.

Table 2 describes the command mapping.

<table>
<thead>
<tr>
<th>TF/Mirror command</th>
<th>TF/Clone Mainframe Snap Facility command</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESTABLISH</td>
<td>SNAP VOLUME PRECOPY(YES) DIFFERENTIAL(YES)</td>
</tr>
<tr>
<td>RE-ESTABLISH</td>
<td>SNAP VOLUME DIFFERENTIAL(YES)</td>
</tr>
<tr>
<td>RESTORE</td>
<td>SNAP VOLUME DIFFERENTIAL(YES)</td>
</tr>
<tr>
<td>SPLIT</td>
<td>ACTIVATE</td>
</tr>
</tbody>
</table>

In the clone emulation mode:

- No mirror positions are required or used.
- You can have up to eight (8) BCVs with one STD.
- All ESTABLISH operations are processed using the Protected BCV feature.

Note: “Protected BCV” on page 41 describes the Protected BCV feature.

- All RESTORE operations are Protected RESTOREs.

Note: “Protected RESTORE” on page 49 describes Protected RESTOREs.
A RESTORE can be accepted only if all tracks were copied from the source device prior to initiating the RESTORE operation.

The Concurrent BCV feature is not used.

Note: “Concurrent BCV” on page 40 describes the Concurrent BCV feature.

The Reverse SPLIT mode is not used.

Note: “Reverse SPLIT” on page 45 describes the Reverse SPLIT mode.

When multiple devices are specified together in ESTABLISH, SPLIT, and RESTORE commands, or in the CONIFG command with the DELINC or MULTIDELINC parameter, TF/Mirror uses multidevice syscalls to process the requests together.

The HOLD attribute (signifying that a device is Not Ready for operations) does not appear on the BCV report for BCVs in a clone emulation session.

Because a HOLD is implicitly set on the BCV as a result of a clone emulation ESTABLISH, that HOLD needs to be ignored for subsequent TF/Mirror operations. Otherwise, a RE-ESTABLISH or another ESTABLISH would be denied.

Cascaded clone emulation

In the clone emulation mode, TF/Mirror supports cascaded clone emulations.

For details on cascaded clone emulation, see the TimeFinder/Clone Mainframe Snap Facility Product Guide.

You cannot use clone emulation to cascade clone emulation. However, you can use clone emulation to cascade from a regular clone session.

This limitation only exists for cascaded clone emulations, where one clone session passes data on to the next session connected to it. Concurrent BCV and Multi-BCV use a one-to-many relationship, where one STD is the source for two or more BCVs.

Note: “Concurrent BCV” on page 40 describes Concurrent BCVs. “Multi-BCV” on page 39 describes the Multi-BCV feature.

Example

Consider the following example of using TF/Clone Mainframe Snap Facility to clone from an original device to an STD and then using clone emulation to copy from the STD to a BCV:

1. Establish an STD/BCV pair in the clone emulation mode using the ESTABLISH command, as described in “ESTABLISH” on page 123.

2. You need to update the STD with any data on the original device that has changed during a specified time period. To do that, use TF/Clone Mainframe Snap Facility to perform a differential SNAP VOLUME from the original device to the STD.

Note: The TimeFinder/Clone Mainframe Snap Facility Product Guide provides information about the SNAP VOLUME command.

3. Ensure that all background copying operations have completed.
Replicating Data

4. Split the STD/BCV pair using the SPLIT command, as described in “SPLIT” on page 161.

5. Ensure that background copying has completed and that the changes are on the BCV.

6. Delete the STD/BCV pair using the CONFIG command with the DELINC parameter, as described in “CONFIG” on page 117.

Supported device types

Diskless SRDF devices

Under PowerMaxOS 5978, HYPERMAX OS 5977 and Enginuity 5876, TF/Mirror recognizes diskless SRDF devices. However, TF/Mirror does not perform operations against diskless SRDF devices. An attempt to issue a command against a diskless SRDF device causes an error.

Thin devices

TF/Mirror supports thin devices under PowerMaxOS 5978, HYPERMAX OS 5977 or Enginuity 5876. A thin device can be an STD or BCV.

EAVs

With PowerMaxOS 5978, HYPERMAX OS 5977 or Enginuity 5876, TF/Mirror can perform operations against Extended Address Volumes (EAVs).

GCM support

Geometry Compatible Mode (GCM) allows SRDF relationships to be established between an FBA device on a storage system running Enginuity 5876 and an FBA device on a storage system running PowerMaxOS 5978 or HYPERMAX OS 5977, where the device under PowerMaxOS 5978/HYPERMAX OS 5977 is exactly a half cylinder larger than the device under Enginuity 5876.

Note: For more information about GCM, see the SRDF Host Component for z/OS Product Guide.

FBA devices operating in the GCM mode (on a storage system running PowerMaxOS 5978 or HYPERMAX OS 5977) are indicated using the FBATG emulation type:

<table>
<thead>
<tr>
<th>Device Type</th>
<th>Action</th>
<th>Last Action</th>
<th>Status</th>
<th>Used</th>
<th>BCV</th>
<th>Emulation</th>
<th>CYLS</th>
<th>Type</th>
<th>Sync</th>
<th>Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>BCVM003I</td>
<td>QUERY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>FBATG</td>
<td>3070</td>
<td>NONE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCVM003I</td>
<td>QUERY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>FBATG</td>
<td>3070</td>
<td>NONE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCVM003I</td>
<td>QUERY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>FBATG</td>
<td>3070</td>
<td>NONE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCVM003I</td>
<td>QUERY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>FBATG</td>
<td>3070</td>
<td>NONE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCVM003I</td>
<td>QUERY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>FBATG</td>
<td>3070</td>
<td>NONE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCVM003I</td>
<td>QUERY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>FBATG</td>
<td>3070</td>
<td>NONE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCVM003I</td>
<td>QUERY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>FBATG</td>
<td>3070</td>
<td>NONE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCVM003I</td>
<td>QUERY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>FBATG</td>
<td>3070</td>
<td>NONE</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

TimeFinder/Mirror for z/OS 8.0 and Higher Product Guide
TF/Mirror treats devices with the GCM attribute as follows:

- If the source device of an ESTABLISH/FULL RESTORE operation has the GCM attribute on, the GCM attribute will also be set on the target device as a result of the operation.
- If the target device of the ESTABLISH/FULL RESTORE operation is already in a LCL/RMT relationship, the GCM attribute cannot be set on the target device and the appropriate error message is issued.
- TF/Mirror does not clear the GCM attribute neither on source nor on target device after their relationship has been terminated.

**Note:** See the *SRDF Host Component for z/OS Product Guide* for instruction on clearing the GCM attribute.

### Online/offline status check

During some operations, devices are unavailable to the hosts to which they are attached.

TF/Mirror checks the online/offline status by verifying path groups of STDs and BCVs on the hosts, as described in Table 3.

<table>
<thead>
<tr>
<th>TF/Mirror command</th>
<th>Device type</th>
<th>BCV</th>
<th>STD</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESTABLISH</td>
<td>All hosts</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>RE-ESTABLISH</td>
<td>All hosts</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Incremental RESTORE</td>
<td>All hosts</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>RESTORE</td>
<td>All hosts</td>
<td>All hosts</td>
<td></td>
</tr>
</tbody>
</table>

The online/offline status check can be bypassed using the SAF Security Interface EMC.ADMIN.CMD.TF.BYPASS-ONLINE-CHECK resource. The *Mainframe Enablers Installation and Customization Guide* describes the SAF Security Interface.

### Device state change

During operations that change the state of a device, the host issues an ENQ for the device to serialize requests. The host uses the resource QNAME of TF/Mirror and the RNAME is `serial-number.dv#`.

### Device locks

TF/Mirror uses Device External Locks (DEL) to serialize operations to a PowerMax/VMAX device. TF/Mirror obtains a lock on the BCV during any active operation, such as ESTABLISH, RE-ESTABLISH, SPLIT, and RESTORE.
During the course of processing, TF/Mirror monitors the duration of any held lock.

- If a device lock has been in effect for less than 15 minutes, any new request for a lock on that device fails.
- If a device lock has been in effect for 15 minutes or more, then TF/Mirror releases the existing lock and the new request for a lock on that device is honored.

You can use the ReleaseDeviceLock parameter of the TF/Mirror batch job to release device locks manually, when needed. “Running TF/Mirror” on page 21 provides information about using the ReleaseDeviceLock parameter.

**Multi-attach**

The Multi-Attach feature allows multiple device pairs to be attached on the same ESTABLISH or RE-ESTABLISH syscall.

Commands at the same sequence level are eligible for multi-attach. For example, the following two ESTABLISH commands result in a multi-attach operation:

```plaintext
ESTABLISH 10,6EFE,6E7E,MULTIATT(Y)
ESTABLISH 10,6EFF,6E7F,MULTIATT(Y)
```

However, changing the sequence level (“10”) in one of these commands results in individual ESTABLISH syscalls.

To enable the Multi-Attach feature, use the MULTIATTach parameter of the GLOBAL or ESTABLISH (or RE-ESTABLISH) command, as described in “GLOBAL” on page 130 and “ESTABLISH” on page 123.

**Multitasking**

TF/Mirror supports multitasking when processing commands.

To enable multitasking and specify the number of subtasks, use the PARALLEL parameter of the GLOBAL command, as described in “GLOBAL” on page 130.

**Assigning requests to subtasks**

In the multitasking mode, the requests within each sequence level are spread among subtasks as follows:

- A single subtask is used for each of the following:
  - All SPLITs for the same storage system
  - All multi-attach requests for the same storage system
  - All CONFIG READY requests for consecutive devices in the same storage system
  - All CONFIG NR requests for consecutive devices in the same storage system
  - All RMT commands to the same storage system through the same gatekeeper
- All remaining requests are sent to separate subtasks unless the device and the storage system are the same as another request. In that case, the remaining requests are sent to the same subtask.
If the maximum number of subtasks is reached, the remaining requests are held and assigned to subtasks as they finish the already assigned work.

Order of command output

Command output within a sequence level is no longer presented in the order in which the commands were input. The output order is determined by the order in which the commands complete execution.

For example, if you have the following commands:

(0002) SPLIT 5,1234
(0003) CONFIG 5,1235,HOLD
(0004) RESTORE 5,1236

The output might appear in order 0002, 0003, 0004 or 0002, 0004, 0003 or 0004, 0003, 0002 or any other permutation.

To preserve execution order, take either of the following steps:

◆ Disable multitasking by setting the PARALLEL parameter of the GLOBAL command to No.
◆ Use distinct sequence levels.

Abnormal ending

In case of a severe error, when TF/Mirror ends without processing all commands, commands at the current sequence level that have already begun execution in another subtask continue until they end. However, new work is not dispatched after such a severe error.
TF/Mirror with SRDF

In an SRDF environment, TF/Mirror can be used in the following configurations:

- R1 is STD
- R1 is BCV
- R2 is STD
- R11 is STD
- R21 is STD
- R22 is STD

In the SRDF environment, the gatekeeper device should not be actively involved in the TF/Mirror operation. For example, it should not be one of the STDs in a Consistent SPLIT.

An R11, R21, R22 device cannot be a BCV. The operating environment does not support BCVs with more than one SRDF mirror.

Any ESTABLISH, SPLIT, RESTORE, or REESTABLISH action involving SRDF/Metro devices (source or target of operation) will be blocked. SRDF/AR will check for SRDF/Metro devices in SRDF/AR process and will not allow the process to start if any such devices found.

**Note:** The PowerMax Family Product Guide, VMAX All Flash Product Guide and VMAX3 Family Product Guide describe SRDF and SRDF/Metro. SRDF/Metro is supported for FBA devices only.

**R1 is STD**

*Figure 11* illustrates a BCV paired with the R1 device.

![Figure 11: R1 is STD](image)

**SPLIT**

Splitting an R1 STD/BCV pair does not affect data transfer with the R2.
RESTORE

When performing a RESTORE to an R1 STD:

- If the SRDF link is active, the RESTORE is made locally to the R1 STD and remotely to the R2.
- If the SRDF link is not active (suspended), the RESTORE is only made locally to the R1. The information about changed tracks is retained for later synchronization with the R2.

R1 is BCV

Figure 12 illustrates an R1 BCV.

Figure 12  R1 is BCV

When the R1 BCV is paired with its STD, the R1 BCV operates as a normal BCV.

When an R1 BCV is split from the STD, the R1 BCV is paired and synchronized with the R2 according to the specified SRDF mode. The WaitSync and R2Sync parameters of the SPLIT command can be used, as described in “SPLIT” on page 161.
**R2 is STD**

Figure 13 illustrates an R2 STD.

RESTORE

TF/Mirror denies RESTORE commands to active R2 STD devices.

When performing a RESTORE to an R2 STD:

- Under PowerMaxOS 5978, HYPERMAX OS 5977 and Enginuity 5876, you can perform SRDF recovery procedures 4 or 6 for R2 to R1 synchronization while a clone emulation RESTORE to the R2 is in progress. (SRDF Host Component forces TF/Clone protected tracks to become SRDF remote invalid tracks).

  After the RESTORE is complete, execute the appropriate SRDF recovery procedure (4 or 6) to copy the changed tracks from R2 to R1.

  **Note:** The *SRDF Host Component for z/OS Product Guide* describes SRDF recovery procedures.

- Under Enginuity 5773, you must suspend SRDF to allow the RESTORE before performing R2 to R1 synchronization with SRDF recovery procedures 4 and 6.

Consistent SPLIT

In case of a Consistent SPLIT, consistency with R1 is achieved in addition to consistency with R2. See “Remote Consistent SPLIT” on page 44.
R11 is STD

TF/Mirror can be used with R11 devices in Concurrent SRDF configurations, as shown in Figure 14.

Figure 14  R11 is STD

For information about the specifics of TF/Mirror operations, see “R1 is STD” on page 58.
R21 is STD

TF/Mirror can be used with R21 devices in Cascaded SRDF configurations, as shown in Figure 15.

For information about the specifics of TF/Mirror operations:

- See “R1 is STD” on page 58 for the R21 and R2 pair.
- See “R2 is STD” on page 60 for the R1 and R21 pair.

Note: If the R21 is diskless, all TF/Mirror operations are prevented.

Consistent SPLIT

When the SRDF link between R21 and R2 is in the adaptive copy mode:

- If you are using the R21 as an STD in a Consistent SPLIT, you have to specify the ALLOWNONSYNC parameter in the GLOBAL command, as described in “GLOBAL” on page 130.
R22 is STD

Under PowerMaxOS 5978, HYPERMAX OS 5977 and Enginuity 5876, you can use TF/Mirror with R22 (concurrent R2) devices, as shown in Figure 16.

**Figure 16  R22 is STD**

TF/Mirror treats R22 devices like R2 devices on an active SRDF link. See “R2 is STD” on page 60.
TF/Mirror operations

Basic operations

Table 4 lists basic TF/Mirror operations.

Note: “Replication process” on page 35 describes basic TF/Mirror operations.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Control</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create a BCV group and fill it with BCVs</td>
<td>CONFIG commands</td>
</tr>
<tr>
<td>Copy data from STDs to BCVs</td>
<td>• ESTABLISH command</td>
</tr>
<tr>
<td></td>
<td>• RE-ESTABLISH command</td>
</tr>
<tr>
<td>Copy data from BCVs to STDs</td>
<td>RESTORE command</td>
</tr>
<tr>
<td>Stop replication between STDs and BCVs</td>
<td>SPLIT command</td>
</tr>
<tr>
<td>Delete STD/BCV pair</td>
<td>CONFIG command, DELINC or MULTIDELINC action</td>
</tr>
</tbody>
</table>

Viewing BCVs

Table 5 lists operations for viewing BCVs.

Note: “TF/Mirror queries” on page 71 describes TF/Mirror queries.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Control</th>
</tr>
</thead>
<tbody>
<tr>
<td>View BCVs</td>
<td>QUERY command</td>
</tr>
<tr>
<td>View extended report on BCVs</td>
<td>QUERY command, EXtended parameter or corresponding site option</td>
</tr>
<tr>
<td>Disable type check when viewing BCVs</td>
<td>GLOBAL command, NOTYPE parameter</td>
</tr>
<tr>
<td>Disable CUU-BCV search when viewing BCVs</td>
<td>GLOBAL command, NOCUU parameter</td>
</tr>
</tbody>
</table>

Managing device status

Table 6 lists operations for managing device status.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Control</th>
</tr>
</thead>
<tbody>
<tr>
<td>Make the BCV Ready</td>
<td>CONFIG command, READY action</td>
</tr>
<tr>
<td>Make the BCV Not Ready</td>
<td>CONFIG command, NR action</td>
</tr>
<tr>
<td>Make the BCV Not Ready after a SPLIT</td>
<td>SPLIT command, NR parameter</td>
</tr>
<tr>
<td>Ignore device status (unavailable for Consistent SPLITs)</td>
<td>GLOBAL command, TolerateDesiredState parameter or corresponding site option</td>
</tr>
</tbody>
</table>
Managing device holds

Table 7 lists operations for managing device holds.

Note: “Device locks” on page 55 provides information about device locks.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Control</th>
</tr>
</thead>
<tbody>
<tr>
<td>Place a device hold</td>
<td>CONFIG command, HOLD action</td>
</tr>
<tr>
<td>Place a hold on the BCV after a SPLIT</td>
<td>SPLIT command, HOLD parameter</td>
</tr>
<tr>
<td>Remove a device hold</td>
<td>CONFIG command, RELEASE action</td>
</tr>
<tr>
<td>Release STD holds automatically</td>
<td>GLOBAL command, AUTORElease parameter or corresponding site option</td>
</tr>
</tbody>
</table>

Setting up clone emulation

IMPORTANT
These operations are not available under PowerMaxOS 5978, HYPERMAX OS 5977 and Enginuity 5876.

Table 8 lists operations for setting up the clone emulation mode.

Note: “Clone emulation” on page 52 describes the clone emulation mode.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Control</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable/disable clone emulation for all commands</td>
<td>GLOBAL command, CLONEemulation parameter or corresponding site option</td>
</tr>
<tr>
<td>Enable/disable clone emulation for an ESTABLISH command</td>
<td>ESTABLISH command, CLONEemulation parameter</td>
</tr>
<tr>
<td>Enable/disable clone emulation for a RE-ESTABLISH command</td>
<td>RE-ESTABLISH command, CLONEemulation parameter</td>
</tr>
<tr>
<td>Enable/disable clone emulation for a RESTORE command</td>
<td>RESTORE command, CLONEemulation parameter</td>
</tr>
<tr>
<td>Enable/disable clone emulation for a SPLIT command</td>
<td>SPLIT command, CLONEemulation parameter</td>
</tr>
</tbody>
</table>
Setting up Multi-Attach

Table 9 lists operations for setting up the Multi-Attach feature.

**Note:** “Multi-attach” on page 56 describes the Multi-Attach feature.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Control</th>
</tr>
</thead>
<tbody>
<tr>
<td>Determine operations for which the Multi-Attach feature is enabled</td>
<td>GLOBAL command, MULTIATTach parameter or corresponding site option</td>
</tr>
<tr>
<td>Enable/disable the Multi-Attach feature for all ESTABLISH or RESTORE commands</td>
<td>GLOBAL command, ATTach parameter or corresponding site option</td>
</tr>
<tr>
<td>Enable/disable the Multi-Attach feature for an ESTABLISH command</td>
<td>ESTABLISH command, MULTIATTach parameter</td>
</tr>
<tr>
<td>Enable/disable the Multi-Attach feature for a RE-ESTABLISH command</td>
<td>RE-ESTABLISH command, MULTIATTach parameter</td>
</tr>
</tbody>
</table>

Setting up Concurrent BCVs

**IMPORTANT**

These operations are not available in the clone emulation mode.

Table 10 lists operations for managing Concurrent BCVs.

**Note:** “Concurrent BCV” on page 40 describes Concurrent BCVs.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Control</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use a Concurrent BCV in an ESTABLISH command</td>
<td>ESTABLISH command, CBCV parameter</td>
</tr>
<tr>
<td>Use a Concurrent BCV in a RE-ESTABLISH command</td>
<td>RE-ESTABLISH command, CBCV parameter</td>
</tr>
</tbody>
</table>
Setting up Protected BCVs

IMPORTANT
These operations are not available in the clone emulation mode.

Table 1 lists operations for managing Protected BCVs.

Note: “Protected BCV” on page 41 describes Protected BCVs.

Table 11 Setting up Protected BCVs

<table>
<thead>
<tr>
<th>Operation</th>
<th>Control</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable/disable the Protected BCV feature for an ESTABLISH command</td>
<td>ESTABLISH command, PROTECTEDBCVEstablish parameter</td>
</tr>
<tr>
<td>Enable/disable the Protected BCV feature for a RE-ESTABLISH command</td>
<td>RE-ESTABLISH command, PROTECTEDBCVEstablish parameter</td>
</tr>
<tr>
<td>Enable/disable the Protected RESTORE feature for a SPLIT command</td>
<td>SPLIT command, PROTECTEDRESTORE parameter</td>
</tr>
<tr>
<td>Enable/disable the Protected RESTORE feature for a RESTORE command</td>
<td>RESTORE command, PROTECTEDRESTORE parameter</td>
</tr>
</tbody>
</table>

Setting up a SPLIT

Table 12 lists operations for setting up a SPLIT operation.

Note: “Stopping synchronization: SPLIT” on page 42 describes the SPLIT operation.

Table 12 Setting up a SPLIT

<table>
<thead>
<tr>
<th>Operation</th>
<th>Control</th>
</tr>
</thead>
<tbody>
<tr>
<td>Perform an Instant SPLIT</td>
<td>SPLIT command, INStant parameter or corresponding site option</td>
</tr>
<tr>
<td>Enable/disable the Reverse SPLIT mode (unavailable in the clone emulation mode)</td>
<td>SPLIT command, BCVRefresh parameter or corresponding site option</td>
</tr>
<tr>
<td>Bypass online/offline status check for STDs on other LPARs for a SPLIT command</td>
<td>SPLIT command, BYPassonlinecheck parameter</td>
</tr>
<tr>
<td>Synchronize only changed tracks on the BCV mirrors for a SPLIT command</td>
<td>SPLIT command, ChangedOnly parameter</td>
</tr>
<tr>
<td>Force a SPLIT before the ESTABLISH or RESTORE operation completes</td>
<td>SPLIT command, FORCE parameter</td>
</tr>
</tbody>
</table>
Setting up Consistent SPLITs

Table 13 lists operations for Consistent SPLITs.

Note: “Consistent SPLIT” on page 43 describes Consistent SPLITs.

Table 13 Setting up Consistent SPLITs

<table>
<thead>
<tr>
<th>Operation</th>
<th>Control</th>
</tr>
</thead>
<tbody>
<tr>
<td>Perform a Consistent SPLIT</td>
<td>SPLIT command, CONSistent parameter</td>
</tr>
<tr>
<td>Perform a multisystem Consistent SPLIT</td>
<td>SPLIT command, GLOBAL parameter</td>
</tr>
<tr>
<td>Perform a single-system Consistent SPLIT</td>
<td>SPLIT command, LOCAL parameter</td>
</tr>
<tr>
<td>Set the I/O control mode (ECA/IOSLEVEL) for</td>
<td>GLOBAL command, CONSistent parameter</td>
</tr>
<tr>
<td>Consistent SPLITs</td>
<td></td>
</tr>
<tr>
<td>Set the I/O control mode to ECA for a SPLIT</td>
<td>SPLIT command, ECA parameter</td>
</tr>
<tr>
<td>command</td>
<td></td>
</tr>
<tr>
<td>Set the I/O control mode to IOSLEVEL for a</td>
<td>SPLIT command, IOSLEVEL parameter</td>
</tr>
<tr>
<td>SPLIT command</td>
<td></td>
</tr>
</tbody>
</table>

Synchronizing R1/R2

Table 14 lists operations for automatic R1/R2 synchronization.

Note: “TF/Mirror with SRDF” on page 58 describes SRDF configurations for TF/Mirror.

Table 14 Synchronizing R1/R2

<table>
<thead>
<tr>
<th>Operation</th>
<th>Control</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable/disable automatic R2 to R1 synchronization for a</td>
<td>RESTORE command, R1SYNC parameter</td>
</tr>
<tr>
<td>RESTORE command</td>
<td></td>
</tr>
<tr>
<td>Enable/disable automatic R1 to R2 synchronization for a</td>
<td>SPLIT command, R2SYNC parameter or corresponding</td>
</tr>
<tr>
<td>SPLIT command</td>
<td>site option</td>
</tr>
</tbody>
</table>

Updating volser information

Table 15 lists operations for updating volser information.

Table 15 Updating volser information

<table>
<thead>
<tr>
<th>Operation</th>
<th>Control</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable/disable volser verification for a RESTORE command</td>
<td>RESTORE command, VERIFY parameter</td>
</tr>
<tr>
<td>Change the BCV volser after a SPLIT</td>
<td>SPLIT command, VOLID parameter</td>
</tr>
<tr>
<td>Change BCV VVDS, index VTOC, and DSCB after a SPLIT</td>
<td>SPLIT command, VOLID(E) parameter</td>
</tr>
</tbody>
</table>
Setting up TF/Mirror

Table 16 lists operations for setting up TF/Mirror.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Control</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set the maximum allowed return code</td>
<td>GLOBAL command, MAXRC parameter or corresponding site option</td>
</tr>
<tr>
<td>Enable/disable multitasking</td>
<td>GLOBAL command, PARallel parameter</td>
</tr>
<tr>
<td>Enable/disable debugging</td>
<td>GLOBAL command, DEBUG parameter</td>
</tr>
</tbody>
</table>

Waiting for synchronization completion and timeouts

Table 17 lists operations to set up waiting for synchronization completion and timeouts.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Control</th>
</tr>
</thead>
<tbody>
<tr>
<td>Determine whether to wait for completion of synchronization for all commands</td>
<td>GLOBAL command, VXWAIT and WAIT</td>
</tr>
<tr>
<td>Determine whether to wait for completion of synchronization for the ESTABLISH command</td>
<td>ESTABLISH command, VXWAIT and WAIT</td>
</tr>
<tr>
<td>Determine whether to wait for completion of synchronization for the RE-ESTABLISH command</td>
<td>RE-ESTABLISH command, VXWAIT and WAIT</td>
</tr>
<tr>
<td>Determine whether to wait for completion of synchronization for the RESTORE command</td>
<td>RESTORE command, VXWAIT and WAIT</td>
</tr>
<tr>
<td>Determine whether to wait for completion of synchronization for the SPLIT command</td>
<td>SPLIT command, VXWAIT and WAIT</td>
</tr>
<tr>
<td>Determine whether to wait for the BCV mirrors (M2, M3, or M4) to synchronize (M1-M4)</td>
<td>SPLIT command, WaitSync parameter</td>
</tr>
<tr>
<td>Set the timeout for Consistent SPLITs</td>
<td>GLOBAL command, CONS(TIMEOUT) parameter or corresponding site option</td>
</tr>
<tr>
<td>Set the timeout for a SPLIT command</td>
<td>SPLIT command, CONS(TIMEOUT) parameter</td>
</tr>
</tbody>
</table>
Table 18 lists miscellaneous parameters that define global behavior of TF/Mirror.

Table 18: Miscellaneous parameters

<table>
<thead>
<tr>
<th>Operation</th>
<th>Control</th>
</tr>
</thead>
<tbody>
<tr>
<td>Allow a partial list of FBA meta group members in commands</td>
<td><strong>GLOBAL</strong> command, FBAMETA parameter or corresponding site option</td>
</tr>
<tr>
<td>Determine how exceeding STD-to-BCV relationships are treated</td>
<td><strong>GLOBAL</strong> command, MULTBCV parameter or corresponding site option</td>
</tr>
<tr>
<td>Convert a failing incremental RE-ESTABLISH to a full ESTABLISH (unavailable in the clone emulation mode)</td>
<td><strong>GLOBAL</strong> command, CONVertFullEstablish parameter or corresponding site option</td>
</tr>
<tr>
<td>Convert a full ESTABLISH to an incremental RE-ESTABLISH</td>
<td><strong>GLOBAL</strong> command, FASTESTablish parameter or corresponding site option</td>
</tr>
<tr>
<td>Allow or prohibit termination of an existing STD-to-BCV relationship for ESTABLISH and RESTORE operations</td>
<td><strong>GLOBAL</strong> command, TerminateExistingRelationship parameter or corresponding site option</td>
</tr>
</tbody>
</table>
TF/Mirror queries

TF/Mirror queries provide information about BCVs and their status.

To make a query, issue the QUERY command, as described in “QUERY” on page 141. Storage systems respond to a valid QUERY command by returning device records for each BCV defined.

You can perform a basic or an extended query. You choose the query type using the EXtended(Y) parameter of the QUERY command.

Depending on the query type, two different messages are used to produce a query report: BCVM003I for a basic query and BCVM073I for an extended query. For information about fields in these messages, see the Mainframe Enablers Message Guide.

Note the following when using TF/Mirror queries:

◆ If you place a CONFIG HOLD on a device, a QUERY report no longer shows the STD with which the BCV is paired.

◆ You can use the NOCUU parameter of the GLOBAL command to disable CUU-BCV search when a QUERY command is executed, as described in “GLOBAL” on page 130.

◆ There are some differences in running queries that include both regular BCVs and BCVs that are part of a clone emulation session and queries that only cover the same clone emulation BCVs:
  ■ In queries covering both regular and clone emulation BCVs, the clone emulation BCVs show HOLD relationship.
  ■ In queries covering only the clone emulation BCVs, the clone emulation BCVs show as Snap targets. This is because the HOLD attribute is suppressed for BCVs with a clone emulation session to provide compatibility for clone emulation.

◆ By default, an extended query is sorted by the STD and the time since the split. The display begins with the first BCV that has had an STD/BCV relationship. However, if you use QUERY Extended(Y, SBCV), then the display begins with the BCV whose PowerMax/VMAX device number you specified in the QUERY symdv# parameter.
Replicating Data
This chapter covers the following topics:

- Introduction to SRDF/AR ................................................................. 74
- SRDF/AR environment ................................................................. 75
- SRDF/AR processes and cycles .................................................. 80
- SRDF/AR operations ........................................................................ 88
- SRDF/AR examples ................................................................. 91
- Recovery and troubleshooting .................................................... 95
Introduction to SRDF/AR

SRDF/AR (Symmetrix Remote Data Facility/Automated Replication) automates data copying across SRDF links to provide a logically consistent, restartable image of data at a remote (recovery) site in the event of a disaster at the production site.

SRDF/AR automatically propagates the restartable image of data to the recovery site in a manner transparent to the host application or database. The result is a series of consecutive data consistency points that you can use as the basis for restarting host applications at the recovery site.

You can use SRDF/AR in single-hop and multihop configurations.

**Note:** SRDF/A is incompatible with SRDF/AR.

Consistent copies

The data copy created by SRDF/AR is a consistent copy.

The term consistent is based on the concept of dependent write I/O. A dependent write is a write that is not issued by the application unless some prior write operation has successfully completed.

Most applications, and in particular database management systems (DBMS), have dependent write logic embedded in them to ensure data integrity if a failure occurs in the host processor, software, or storage system. An example is a database update.

When updating a database, a DBMS takes the following steps:

- Writes to the disk containing the log
- Writes the data to the actual database dataset
- Writes again to the log volume to indicate that the database update was made

These three write I/Os (log, database, and log again) are related and each I/O is not issued until the prior I/O has successfully completed.
SRDF/AR environment

Single-hop configuration

A single-hop SRDF/AR environment is illustrated in Figure 17.

![Figure 17 Single-hop SRDF/AR configuration](image)

**Note:** The SRDF/AR configuration is based on the “R1 is BCV” configuration shown in Figure 12 on page 59.
Multihop configuration

In a multihop configuration, SRDF/AR is configured, controlled, and monitored by the production site but runs between the two recovery sites. A multihop SRDF/AR environment is illustrated in Figure 18.

Figure 18 Multihop SRDF/AR configuration

Consistency Group

Dell EMC Consistency Group (ConGroup) protects the data between sites A and B in the event of an unplanned disaster (rolling disaster).

Note: The Consistency Groups for z/OS Product Guide provides information about Dell EMC Consistency Group.

ConGroup trips and a dependent write consistent copy (restartable image) of the data as it was at the beginning point of the rolling disaster is captured on the R2 STD_b devices at Site B. One final SRDF/AR cycle propagates the restartable image of the data to Site C.

Note: “SRDF/AR processes and cycles” on page 80 describes SRDF/AR cycles.

In the event of a ConGroup trip event, Site B contains a consistent, restartable image of data of the production site. A final split of the R1 BCVs followed by their resynchronization to the R2 STD_c at Site C produces a restartable environment at long distances with data current to the point at which the disaster happened.

Note: “Stopping synchronization: SPLIT” on page 42 describes the SPLIT operation.
SRDF/AR protection

SRDF/AR automates sending the dependent write consistent copy of data from Site B to Site C. No channel connectivity is required to Site B. A restartable copy, with some data loss, still exists if the production Site A and recovery Site B are both lost. SRDF/AR performs a remote Consistent SPLIT in order to create the restartable images by holding the I/O at Site A while issuing the remote Instant SPLIT to Site B and releasing the I/O at Site A.

**Note:** “Stopping synchronization: SPLIT” on page 42 describes the SPLIT operation.

Device addressing

In a multihop configuration, the only host-addressable devices are the R1 devices at the production Site A. SRDF/AR automatically identifies R2 STD devices during execution. For all other devices, PowerMax/VMAX device numbers must be used. SRDF/AR automatically identifies the SRDF groups used at both recovery sites.

ADCOPY mode

Adaptive copy is not allowed in a multihop environment.

Initial configuration requirements

Initially, the original data is stored on the STDs at the production Site A, and a consistent copy of data exists on the R1 BCVs (synchronized using TF/Mirror) and R2 STDs (synchronized using SRDF).

The initial single-hop SRDF/AR configuration is shown in Figure 19.

![Figure 19](image-url) Initial single-hop SRDF/AR configuration

The requirements to the initial SRDF/AR environment are as follows:

- All R1 BCVs must be ESTABLISHEDed.

**Note:** For more information about the ESTABLISH operation, see “Synchronizing from STD to BCV: ESTABLISH” on page 37.
The R1 BCVs cannot be HELD.

The R1 BCVs must be offline.

There must be no invalid tracks.

All R2 STDs must be SPLIT from their BCVs.

**Note:** The SRDF/AR configuration does not use BCVs paired to R2 STDs, if the BYPTBCV parameter of the SRDF/AR MODIFY DEFINE or MODIFY START command is set to Yes, as described in “MODIFY DEFINE” on page 175.

The BCVs paired to R2 STDs must be offline.

There must be no invalid tracks.

The SRDF link is active (not suspended).

**Note:** If the SRDF/AR environment includes devices that may operate in Geometry Compatible Mode (GCM), this may require special processing. See “GCM support” on page 54 and SRDF Host Component for z/OS Product Guide for details.

## Creating an SRDF/AR environment

**IMPORTANT**

Before start using the SRDF/AR environment for the first time, run a Consistent SPLIT of the R1 BCVs followed by a RE-ESTABLISH. See “SPLIT” on page 161 and “RE-ESTABLISH” on page 147 for a description of the corresponding commands.

To create an SRDF/AR environment:

1. Create an SRDF link between the R1 and R2 devices using the #SC VOL CREATEPAIR command of SRDF Host Component.

   **Note:** See the SRDF Host Component for z/OS Product Guide for information about the #SC VOL CREATEPAIR command.

2. Optionally, to create a multihop environment:

   Create an SRDF link between the Site A R1 and Site B R2 STD devices using the #SC VOL CREATEPAIR command of SRDF Host Component.

3. Establish the STD/R1 BCV pairs at the production site using the ESTABLISH command of TF/Mirror.

   **Note:** “Synchronizing from STD to BCV: ESTABLISH” on page 37 describes the ESTABLISH operation. See “ESTABLISH” on page 123 for information about the ESTABLISH command.

4. Establish the R2 STD/BCV pairs at the recovery site using the ESTABLISH command of TF/Mirror.
5. Perform a SPLIT of the R2 STD/BCV pairs at the recovery site using the SPLIT command of TF/Mirror.

**Note:** See “SPLIT” on page 161 for information about the SPLIT command.

6. Add a new process using the ADD command of SRDF/AR.

**Note:** See “ADD” on page 173 for information about the ADD command.

7. Configure the new process using the MODIFY DEFINE command of SRDF/AR.

**Note:** See “MODIFY DEFINE” on page 175 for information about the ADD command.

**SRDF/AR storage estimates**

Table 19 shows the defaults for the SRDF/AR control blocks:

<table>
<thead>
<tr>
<th>Control block</th>
<th>Default</th>
<th>Overriding parameter</th>
<th>Length @ defaults</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>TFACT</td>
<td>1</td>
<td>N/A</td>
<td>64</td>
<td>CSA</td>
</tr>
<tr>
<td>TFAGRP</td>
<td>64</td>
<td>MAXGrp</td>
<td>1024+ (MAXMsg * 112)</td>
<td>ECSA</td>
</tr>
<tr>
<td>Message buffers</td>
<td>256</td>
<td>MAXMsg</td>
<td>256+ (MAXDev * 128)</td>
<td>ECSA</td>
</tr>
<tr>
<td>TFASYM</td>
<td>Unlimited</td>
<td>N/A</td>
<td>256+ (MAXDev * 128)</td>
<td>ECSA</td>
</tr>
<tr>
<td>Device entries</td>
<td>4096</td>
<td>MAXDev</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

a. The parameter used to override the default values. For more information, see “ADD” on page 173.
b. If storage becomes constrained, most of the SRDF/AR storage requirements can be eliminated by using the MSGOPT(WTO) option and MAXMsg(1).
Automating Replication with SRDF/AR

SRDF/AR processes and cycles

SRDF/AR automates the process of replicating data from the production site to the recovery site. You set up and manage processes using SRDF/AR commands.

The SRDF/AR process life cycle is illustrated in Figure 20.

Figure 20 SRDF/AR process life cycle

A process can include one or more cycles. A cycle is a repeated sequence of actions called steps. You can stop a cycle at the end of a step and restart it later from that point.

An SRDF/AR cycle can include the following steps:

- Step 0: Health check
- Step 1A: Ensure R1 BCVs are offline
Automating Replication with SRDF/AR

- Step 1B: Ensure BCVs paired to R2 STDs are offline
- Step 1C: Check R1-R2 invalid tracks and status
- Step 1D: Check R2 STD local mirrors
- Step 1E: Check R1 BCV status
- Step 2: Consistent SPLIT of R1 BCVs
- Step 3: Wait for SRDF sync of R1s with R2s
- Step 4: Check if R1 BCVs are Ready
- Step 5: RE-ESTABLISH R1 BCVs
- Step 6: Wait for mirror sync on BCVs paired to R2 STDs
- Step 7: RE-ESTABLISH BCVs paired to R2 STDs
- Step 8: Wait for TF/Mirror sync of R2 STDs
- Step 9: SPLIT BCVs paired to R2 STDs
- Step 10: Wait for mirror sync on R1 BCVs
- Step 10b: Wait for mirror sync on BCVs paired to R2 STDs
- Step 11: Cycle-end health check
- Step 11b: Wait for mirror sync on BCVs paired to R2 STDs

The system displays the number and name of the executed process step in the log. Depending on SRDF/AR parameter values, some steps can be skipped.

The cycle is repeated starting at Step 1 until a MODIFY STOP command is issued or the specified number of cycles is reached.

**Note:** “MODIFY STOP” on page 189 describes the MODIFY STOP command.

A minimum cycle time exists. When the cycle comes to the end too quickly, the system must wait.

The system gets the current time and calculates the wait interval. If the specified cycle time is greater than the actual time for the previous cycle, the system waits for the start of the next cycle. If it is less, the next cycle is started immediately if the CYCLE_OverFlow parameter of the MODIFY DEFINE or MODIFY START command is set to IMMEDIATE, as described in “MODIFY DEFINE” on page 175; otherwise, the system waits for the start of the next cycle time.

TF/Mirror automatically detects SRDF and mirror synchronization problems. When synchronization is stalled, repetitive BCVA000I messages are issued (if you specified the DEBUG(ITRKS) parameter of the MODIFY DEFINE command), each indicating the same invalid track count, followed by the BCVA091I message.

**Step 0: Health check**

The system checks for any failed devices and ensures that all devices are in the Ready status. This step is performed in the first cycle only.
Step 1A: Ensure R1 BCVs are offline

The system ensures that the R1 BCVs are offline.

**Note:** Currently, this step is bypassed.

Step 1B: Ensure BCVs paired to R2 STDs are offline

The system ensures that the BCVs paired to R2 STDs are offline.

**Note:** This check is skipped if the BYPTONL parameter of the MODIFY DEFINE or MODIFY START command is set to Yes and/or the BYPTBCV parameter of the MODIFY DEFINE or MODIFY START command is set to Yes, as described in “MODIFY DEFINE” on page 175.

Step 1C: Check R1-R2 invalid tracks and status

The system performs the following checks:

- Ensures that there is no invalid tracks on R2 owed to R1.
- Ensures that R2 STD status is R/O (read-only).

Step 1D: Check R2 STD local mirrors

The system ensures that R2 STD local mirrors have no invalid tracks.

SRDF/AR interrupts with messages BCVA067E and BCVA042E if an R2 STD is detected with invalid tracks on its single local mirror (unprotected R2) or both local mirrors (protected R2).

Step 1E: Check R1 BCV status

The system ensures that R1 BCVs are not in the TNR (Target Not Ready) status.
Step 2: Consistent SPLIT of R1 BCVs

The system performs a Consistent SPLIT of R1 BCVs, as shown in Figure 21.

![Figure 21 Consistent SPLIT of R1 BCVs](image)

**Note:** For more information about Consistent SPLITs, see “Consistent SPLIT” on page 43.

In the ECA mode, a Consistent SPLIT includes setting and clearing the ECA window. Use of ECA is controlled by the SYSTEM ECA parameter of the ADD command, as described in “ADD” on page 173.

**Note:** The Consistency Groups for z/OS Product Guide describes ECA.

After SPLIT requests to all involved R1 BCVs have been issued, the system waits for a predefined time and then resumes the SRDF link to propagate data from R1 BCVs to R2 STDs. The RESUME requests are sent in groups of a predefined group size with a predefined wait interval.

**Note:** The RESUME group size and the wait intervals are controlled using the RESUME parameter of the MODIFY DEFINE or MODIFY START command, as described in “MODIFY DEFINE” on page 175.

In a multihop environment, the R1 BCVs are split from their STDs at the recovery site B (see Figure 18 on page 76).

After the split, the R1 BCVs are in the Not Ready state.
Step 3: Wait for SRDF sync of R1s with R2s

The system checks that the R2 STDs are split from their BCVs, that is, the SPLIT condition still exists at the recovery site.

After resuming the SRDF link, the system waits for the R1-R2 synchronization to complete, as shown in Figure 22.

![Figure 22](image)

The system checks the current number of R1-R2 invalid tracks according to the settings made with the ITRKSYNC parameter of the MODIFY DEFINE or MODIFY START command, as described in “MODIFY DEFINE” on page 175. The synchronization is complete when there are no invalid tracks between R1 and R2. This means that R2 STDs now store a consistent copy of data, as shown in Figure 23.

![Figure 23](image)
Step 4: Check if R1 BCVs are Ready

**IMPORTANT**
Currently this step is bypassed.

The system checks if the R1 BCVs are in the *Ready* status.

Step 5: RE-ESTABLISH R1 BCVs

The system suspends the SRDF link between R1 and R2 and waits for R1 BCVs to synchronize their local mirrors.

Then the system performs a RE-ESTABLISH for the R1 BCVs, as shown in Figure 24.

![Figure 24](image)

**Figure 24** RE-ESTABLISH of R1 BCVs

*Note:* For more information about the RE-ESTABLISH operation, see “Re-synchronizing from STD to BCV: RE-ESTABLISH” on page 47.

You can set the number of concurrent RE-ESTABLISH requests and the corresponding wait intervals using the REEST parameter of the MODIFY DEFINE and MODIFY START command, as described in “MODIFY DEFINE” on page 175.

Step 6: Wait for mirror sync on BCVs paired to R2 STDs

The system waits for BCVs paired to R2 STDs to synchronize their local mirrors.

This step is skipped if the BYPTBCV parameter of the MODIFY DEFINE or MODIFY START command is set to Yes, as described in “MODIFY DEFINE” on page 175.
Step 7: RE-ESTABLISH BCVs paired to R2 STDs

The system checks that the R2 STDs are split from their BCVs, that is, the SPLIT condition still exists at the recovery site.

Then the system ensures that the BCVs paired to R2 STDs are offline.

**Note:** This check is performed only if the BYPTONL parameter of the MODIFY DEFINE or MODIFY START command is set to Yes, as described in “MODIFY DEFINE” on page 175.

After that, the system performs a RE-ESTABLISH for the BCVs paired to R2 STDs, as shown in Figure 25.

![Figure 25](image-url) **Figure 25** RE-ESTABLISH of BCVs paired to R2 STDs

**Note:** For more information about the RE-ESTABLISH operation, see “Re-synchronizing from STD to BCV: RE-ESTABLISH” on page 47.

You can set the number of concurrent RE-ESTABLISH requests and the corresponding wait intervals using the REEST _TGT parameter of the MODIFY DEFINE and MODIFY START command, as described in “MODIFY DEFINE” on page 175.

This step is skipped if the BYPTBCV parameter of the MODIFY DEFINE or MODIFY START command is set to Yes, as described in “MODIFY DEFINE” on page 175.

Step 8: Wait for TF/Mirror sync of R2 STDs

The system waits for R2 STDs to synchronize with their BCVs.

This step is skipped if the BYPTBCV parameter of the MODIFY DEFINE or MODIFY START command is set to Yes, as described in “MODIFY DEFINE” on page 175.
Step 9: SPLIT BCVs paired to R2 STDs

The system performs a SPLIT for the BCVs paired to R2 STDs, as shown in Figure 26.

Figure 26 SPLIT of BCVs paired to R2 STDs

This step is skipped if the BYPTBCV parameter of the MODIFY DEFINE or MODIFY START command is set to Yes, as described in “MODIFY DEFINE” on page 175.

Step 10: Wait for mirror sync on R1 BCVs

The system waits for R1 BCVs to synchronize their local mirrors.

Step 10b: Wait for mirror sync on BCVs paired to R2 STDs

The system waits for BCVs paired to R2 STDs to synchronize their local mirrors.

This step is executed only when the TargetBcvMirrorSyncAtEnd parameter of the MODIFY DEFINE command is set to CYCLE, as described in “MODIFY DEFINE” on page 175.

This step is skipped if the BYPTBCV parameter of the MODIFY DEFINE or MODIFY START command is set to Yes, as described in “MODIFY DEFINE” on page 175.

Step 11: Cycle-end health check

The system checks for any failed devices and ensures that all devices are in the Ready status.

Step 11b: Wait for mirror sync on BCVs paired to R2 STDs

The system waits for BCVs paired to R2 STDs to synchronize their local mirrors.

This step is executed only when the TargetBcvMirrorSyncAtEnd parameter of the MODIFY DEFINE is set to PROCESS, as described in “MODIFY DEFINE” on page 175.

This step is skipped if the BYPTBCV parameter of the MODIFY DEFINE or MODIFY START command is set to Yes, as described in “MODIFY DEFINE” on page 175.
SRDF/AR operations

Managing SRDF/AR processes

Table 20 lists operations for managing SRDF/AR processes.

Table 20 Managing SRDF/AR processes

<table>
<thead>
<tr>
<th>Operation</th>
<th>Associated statement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add process</td>
<td>ADD command</td>
</tr>
<tr>
<td>Configure process</td>
<td>MODIFY DEFINE command</td>
</tr>
<tr>
<td>Start process</td>
<td>MODIFY START command</td>
</tr>
<tr>
<td>View process details</td>
<td>MODIFY QUERY command</td>
</tr>
<tr>
<td>Pause process</td>
<td>MODIFY PAUSE command</td>
</tr>
<tr>
<td>Restart process</td>
<td>MODIFY RESTART command</td>
</tr>
<tr>
<td>Stop process</td>
<td>MODIFY STOP command</td>
</tr>
<tr>
<td>Delete process</td>
<td>DELETE command</td>
</tr>
</tbody>
</table>

Specifying devices for SRDF/AR process

Table 21 lists operations for specifying devices for an SRDF/AR process.

Table 21 Specifying devices for SRDF/AR process

<table>
<thead>
<tr>
<th>Operation</th>
<th>Associated statement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specify devices for process</td>
<td>MODIFY DEFINE command, DEVICE_List parameter</td>
</tr>
<tr>
<td>Use PowerMax/VMAX device numbers</td>
<td>GLOBAL command, LCLR1BCV parameter</td>
</tr>
<tr>
<td>for R1 BCVs</td>
<td></td>
</tr>
</tbody>
</table>

Setting up SRDF/AR environment

Table 22 lists operations for setting up the SRDF/AR environment.

Table 22 Setting up SRDF/AR environment

<table>
<thead>
<tr>
<th>Operation</th>
<th>Associated statement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Whether the SRDF/AR environment is single-hop</td>
<td>ADD command, HOP_TYPE parameter</td>
</tr>
<tr>
<td>or multihop</td>
<td></td>
</tr>
<tr>
<td>Whether SRDF/AR environment spans multiple</td>
<td>ADD command, SYSTEM parameter</td>
</tr>
<tr>
<td>LPARs</td>
<td></td>
</tr>
<tr>
<td>Set the maximum number of SRDF/AR devices per</td>
<td>ADD, MODIFY DEFINE, MODIFY START commands, MAXDEV parameter</td>
</tr>
<tr>
<td>storage system</td>
<td></td>
</tr>
<tr>
<td>Set the maximum number of SRDF/AR groups</td>
<td>ADD, GLOBAL commands, MAXGr parameter</td>
</tr>
<tr>
<td>Set the maximum number of SRDF/AR message</td>
<td>ADD command, MAXMsg parameter</td>
</tr>
<tr>
<td>buffers</td>
<td></td>
</tr>
</tbody>
</table>
Automating Replication with SRDF/AR

Table 22 Setting up SRDF/AR environment

<table>
<thead>
<tr>
<th>Operation</th>
<th>Associated statement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set up message handling</td>
<td>MODIFY DEFINE or MODIFY START command, MSGOPT parameter</td>
</tr>
<tr>
<td>Set I/O control mode to ECA or IOS level</td>
<td>ADD command, SYSTEM parameter</td>
</tr>
<tr>
<td>Enable debugging</td>
<td>MODIFIY DEFINE or MODIFY START command, GLOBAL command, DEBUG parameter</td>
</tr>
<tr>
<td>Export SRDF/AR statements to a file</td>
<td>MODIFY EXPORT command</td>
</tr>
<tr>
<td>Set the maximum allowed return code</td>
<td>GLOBAL command, MAXRC parameter</td>
</tr>
</tbody>
</table>

Setting up cycle

Table 23 lists operations for setting up a process cycle.

Table 23 Setting up process cycle

<table>
<thead>
<tr>
<th>Operation</th>
<th>Associated statement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set cycle time</td>
<td>MODIFY DEFINE or MODIFY START command, CYCLE parameter</td>
</tr>
<tr>
<td>Set number of cycles</td>
<td>MODIFY DEFINE or MODIFY START command, CYCLE parameter</td>
</tr>
<tr>
<td>Determine when to start a new cycle</td>
<td>MODIFY DEFINE or MODIFY START command, CYCLE_OverFlow parameter</td>
</tr>
</tbody>
</table>

Setting up cycle steps

Table 24 lists operations for setting up cycle steps.

Table 24 Setting up cycle steps

<table>
<thead>
<tr>
<th>Operation</th>
<th>Associated statement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apply process policies</td>
<td>MODIFY DEFINE, POLICY parameter</td>
</tr>
<tr>
<td>Bypass processing of BCVs paired to R2 STDs</td>
<td>MODIFY DEFINE or MODIFY START command, BYPTBCV parameter</td>
</tr>
<tr>
<td>Bypass the online/offline status check for BCVs paired to R2 STDs</td>
<td>MODIFY DEFINE or MODIFY START command, BYPTONL parameter</td>
</tr>
<tr>
<td>When to run mirror synchronization check for the BCVs paired to R2 STDs</td>
<td>MODIFY DEFINE command, TARGETBCVMIRRORSYNCATEND parameter</td>
</tr>
<tr>
<td>Set up invalid track check</td>
<td>MODIFY DEFINE command, ITRKSYNC parameter</td>
</tr>
<tr>
<td>Set I/O processing timeout and action during Step 2: Consistent SPLIT of R1 BCVs</td>
<td>MODIFY DEFINE or MODIFY START command, TIMEOUT parameter</td>
</tr>
<tr>
<td>Set up SRDF RESUME</td>
<td>MODIFY DEFINE or MODIFY START command, RESUME parameter</td>
</tr>
<tr>
<td>Set up RE-ESTABLISH for R1 BCVs</td>
<td>MODIFY DEFINE or MODIFY START command, REEST parameter</td>
</tr>
</tbody>
</table>
## Table 24 Setting up cycle steps

<table>
<thead>
<tr>
<th>Operation</th>
<th>Associated statement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set up RE-ESTABLISH for BCVs paired to R2 STDs</td>
<td>MODIFY DEFINE or MODIFY START command, REEST_TGT parameter</td>
</tr>
<tr>
<td>Convert a full ESTABLISH to an incremental RE-ESTABLISH</td>
<td>GLOBAL command, FASTESTablish parameter</td>
</tr>
<tr>
<td>Automatically release STD holds for failed ESTABLISH and RE-ESTABLISH commands</td>
<td>GLOBAL command, AUTORELease parameter</td>
</tr>
<tr>
<td>Define operations for which the Multi-Attach feature is enabled</td>
<td>GLOBAL command, MULTIATTach parameter</td>
</tr>
</tbody>
</table>
**SRDF/AR examples**

**Single-hop environment**

**Example 1** To add, configure, and start an SRDF/AR process named PROCESSONE:

```
ADD 10, PROCESSONE, HOP_TYPE(SINGLE)
MODIFY 11, PROCESSONE, DEFINE, DEVICE_LIST(SRCSTD(A000-A002),
    SRCR1BCV(A100-A102),
    TGTCV(000001-000003))
MODIFY 11, PROCESSONE, DEFINE, DEVICE_LIST(SRCSTD(B100-B102),
    SRCR1BCV(B180-B182),
    TGTCV(000004-000006))
MODIFY 12, PROCESSONE, START, CYCLE(00:00:00), TIMEOUT(15)
```

**Note:** “SRDF/AR commands” on page 173 describes SRDF/AR commands.

The ADD statement at sequence level 10 creates a new process named PROCESSONE. The HOP_TYPE(SINGLE) parameter indicates that the process is to be run in a single-hop environment.

The two MODIFY DEFINE statements at sequence level 11 add the configuration shown in Figure 27. Note that Site A and Site B devices are identified with their CUUs, whereas Site C devices are specified by their PowerMax/VMAX device numbers.

![Figure 27 Device configuration for SRDF/AR example 1](image)

The MODIFY START statement at sequence level 12 starts the process PROCESSONE. The process runs until an explicit STOP is issued for the process. CYCLE(00:00:00) defines zero waiting time between the cycles, which means that each subsequent cycle begins immediately after the previous one. TIMEOUT(15) specifies that the maximum time I/O is held waiting for the Consistent SPLIT to complete is 15 seconds.
Sample debug log

The following output is a sample SRDF/AR debug log for single-hop implementation.

Note: The actual debug output may differ. See the Mainframe Enablers Message Guide for explanation of additional messages that may appear in the log.

Example 2 To stop process PROCESSONE:
MODIFY 20,PROCESSONE,STOP

Example 3 To query the status of process PROCESSONE:
MODIFY 20,PROCESSONE,QUERY(STATUS)

Example 4 To restart the process PROCESSONE:
MODIFY 20,PROCESSONE,RESTART

Example 5 To stop again and delete process PROCESSONE:
MODIFY 30,PROCESSONE,STOP
DELETE 31,PROCESSONE
Multihop environment

Example 6 To add, configure, and start an SRDF/AR process named PROCESSTWO:

ADD 01, PROCESSTWO, HOP_TYPE(MULTI)
MODIFY 02, PROCESSTWO, DEFINE, DEVICE_LIST(SRCR1BCV(000140-000143),
SRCSTD(D010-D013),
TGTBCV(00A0-00A3))
MODIFY 03, PROCESSTWO, START, CYCLE(00:00:00), TIMEOUT(15)

Note: “SRDF/AR commands” on page 173 describes SRDF/AR commands.

The ADD statement at sequence level 01 creates a new process named PROCESSTWO. The HOP_TYPE(MULTI) parameter indicates that the process is to be run in a multihop environment.

The MODIFY DEFINE statement at sequence level 02 adds the configuration shown in Figure 28.

Figure 28 Device configuration for SRDF/AR example 6

The MODIFY START statement at sequence level 03 starts the process PROCESSTWO. The process runs until an explicit STOP is issued for the process. CYCLE(00:00:00) defines zero waiting time between the cycles, which means that each subsequent cycle begins immediately after the previous one. TIMEOUT(15) specifies that the maximum time I/O is held waiting for the Consistent SPLIT to complete is 15 seconds.
The following output is a sample SRDF/AR debug log for multihop implementation.

06:00:31.10 BCVA002I Process MHVV2SAR, Beginning AMH cycle 1, Version 8.3.0 (00)
06:00:39.67 BCVA086I Process MHVV2SAR, Source 0001967-01307, MCL 5977, QK 4310
06:00:39.67 BCVA087I Process MHVV2SAR, Bunker 0001977-00004, MCL 5977, RAG D8 (Clone)
06:00:39.67 BCVA088I Process MHVV2SAR, Target 0001970-00165, MCL 5977, RAG D8.10 (Clone,Multa)

06:00:40.19 BCVA000I 00: Mirror Status Check
06:01:45.28 BCVA000I 01C: Target R2-R1 Status check
06:01:51.77 BCVA094I Process MHVV2SAR, R1 007E30 has R2 Invalid Tracks, Symm 0001967-01307, RAG D8.10
06:01:51.77 BCVA000I      R1INV 007E30 ITRK count = 000006DE
06:01:51.77 BCVA092I Process MHVV2SAR, first cycle will not be consistent
06:01:51.77 BCVA094I Process MHVV2SAR, R1 007E31 has R2 Invalid Tracks, Symm 0001967-01307, RAG D8.10
06:01:51.77 BCVA000I      R1INV 007E31 ITRK count = 000006DE
06:01:51.77 BCVA092I Process MHVV2SAR, first cycle will not be consistent
06:01:51.77 BCVA094I Process MHVV2SAR, R1 007E32 has R2 Invalid Tracks, Symm 0001967-01307, RAG D8.10
06:01:51.77 BCVA000I      R1INV 007E32 ITRK count = 000006DE
06:01:51.77 BCVA092I Process MHVV2SAR, first cycle will not be consistent
06:01:51.77 BCVA094I Process MHVV2SAR, first cycle will not be consistent
06:01:51.77 BCVA000I 01D: Target R2 Local Mirror Check
06:01:58.48 BCVA000I 01E: Source R1 Status check

06:02:02.26 BCVA000I 02: Consistent Split
06:02:03.26 BCVA000I      PAV Optimizer suspended, Symm 0001967-01307, TOD=D1BA709083EDFA95, ID=D083899F
06:02:03.26 BCVA000I      Window Set, Symm 0001967-01307 Duration 0076, Interval 0038, Run# 0001
06:02:03.26 BCVA000I        Start device 001523, Last 001525

06:02:03.37 BCVA000I      PAV Optimizer resumed, Symm 0001967-01307, TOD=D1BA7091930CC493, ID=D183899F
06:02:03.37 BCVA000I      Window Clear, Symm 0001967-01307, Run# 0001
06:02:03.37 BCVA000I        Start device 001523, Last 001525

06:02:08.42 BCVA000I 03: Source R1 sync wait
06:02:08.42 BCVA000I      Check for Split-in-progress

06:02:08.78 BCVA000I 04: Source R1 Status check

06:03:09.19 BCVA000I 08: Target R2 sync wait
06:03:09.19 BCVA000I      Check for BCV Synchronization
06:03:39.76 BCVA000I 09: Split Target BCVs
06:05:09.94 BCVA000I 05: Source BCV Ready - Bypassed
06:05:40.15 BCVA000I 06: Target BCV mirror sync check
06:06:10.33 BCVA000I 07: Re-Est Target volumes

06:06:12.67 BCVA000I 10: Source R1 sync wait
06:06:12.67 BCVA000I      Check for Split-in-progress
06:06:12.67 BCVA000I      Symm 0001967-01307, Split-in-progress for 007E32 (3 devices), RAG D8
06:06:10.33 BCVA000I 11: Mirror Status check
06:06:10.33 BCVA000I      Invalid Track Check
06:06:12.67 BCVA000I      Symm 0001967-01307, Dev 007E30 0000006DE, Total 000000DBC
06:06:29.90 BCVA000I      Symm 0001967-01307, Dev 007E30 0000006DE, Total 000000DBC
06:06:47.26 BCVA000I      Symm 0001967-01307, Dev 007E30 0000006DE, Total 0000006DE
06:07:04.66 BCVA000I      Symm 0001967-01307, Dev 007E30 0000006DE, Total 0000006DE
06:07:21.92 BCVA000I      Symm 0001967-01307, Dev 007E30 0000006DE, Total 0000006DE
06:07:39.15 BCVA000I      Symm 0001967-01307, Dev 007E30 0000006DE, Total 0000006DE
06:07:56.53 BCVA000I 10: Source BCV sync wait
06:07:56.53 BCVA000I      Source Resume
06:08:11.53 BCVA000I 05: Re-Est Source volumes
06:08:22.14 BCVA000I 06: Target BCV mirror sync check
06:08:22.14 BCVA000I      Source Suspend
06:08:27.70 BCVA000I 07: Re-Est Target volumes
06:08:35.73 BCVA000I 08: Target R2 sync wait
06:08:43.57 BCVA000I 09: Split Target BCVs
06:08:43.66 BCVA000I 10: Source BCV sync wait
06:08:46.48 BCVA000I 11: Mirror Status check
06:09:48.69 BCVA000I 11: Source BCV Ready - Bypassed
Recovery and troubleshooting

Restarting applications at recovery site

In case of outage at the production site, you can restart applications using the consistent copy of data at the recovery site.

⚠️ CAUTION

Your configuration, the specific nature of the outage, and any unique circumstances that may exist dictate the specific recovery steps that are required. ALWAYS contact Dell EMC Customer Support for assistance in a recovery situation. An incorrect action during the recovery process may result in data corruption.

Recovery steps are as follows:

- **Step 1**: Obtain operational z/OS image
- **Step 2**: Clear SRDF/AR locks
- **Step 3**: Bring SRDF links offline
- **Step 4**: Determine SPLIT state of BCVs paired to R2 STDs
- **Step 5**: Synchronize R2 STDs and their BCVs
- **Step 6**: Check synchronization of R2 STDs and their BCVs
- **Step 7**: Split BCVs paired to R2 STDs
- **Step 8**: Restart applications

When performing the steps, you can also use the RMT(\(\text{cuu,srdfgroup}\)) syntax of TF/Mirror commands to redirect the command to a remote storage system.

**Step 1: Obtain operational z/OS image**

The host must be connected to the recovery site and have SRDF Host Component running. The R2 STDs must be read/write-enabled.

SRDF Host Component must be running to make the recovery site available. An operational z/OS image attached to the storage system containing the SRDF/AR target volumes is required.

---

**Note**: The *SRDF Host Component for z/OS Product Guide* provides information about making the operational z/OS image available.
Step 2: Clear SRDF/AR locks

SRDF/AR obtains a long-term lock on all STDs and BCVs except R2 STDs when the SRDF/AR process is first started. These locks are held until the process ends and cannot be taken over by TF/Mirror.

You can use the ReleaseDeviceLock parameter of the TF/Mirror batch job to release any SRDF/AR locks that were not freed due to abnormal termination of the SRDF/AR process, as described in “Running TF/Mirror recovery jobs” on page 23.

If a process abnormally terminated without clearing the Active flag, you must delete the process using the SRDF/AR DELETE command with the FORCE parameter, as described in “DELETE” on page 174, and redefine the process.

After the locks are cleared, the named SRDF/AR process can be started normally.

If, for some reason, the SRDF/AR process does not exist, you need to define the SRDF/AR process using TF/Mirror commands.

Step 3: Bring SRDF links offline

At the recovery site, bring the SRDF links offline to protect R2 STDs from inadvertent resynchronization from R1 BCVs using the following SRDF Host Component command:

```
#SC LINK,ALL,OFFLINE
```

Note: The SRDF Host Component for z/OS Product Guide describes the #SC LINK command and its parameters.

Step 4: Determine SPLIT state of BCVs paired to R2 STDs

Determine the current SPLIT state of BCVs paired to R2 STDs by issuing the following TF/Mirror command:

```
QUERY seq#,LCL(cuu)
```

Note: “QUERY” on page 141 describes the QUERY command.

Note status values in the STATUS field and proceed to the next step.

Step 5: Synchronize R2 STDs and their BCVs

Synchronize devices from the consistent copy.

* If all BCVs have the “AVAIL” value in the STATUS field, check for R1-R2 invalid tracks in the SRDF/AR configuration by issuing the following SRDF Host Component command:

```
#SQ VOL,LCL(cuu,srdfgroup_of_R2STD),INV_TRKS
```

Note: The SRDF Host Component for z/OS Product Guide describes the #SQ VOL command and its parameters.
If there are no R1-R2 invalid tracks, the SRDF synchronization is complete, and the R2 STDs hold the most recent point-in-time consistent copy. Issue the following TF/Mirror command to all BCVs paired to R2 STDs:

\[
\text{RE-ESTABLISH } \text{seq}, \text{LCL}(\text{cuu}, \text{symdv#bcv-symdv#bcv}), \text{WAIT}
\]

**Note:** “RE-ESTABLISH” on page 147 describes the RE-ESTABLISH command.

If any of R2 STDs has R1-R2 invalid tracks, then the BCVs paired to R2 STDs hold the most recent point-in-time consistent copy. Issue the following TF/Mirror command to all BCVs paired to R2 STDs:

\[
\text{RESTORE } \text{seq}, \text{LCL}(\text{cuu}, \text{symdv#bcv-symdv#bcv}), \text{WAIT}
\]

**Note:** “RESTORE” on page 155 describes the RESTORE command.

If some of BCVs do not have the “AVAIL” value in the STATUS field, the R2 STDs hold the most recent point-in-time consistent copy. Issue the following TF/Mirror command to BCVs that have the “AVAIL” status:

\[
\text{RE-ESTABLISH } \text{seq}, \text{LCL}(\text{cuu}, \text{symdv#bcv-symdv#bcv}), \text{WAIT}
\]

**Note:** “RE-ESTABLISH” on page 147 describes the RE-ESTABLISH command.

If none of BCVs have the “AVAIL” value in the STATUS field, proceed to Step 6: Check synchronization of R2 STDs and their BCVs.

**Step 6: Check synchronization of R2 STDs and their BCVs**

Issue the following TF/Mirror command and ensure that there are no invalid tracks between R2 STDs and their BCVs:

\[
\text{QUERY } \text{seq}, \text{LCL}(\text{cuu})
\]

**Note:** “QUERY” on page 141 describes the QUERY command.

**Step 7: Split BCVs paired to R2 STDs**

Issue the following TF/Mirror command for all BCVs:

\[
\text{SPLIT } \text{seq}, \text{LCL}(\text{cuu}, \text{symdv#bcv-symdv#bcv}), \text{HOLD}
\]

**Note:** “SPLIT” on page 161 describes the SPLIT command.

**Result:** Data on R2 STDs and their BCVs are identical.

**Step 8: Restart applications**

You can now restart applications from R2 STDs or from BCVs.

Restart from R2 STDs is recommended for the following reasons:

- This allows using BCVs for normal TF/Mirror operations at the recovery site if needed.
- This positions for using R1/R2 personality swap as part of the go-home process.
The go-home process requires use of the SRDF recovery procedure 2, where R2s are used for operations (read-write enabled) and later resynchronized with R1s.

**Note:** For information about the SRDF recovery procedures, see *SRDF Host Component for z/OS Product Guide*.

### Drive failure recovery

When devices are *Not Ready* as a result of a drive failure, SRDF/AR identifies the event and issues the BCVA012W/E, BCVA063E, and BCVA058A messages.

During drive replacement, there are periods of configuration locking and dynamic drive replacements that can cause collisions with SRDF/AR. SRDF/AR recognizes this, and issues the BCVA012W/E and BCVA058A messages.

**Table 25** list drive failure recovery procedures with the following assumptions:

- R1 BCVs and BCVs paired to R2 STDs reside on separate physical systems from the STDs or R2 STDs they are associated with during SRDF/AR RE-ESTABLISH.
- R2 STDs are protected devices.
- Hot spares are not active for R1 BCVs or BCVs paired to R2 STDs.

For all steps except 1A-E, 8 and 9, the BCVA063E message shows the device type, PowerMax/VMAX device number, physical drive information and storage system serial number.

**Table 25** Drive failure recovery procedures  (page 1 of 3)

<table>
<thead>
<tr>
<th>Step</th>
<th>Device failure</th>
<th>SRDF/AR environment type</th>
<th>SRDF/AR status</th>
<th>Procedure</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 0: Health check</td>
<td>STD</td>
<td>All</td>
<td>Paused*</td>
<td>a. Replace drive. b. CONTinue.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>R1 BCV</td>
<td>All</td>
<td>Paused</td>
<td>a. Replace drive. b. CONTinue.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>R2 STD</td>
<td>All</td>
<td>Paused</td>
<td>a. Replace drive. b. CONTinue.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>BCV paired to R2 STD</td>
<td>All</td>
<td>Paused</td>
<td>a. Replace drive. b. CONTinue.</td>
<td></td>
</tr>
<tr>
<td>Step 1A: Ensure R1 BCVs are offline</td>
<td>R1 BCV - not ESTABLISHED</td>
<td>All</td>
<td>Stopped</td>
<td>a. RE-ESTABLISH. b. Start SRDF/AR.</td>
<td></td>
</tr>
<tr>
<td>Step 1B: Ensure BCVs paired to R2 STDs are offline</td>
<td>BCV paired to R2 STD - not SPLIT</td>
<td>All</td>
<td>Stopped</td>
<td>a. SPLIT. b. Start SRDF/AR.</td>
<td></td>
</tr>
<tr>
<td>Step 1C: Check R1-R2 invalid tracks and status</td>
<td>STD paired to R1 BCVs - TNR or LNR</td>
<td>Single-hop</td>
<td>Stopped</td>
<td>Resume and start SRDF/AR.</td>
<td></td>
</tr>
</tbody>
</table>
Table 25  Drive failure recovery procedures  (page 2 of 3)

<table>
<thead>
<tr>
<th>Step</th>
<th>Device failure</th>
<th>SRDF/AR environment type</th>
<th>SRDF/AR status</th>
<th>Procedure</th>
<th>Notes</th>
</tr>
</thead>
</table>
| Step 2: Consistent SPLIT of R1 BCVs       | R1 BCV         | All                      | Paused         | a. Replace drive.  
                             |                |                          |                | b. CONTinue.                                                | If resume not used, R2 STDs may be out of sync. In this case, operations on BCVs paired to R2 STDs are bypassed for this cycle to preserve the last consistency point. |
| Step 3: Wait for SRDF sync of R1s with R2s| R2 STD         | All                      | Paused         | a. Replace drive.  
                             |                |                          |                | b. CONTinue.                                                                                                                     | 1) If SRDF/AR is paused again due to links or R1 BCV in the TNR status, correct and confirm decreasing the ITRK parameter value. Then CONTinue. Otherwise, contact Dell EMC Customer Support. |
|                                           | R1 BCV         | All                      | Paused         | a. Replace drive.  
                             |                |                          |                | b. CONTinue.                                                                                                                     | 2) If SRDF/AR is paused again, repeat 1). |
| Step 4: Check if R1 BCVs are Ready        | R1 BCV         | All                      | Bypassed       | No action.             |                                                                                                                                 |
|                                           |                |                          | Paused         | a. Replace drive.  
                             |                |                          |                | b. CONTinue.                                                                                                                     |                                                                                                                                 |
| Step 5: RE-ESTABLISH R1 BCVs              | R2 STD_b       | Multihop                 | Paused         | a. Replace drive.  
                             |                |                          |                | b. CONTinue.                                                                                                                     | RE-ESTABLISH of R1 BCVs resumed. |
|                                           | R1 BCV         | All                      | Paused         | a. Replace drive.  
                             |                |                          |                | b. CONTinue.                                                                                                                     | RE-ESTABLISH of R1 BCVs resumed. |
|                                           | STD paired to  | Single-hop               | Paused         | a. Replace drive.  
                             | R1 BCV         |                          |                | b. CONTinue.                                                                                                                     | RE-ESTABLISH of R1 BCVs resumed. |
| Step 6: Wait for mirror sync on BCVs      | BCV paired to  | All                      | Paused         | a. Replace drive.  
                             | R2 STD         |                          |                | b. CONTinue.                                                                                                                     | If no TGTBCV ITRKs this step. |
| paired to R2 STDs                         | M2 of BCV      | All                      | Paused         | a. Replace drive.  
                             | paired to R2   |                          |                | b. CONTinue.                                                                                                                     | Procedure is bypassed, otherwise follow the procedure for paused process. |
|                                           | STD           | All                      | Paused         | a. Replace drive.  
                             | RCV paired to  |                          |                | b. CONTinue.                                                                                                                     |                                                                 |
|                                           | R2 STD         | All                      | Paused         | a. Replace drive.  
                             |                |                          |                | b. CONTinue.                                                                                                                     | RE-ESTABLISH of BCVs paired to R2 STDs resumed. |
| Step 7: RE-ESTABLISH BCVs paired to R2 STDs| BCV paired to  | All                      | Paused         | a. Replace drive.  
                             | R2 STD         |                          |                | b. CONTinue.                                                                                                                     |                                                                 |
|                                           | R2 STD         | All                      | Paused         | a. Replace drive.  
                             |                |                          |                | b. CONTinue.                                                                                                                     |                                                                 |
| Step 8: Wait for TF/Mirror sync of R2 STDs| BCV paired to  | All                      | Paused         | a. Replace drive.  
                             | R2 STD         |                          |                | b. CONTinue.                                                                                                                     | 1) If SRDF/AR is paused again, correct and confirm decreasing the ITRK parameter value. Then CONTinue. Otherwise, contact Dell EMC Customer Support.  
                             |                |                          |                | b. CONTinue.                                                                                                                     | 2) If SRDF/AR is paused again, repeat 1).                                 |
|                                           | R2 STD         | All                      | Paused         | a. Replace drive.  
                             |                |                          |                | b. CONTinue.                                                                                                                     |                                                                 |
When reporting a problem to Dell EMC, provide the following information to reduce phone time and speed up response from Dell EMC Customer Support.

**General information**

- SRDF/AR definitions for all your SRDF/AR processes
  
  Provide this information initially and each time you make changes to the SRDF/AR environment.

- Storage system serial number and operating environment level for each storage system in the SRDF/AR configuration

- Host-to-PowerMax/VMAX device relationships for each storage system in the SRDF/AR configuration (UCB to PowerMax/VMAX device number)

- Current host software levels for Dell EMC products including maintenance levels updated as needed or on a regular basis.

---

### Table 25: Drive failure recovery procedures (page 3 of 3)

<table>
<thead>
<tr>
<th>Step</th>
<th>Device failure</th>
<th>SRDF/AR environment type</th>
<th>SRDF/AR status</th>
<th>Procedure</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 9: SPLIT BCVs paired to R2 STDs</td>
<td>BCV paired to R2 STD</td>
<td>All</td>
<td>Paused</td>
<td>a. Replace drive. b. CONTinue.</td>
<td>SPLIT resumed.</td>
</tr>
<tr>
<td>Step 10: Wait for mirror sync on R1 BCVs Step 10b: Wait for mirror sync on BCVs paired to R2 STDs</td>
<td>R2 STD</td>
<td>Multihop</td>
<td>Paused</td>
<td>a. Replace drive. b. CONTinue.</td>
<td>1) If SRDF/AR is paused again, correct and confirm decreasing the ITRK parameter value. Then CONTinue. Otherwise, contact Dell EMC Customer Support. 2) If SRDF/AR is paused again, repeat 1).</td>
</tr>
<tr>
<td></td>
<td>R1 BCV</td>
<td>All</td>
<td>Paused</td>
<td>a. Replace drive. b. CONTinue.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>R1 BCV</td>
<td>Single-hop</td>
<td>Paused</td>
<td>a. Replace drive. b. CONTinue.</td>
<td></td>
</tr>
<tr>
<td>Step 11: Cycle-end health check Step 11b: Wait for mirror sync on BCVs paired to R2 STDs</td>
<td>STD paired to R1 BCV</td>
<td>Single-hop</td>
<td>Paused</td>
<td>a. Replace drive. b. CONTinue.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>R1 BCV</td>
<td>Single-hop</td>
<td>Paused</td>
<td>a. Replace drive. b. CONTinue.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>BCV paired to R2 STD</td>
<td>Single-hop</td>
<td>Paused</td>
<td>a. Replace drive. b. CONTinue.</td>
<td></td>
</tr>
</tbody>
</table>

a. TF/Mirror automatically pauses the SRDF/AR process after all API and SYSCALL errors. You no longer have to issue a SRDF/AR PAUSE command to pause the SRDF/AR process when a drive failure is detected.
Step-dependent information

Table 23 lists information to be provided depending on the SRDF/AR process step at which the failure occurred.

To diagnose certain intermittent problems, Dell EMC often requests a GTF trace of the gatekeeper devices for the SRDF/AR process. This would apply to Consistent SPLIT timeouts, elongation, other unknown causes. This allows Dell EMC to further diagnose issues and to help identify areas of concern.

Table 26  Step-dependent information

<table>
<thead>
<tr>
<th>Step</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 0: Health check</td>
<td>TF/Mirror queries of all storage systems</td>
</tr>
<tr>
<td>Step 1A: Ensure R1 BCVs are offline</td>
<td>SRDF/AR query to display the device relationships</td>
</tr>
<tr>
<td>Step 1B: Ensure BCVs paired to R2 STDs are offline</td>
<td>SRDF query for each pair of storage systems</td>
</tr>
<tr>
<td>Step 1C: Check R1-R2 invalid tracks and status</td>
<td>In addition, check SRDF/AR definitions, device statuses, bin files and device relationships for conflicts.</td>
</tr>
<tr>
<td>Step 1D: Check R2 STD local mirrors</td>
<td></td>
</tr>
<tr>
<td>Step 1E: Check R1 BCV status</td>
<td></td>
</tr>
<tr>
<td>Step 2: Consistent SPLIT of R1 BCVs</td>
<td>TF/Mirror queries of storage systems where R1 devices reside: Site A for single-hop, Site B for multihop</td>
</tr>
<tr>
<td></td>
<td>SRDF queries of all storage systems</td>
</tr>
<tr>
<td></td>
<td>SCFLOG</td>
</tr>
<tr>
<td></td>
<td>SYSLOG +/- 15 minutes</td>
</tr>
<tr>
<td></td>
<td>GTF trace for Consistent SPLIT timeouts or other errors</td>
</tr>
<tr>
<td></td>
<td>Possibly EREP</td>
</tr>
<tr>
<td>Step 3: Wait for SRDF sync of R1s with R2s</td>
<td>TF/Mirror queries of storage systems where R1 devices reside: Site A for single-hop, Site B for multihop</td>
</tr>
<tr>
<td></td>
<td>SRDF queries of all storage systems</td>
</tr>
<tr>
<td></td>
<td>SYSLOG +/- 1 hour</td>
</tr>
<tr>
<td>Step 4: Check if R1 BCVs are Ready</td>
<td>TF/Mirror queries of storage systems where R1 devices reside: Site A for single-hop, Site B for multihop</td>
</tr>
<tr>
<td></td>
<td>SRDF queries of all storage systems</td>
</tr>
<tr>
<td></td>
<td>SYSLOG +/- 15 minutes</td>
</tr>
<tr>
<td></td>
<td>SCFLOG</td>
</tr>
<tr>
<td>Step 5: RE-ESTABLISH R1 BCVs</td>
<td></td>
</tr>
<tr>
<td>Step 6: Wait for mirror sync on BCVs paired to R2 STDs</td>
<td>TF/Mirror queries of storage systems where R2 STDs reside: Site B for single-hop, Site C for multihop</td>
</tr>
<tr>
<td></td>
<td>SYSLOG +/- 15 minutes</td>
</tr>
<tr>
<td></td>
<td>SCFLOG</td>
</tr>
<tr>
<td>Step 7: RE-ESTABLISH BCVs paired to R2 STDs</td>
<td></td>
</tr>
<tr>
<td>Step 8: Wait for TF/Mirror sync of R2 STDs</td>
<td>TF/Mirror queries of storage systems where R2 STDs reside: Site B for single-hop, Site C for multihop</td>
</tr>
<tr>
<td></td>
<td>SYSLOG +/- 15 minutes</td>
</tr>
<tr>
<td></td>
<td>SCFLOG</td>
</tr>
<tr>
<td>Step 9: SPLIT BCVs paired to R2 STDs</td>
<td></td>
</tr>
<tr>
<td>Step 10: Wait for mirror sync on R1 BCVs</td>
<td>TF/Mirror queries of storage systems where R1 devices reside: Site A for single-hop, Site B for multihop</td>
</tr>
<tr>
<td></td>
<td>SYSLOG +/- 15 minutes</td>
</tr>
<tr>
<td></td>
<td>SCFLOG</td>
</tr>
<tr>
<td>Step 10b: Wait for mirror sync on BCVs paired to R2 STDs</td>
<td></td>
</tr>
<tr>
<td>Step 11: Cycle-end health check</td>
<td></td>
</tr>
<tr>
<td>Step 11b: Wait for mirror sync on BCVs paired to R2 STDs</td>
<td></td>
</tr>
</tbody>
</table>
CHAPTER 5
Monitoring SRDF/AR with SRDF/AR Monitor

This chapter covers the following topics:

♦ Introduction to SRDF/AR Monitor ................................................................. 104
♦ Getting started ............................................................................................. 104
♦ Monitoring SRDF/AR environment ............................................................ 109
Introduction to SRDF/AR Monitor

The SRDF/AR Monitor\(^1\) (SARTOOL) allows you to monitor and manage SRDF/AR processes.

Restrictions

The following restrictions apply to SARTOOL:

- SARTOOL does not support Kanji terminal types.
- Before you use the SRDF/AR option, you must specify the session control options. The options let you set the parameters you want to use for the session. Note that you may receive errors if the dataset you are trying to allocate is on the allocated volume but is not cataloged. To resolve this, you need to catalog the dataset.
- There is no support for attention exit. If you use attention to exit a long running request, you may experience one of the following conditions:
  - You may receive an S001 return code.
  - You may need to reply to the TSO “HI” prompt.
  - The SARTOOL EXEC may be terminated.

If the SARTOOL EXEC abends, the LOCK file remains allocated. If this occurs, you must free the LOCK file (with TSO FREE F(LOCK)) before you can execute the SARTOOL EXEC again.

Getting started

SARTOOL is installed with ResourcePak Base.

Note: The Mainframe Enablers Installation and Customization Guide describes the ResourcePak Base installation process.

The ISPF interface requires the following software:

- ISPF 3.3.0 or later
- TSO/E 2.3.1 or later

After installation, SARTOOL is required to customize the primary ISPF panel. SARTOOL is located in the Mainframe Enablers SAMPLIB.

The following is a listing of the SARTOOL REXX member.

```rexx
/* REXX */
/* ******************************************
/* Dell EMC Product Component ISPF Interface Libdef Exec*/
/* ******************************************
y=OUTTRAP('Alloc.')
"ALLOC F(LOCK) DUMMY OLD"
alloc_rc = rc
y=OUTTRAP('Off')
IF alloc_rc >0 THEN
```

1. Another product name is SAR Monitor (Symmetrix Automated Replication).
DO
ZEDMSG="Exec Already Active"
ZEDLMSG="This Exec cannot be started because it is already active"
"ISPEXEC SETMSG MSG(ISRZ001)"
Exit
END

/**************************** DEFINES THE LIBRARY SEARCH ORDER ****************************/
DS_PREFIX = "INSTALLED.LIB" /* specify the EMCSF DS-PREFIX here */
TYPEM = ''
IDM = ''
APPID = "EMC"
/**************************** DEFINES THE LIBRARY SEARCH ORDER ****************************/
/* Note: The code below allows you to set variables for all users using */
/* this REXX clist. While the users can modify these variables in */
/* their current session, these variables will override them the */
/* next time this exec is invoked. Set these variables to a value */
/* or '' otherwise unwanted variable values will result. Setting */
/* a variable to '' allows the user to specify their own value */
/* and retain it until the variable is set in this procedure. */
/**************************** DEFINES THE LIBRARY SEARCH ORDER ****************************/
ISPLLIB = '''YOUR.ISP.SISPLOAD'''
ISPLLIB = '''YOUR.ISP.SISPMENU'''
ISPSLIB = '''YOUR.ISP.SISPSENU'''
ISPPLIB = '''YOUR.ISP.SISPPENU'''
ISPTLIB = '''YOUR.ISP.SISPTENU'''
SCF_SUFFIX = "EMC"
CMD_PREFIX = "YOUR_HC_CMD_PREFIX"
UNIT = 'SYSALLDA'
VOLUME = ''
UNITSW = '3390'
SAR_RACF_START = 'SAR-START'
SAR_RACF_STOP = 'SAR-STOP'
SAR_LOAD_LIBRARY = 'DS_PREFIX.LINKLIB'
TIMEFINDER_LIBRARY = 'DS_PREFIX.LINKLIB'
/**************************** DEFINES THE LIBRARY SEARCH ORDER ****************************/
ADDRESS ISPEXEC 'CONTROL ERRORS RETURN'
/******************** EXECUTE THE EMC TOOLS MAIN EXEC ********************/
"ISPEXEC SELECT CMD(%ESAXMAIN
"ISPLLIB"/
"ISPLLIB"/
"ISPSLIB"/
Customizing SARTOOL

To customize SARTOOL REXX:

1. Edit the following line:

   DS_PREFIX = INSTALLED.LIB => specify the EMCSCF library-set prefix

2. APPID defaults to EMC#. If this changes after the first use of SARTOOL, you must reenter the setup information. The APPID specifies how TSO is to keep track of session information.

3. Set variables to valid values.

   The variables allow you to globally set certain variables for all users. While users can modify these variables in their current session, they will not be retained across sessions. Set these variables to a valid value or " (two single quotes), otherwise unwanted variable values will result. Setting a variable to "allows users to specify their own value which will be retained (until the variable is set in SARTOOL) across sessions. Any variables that will not be used must be set to ".

   **Note:** Single quotes are required for dataset names and double quotes are required for all variables.

   The following variables specify system ISPF libraries:

   ISPLLIB = "'YOUR.ISP.SISLOAD'
   ISPMLIB = "'YOUR.ISP.SISPMENU'
   ISPSLIB = "'YOUR.ISP.SISPENU'
   ISPPLIB = "'YOUR.ISP.ISPPTENU'
   ISPTLIB = "'YOUR.ISP.ISPTENU'"
The following variables specify the SCF suffix and SRDF Host Component command prefix:

```
SCF_SUFFIX= "EMC"
CMD_PREFIX= "YOUR_HC_CMD_PREFIX"
```

The following variables set temporary dataset and work volumes:

```
UNIT = "SYSALLDA" Temporary datasets will go here
VOLUME = ' ' or you can assign them to this volser
UNITSW = "3390" Sort work datasets will go here
```

The following variables allow you to set SRDF/AR variables:

```
SAR_RACF_START = "SAR-START"
SAR_RACF_STOP = "SAR-STOP"
SAR_LOAD_LIBRARY = "DS_PREFIX.LINKLIB"
```

The following variables set the various product libraries:

```
TIMEFINDER_LIBRARY= "DS_PREFIX.LINKLIB"
```

4. Move a copy of the edited SARTOOL from the Mainframe Enablers SAMPLIB to the SCF.ISPELIB for execution.

5. To use each product’s ISPF panels, use TSO option 6 and type:

```
exec 'DS_PREFIX.ISPELIB(SARTOOL)'
```

**Result:** The Dell EMC SRDF/AR Monitor panel is displayed:

```
------------------------ Dell EMC SRDF/AR Monitor -------------------------
COPYRIGHT (C) 2001-2018 Dell Inc. ALL RIGHTS RESERVED. Vv.r.m
Enter a command option ===>
S  SRDF/AR Interface Functions

PF1: Help     PF3: Exit
```

**Figure 29** Dell EMC SRDF/AR Monitor panel

**Note:** Vv.r.m is the version of ResourcePak Base in use.
Customizing ISPF interface

If the following programs do not reside in an authorized library, complete the following steps to set up the ISPF interface environment:

◆ EMCTF (TimeFinder/Mirror)
◆ EMCTFA (SRDF/AR)

1. Update the IKJTSOxx member in the SYS1.PARMLIB by adding the following program names to AUTHPGM NAMES, AUTHTSF NAMES and AUTHCMD NAMES statements:
   - EMCTF
   - EMCTFA

2. For these changes to take effect, perform one of the following:
   - Use the "PARMLIB" TSO authorize command to dynamically change the IKJTSOxx active member without an IPL.
   - IPL the system.

   ```
   AUTHPGM NAMES( /* AUTHORIZED PROGRAMS */+
   EMCTF /* TF INTERFACE */+
   EMCTFA /* TF INTERFACE */) /* */
   AUTHTSF NAMES( /* PROGRAMS TO BE AUTHORIZED */+
   /* WHEN CALLED THROUGH THE */+
   /* TSO SERVICE FACILITY. */+
   EMCTF /* TF INTERFACE */+
   EMCTFA /* TF INTERFACE */) /* */
   ```

**Note:** It is recommended that you examine PARMLIB CHECK(xx) (where xx is the member name suffix) to ensure that there are no syntax errors.
Monitoring SRDF/AR environment

Starting ISPF interface session

**Note:** If you have not done so already, move a copy of the edited SARTOOL from SAMPLIB to SCF.ISPELIB for execution.

To start an ISPF interface session, take one of the following steps:

- From the ISPF Primary Option Menu, type the following command:
  
  ```
  TSO EX 'ds-prefix.ispelib(SARTOOL)'
  ```

- Use TSO option 6 and type:
  
  ```
  exec 'ds-scf-prefix.ISPELIB(SARTOOL)'
  ```

- Execute the REXX member SARTOOL placed in the SAMPLIB by typing "ex" in front of the member as shown in Figure 30.

![Figure 30 Executing SARTOOL REXX](image)

**Result:** The SRDF/AR Monitor panel appears.

Key controls

On any of these menus, or the panels they invoke, you can use function keys to control processing, as shown in Table 27.

<table>
<thead>
<tr>
<th>Key</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>PF1</td>
<td>Press on any field for help on that field.</td>
</tr>
<tr>
<td>Enter</td>
<td>Save information you entered on the current panel.</td>
</tr>
<tr>
<td>PF3</td>
<td>Exit from the current panel.</td>
</tr>
<tr>
<td>PF7</td>
<td>Scroll up on the current panel.</td>
</tr>
<tr>
<td>PF8</td>
<td>Scroll down on the current panel.</td>
</tr>
</tbody>
</table>
Setting session parameters

Type S at the command option prompt on the SRDF/AR Monitor panel.

Result: The Specify the Session Parameters panel shown in Figure 31 is displayed.

Figure 31 Specify the Session Parameters panel

The Specify the Session Parameters panel allows you to update the configuration parameters to fit your processing needs for the session.

Note: See the Mainframe Enablers Installation and Customization Guide to see if any default variables have been set. If so, the session parameter screens may reflect those values and look different from the screens shown here.

Take one of the following steps:

- If you do not want to make any changes, press PF3.

  Result: You see the SRDF/AR Function Controls panel is displayed, described in “Using SRDF/AR Function Controls” on page 111.

- If you do want to make changes, update the following fields as necessary.

  ◆ SCF Suffix
  The suffix value used to define the SCF to run against. This field is propagated to all setup panels. The default is EMC.

  ◆ Product Load Library
  The library where the SRDF/AR tool resides. Leave blank for the default.

  ◆ TimeFinder Load Library
  The library where the TimeFinder software resides.

  ◆ UNIT
  Specify the UNIT to be used for dataset allocations. The default is SYSALLDA.

  ◆ VOLUME
  Optionally, specify the VOLSER used for dataset allocations.

  ◆ RACF Start Process Resource Name
  The name given through RACF to the start process resource. The default is SAR-START.

  ◆ RACF Stop Process Resource Name
  The name given through RACF to the start process resource. The default is SAR-STOP.

  ◆ IBM Distribution Datasets
  The names of the IBM ISPF distribution datasets. The necessary ISPF libraries are ISPLLIB, ISPMLIB, ISPSLIB, ISPPLIB, and ISPTLIB.
- **Debug Mode**

Setting this value to Y enables the REXX/CLIST interactive trace facility.

---

**Note:** Dell EMC advises you not to set Debug Mode to Y unless you have been advised by your Dell EMC Customer Service representative.

When you finish, press **Enter** to save the changes. Then press **PF3**.

**Result:** The SRDF/AR Function Controls panel is displayed.

### Using SRDF/AR Function Controls

**Figure 32** shows the Dell EMC SRDF/AR Function Controls panel.

![Dell EMC SRDF/AR Function Controls panel](image)

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>EALL</td>
<td>List definition information</td>
</tr>
<tr>
<td>IALL</td>
<td>List invalid tracks</td>
</tr>
<tr>
<td>LALL</td>
<td>List locks</td>
</tr>
</tbody>
</table>

**Table 28** shows the primary commands you can use on the command line to display information for all processes.

---

**Table 28** SRDF/AR Process Information primary commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>EALL</td>
<td>List definition information</td>
</tr>
<tr>
<td>IALL</td>
<td>List invalid tracks</td>
</tr>
<tr>
<td>LALL</td>
<td>List locks</td>
</tr>
</tbody>
</table>

---
Table 28  SRDF/AR Process Information primary commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>RALL</td>
<td>Generate a report</td>
</tr>
<tr>
<td>SALL</td>
<td>List configuration information</td>
</tr>
<tr>
<td>UALL</td>
<td>Generate an extended report</td>
</tr>
<tr>
<td>YALL</td>
<td>List cycle information</td>
</tr>
</tbody>
</table>

Table 29 shows the line commands you can use to display process information.

Table 29  SRDF/AR Process Information line commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Information shown or action performed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cmdb</td>
<td>List commands used to define the process</td>
</tr>
<tr>
<td>Cnfg</td>
<td>List device configuration information</td>
</tr>
<tr>
<td>Cycl</td>
<td>List process cycle information</td>
</tr>
<tr>
<td>Global</td>
<td>List internal information</td>
</tr>
<tr>
<td>Invtrk</td>
<td>Displays the process invalid track information</td>
</tr>
<tr>
<td>Lock</td>
<td>Process Held Lock information</td>
</tr>
<tr>
<td>Pause</td>
<td>Pause a running process</td>
</tr>
<tr>
<td>Rept</td>
<td>Produce a report for a process</td>
</tr>
<tr>
<td>Rstrt</td>
<td>Restart a process; you are prompted for a start verification</td>
</tr>
<tr>
<td>Rptx</td>
<td>Produce an extended data report for a process. This option could produce extended output because each device and its attributes are displayed.</td>
</tr>
<tr>
<td>Sel</td>
<td>Process configuration information</td>
</tr>
<tr>
<td>Start</td>
<td>Start a process; you are prompted for a start verification</td>
</tr>
<tr>
<td>Stop</td>
<td>Stop a process; you are prompted for a stop option</td>
</tr>
<tr>
<td>Symt</td>
<td>List Symmetrix Table information for a process</td>
</tr>
</tbody>
</table>

◆ S - Set Session Options

Type S at the command option prompt to access Set Session Options. The panel allows you to modify the parameters being used for the session.
Generating batch reports

You can generate a batch SRDF/AR report for one or all SRDF/AR processes. You can specify only one report type on each request. Table 30 lists the JCL needed to run the SRDF/AR Monitor reports in batch mode. These can be found in SSPF SAMPLIB.

Table 30  JCL to run reports in batch mode

<table>
<thead>
<tr>
<th>JCL</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>SARJCL</td>
<td>Executes SAXSAR</td>
</tr>
<tr>
<td>SARJCL1</td>
<td>Executes SAAPIT</td>
</tr>
<tr>
<td>ESAJSTAR</td>
<td>Executes ESAXSTAR</td>
</tr>
</tbody>
</table>

The following SYSTSIN input code is used to generate the report:

ISPSTART CMD('ESAXAR report_options/process_name|ALL/'LIB CONTAINING SAR PGMS') NEWAPPL(SARB)

Table 31 lists valid process batch report options.

Table 31  Valid SRDF/AR process batch report options

<table>
<thead>
<tr>
<th>Option</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>REPT</td>
<td>Generate report</td>
</tr>
<tr>
<td>RPTX</td>
<td>Generate extended report</td>
</tr>
<tr>
<td>PARM</td>
<td>Display definition parameters</td>
</tr>
<tr>
<td>CMDB</td>
<td>Display command block</td>
</tr>
<tr>
<td>LOCK</td>
<td>Display lock information</td>
</tr>
<tr>
<td>CYCL</td>
<td>Display cycle information</td>
</tr>
<tr>
<td>SYMT</td>
<td>Display Symmetrix table information</td>
</tr>
<tr>
<td>CNFG</td>
<td>Display configuration</td>
</tr>
<tr>
<td>GLOBAL</td>
<td>Display global information</td>
</tr>
</tbody>
</table>

a. The ALL parameter is not valid with this option.
Monitoring SRDF/AR with SRDF/AR Monitor
CHAPTER 6
Command Reference

This chapter covers the following topics:

- Syntax conventions ................................................................. 116
- TF/Mirror commands ............................................................. 117
- SRDF/AR commands ............................................................. 173
Syntax conventions

The command syntax conventions are as follows:

- For easy reference, command keywords are supplemented by lowercase letters to form a meaningful word (for example, CONSistent). When typing a command, use only CAPITALIZED characters of any keyword.
- Variables are in lowercase and italics (for example, column-name). They represent user-supplied names or values in the syntax.
- Default values are indicated by underlining the value. For example, (Yes|No) means that “No” is the default value.
- Optional parameters are in square brackets [ ]. When specifying commands, optional parameters must be separated from the required syntax and from each other by commas. Optional parameters cannot have any embedded spaces.
- Alternative argument values are enclosed in “{}” and separated by “|”.
- If punctuation marks, parentheses, arithmetic operators, or other such symbols are shown, type them as part of the syntax.
TF/Mirror commands

Restrictions

A device range specified on a TF/Mirror command cannot span more than 4096 devices.

CONFIG

The CONFIG command allows you to do the following:

- Place or remove a device hold
- Remove the STD-to-BCV relationship for a device
- Make the BCV Ready or Not Ready.

Note: Failure of a CONFIG command results in a warning message (RC=4). This means that you can specify MAXRC(4) in the GLOBAL command if you are unsure of the state of the device.

Syntax

CONFIG seq#, cuup, action
CONFIG seq#, cuup-cuup, action
CONFIG seq#, cuus, action
CONFIG seq#, cuus-cuus, action
CONFIG seq#, LCL(cuu, symdv#bcv), action
CONFIG seq#, LCL(cuu, symdv#bcv-symdv#bcv), action
CONFIG seq#, LCL(cuu, symdv#std), action
CONFIG seq#, LCL(cuu, symdv#std-symdv#std), action
CONFIG seq#, RMT(cuu, symdv#bcv[, srdfgrp]), action
CONFIG seq#, RMT(cuu, symdv#bcv-symdv#bcv[, srdfgrp]), action
CONFIG seq#, RMT(cuu, symdv#std[, srdfgrp]), action
CONFIG seq#, RMT(cuu, symdv#std-symdv#std[, srdfgrp]), action

Parameters

action

The action performed using the CONFIG command. You can specify one of the following:

DELINC

Removes the BCV-to-STD relationship for the BCV or range.

IMPORTANT

Do not use the DELINC action if the synchronization has not been completed because all the source tracks may not have been copied to the BCV.
In the clone emulation mode, the SDDF session between the STD and BCV is deleted.

Note that after you execute a CONFIG DELINC command, you are not able to successfully execute a GLOBAL ConvertFullEstablish command for the BCVs.

If you are using the TF/Mirror clone emulation mode, TF/Mirror uses the MULTIDELINC option.

Under PowerMaxOS 5978, HYPERMAX OS 5977 and Enginuity 5876, you can specify a STD, instead of a BCV, to terminate all incomplete clone emulation sessions (session IDs 45/C5/F5) on the STD.

HOLD

Places an explicit device lock on the specified BCVs or STDs.

When you place a CONFIG HOLD on a device:

– If you attempt to execute an ESTABLISH, RE-ESTABLISH, RESTORE, or SPLIT command on the device, you receive an error (BCVM146E).

– The STD with which the BCV is paired is not indicated in QUERY reports (see example 9 in “Examples” on page 120).

MULTIDELINC

Removes the BCV-to-STD relationship for the range of BCVs.

IMPORTANT

Do not use the MULTIDELINC action if the synchronization has not been completed because all the source tracks may not have been copied to the BCV.

After you execute a CONFIG MULTIDELINC command, you are not able to successfully execute a GLOBAL ConvertFullEstablish command for the BCVs.

TF/Mirror uses a single syscall to process the requests together in a multidevice list.

You can specify a range of STDs (instead of BCVs) to terminate all incomplete clone emulation sessions (session IDs 45/C5/F5) on the STDs.

NR

Makes the BCV Not Ready to the host.

READY

Makes the BCV Ready to the host.

If you issue a QUERY command immediately after this action, the status of the BCV may show AVAILB for up to five seconds.

Note: “QUERY” on page 141 describes the QUERY command.
Releases device locks, including the CONFIG HOLD device lock and the TF/Clone Mainframe Snap Facility hold.

**Note:** The RELEASE parameter helps overcome interoperability problems of the automatic hold placed by TF/Clone Mainframe Snap Facility and the explicit hold placed using the CONFIG HOLD command of TF/Mirror.

**cuu**

A device identified with its CUU.

For a RMT operation, if you do not specify the SRDF group (srdgrp), then cuu must be an R1 device.

**cuup**

An STD identified with its CUU.

**cuup-cuup**

A range of STDs identified with their CUUs. All devices in the range must reside in the same storage system.

**cuus**

A BCV identified with its CUU.

**cuus-cuus**

A range of BCVs identified with their CUUs. All devices in the range must reside in the same storage system.

**LCL**

Specifies that the devices are on the local side of an SRDF configuration.

**RMT**

Specifies that the devices are on the remote side of an SRDF configuration.

**seq#**

A 1-to-128 sequence number that defines the command execution order. Commands with equal sequence numbers are executed in parallel.

**srdgrp**

The SRDF group or the hop list for the RMT operation.

Each SRDF group is represented by a one- or two-digit value. You can specify a hop list of up to four SRDF groups separated by periods, for example:

```
nn.nnn.nnn
```

**symdv#bcv**

A BCV identified with its PowerMax/VMAX device number.

**symdv#bcv-symdv#bcv**

A range of BCVs identified with their PowerMax/VMAX device numbers.

**symdv#std**

An STD identified with its PowerMax/VMAX device number.
symdv#std-symdv#std

A range of STDs identified with their PowerMax/VMAX device numbers.

Examples

1. To place a hold on the BCV identified with CUU DE18:

   CONFIG 2,DE18,HOLD

   BCVI020I Start of INPUT control statement(s) from SYSIN
   BCVI018I (0002) CONFIG 2,DE18,HOLD
   BCVI021I End of INPUT control statement(s) from SYSIN

   BCVM039I (0002) Process input statement
   BCVM004I HOLD BCV DE18

2. To remove a hold from the BCV identified with CUU DE18:

   CONFIG 2,DE18,RELEASE

   Start of INPUT control statement(s) from SYSIN
   (0002) CONFIG 2,DE18,RELEASE

   End of INPUT control statement(s) from SYSIN

   BCVM039I (0002) Process input statement
   BCVM004I RELEASE BCV DE18

3. To set the BCV identified with CUU DE18 to Not Ready:

   CONFIG 2,DE18,NR

   BCVI020I Start of INPUT control statement(s) from SYSIN
   BCVI018I (0002) CONFIG 2,DE18,NR
   BCVI021I End of INPUT control statement(s) from SYSIN

   BCVM039I (0002) Process input statement
   BCVM004I NOT-READY BCV DE18

4. To set the BCV identified with CUU DE18 to Ready:

   CONFIG 2,DE18,READY

   BCVI020I Start of INPUT control statement(s) from SYSIN
   BCVI018I (0002) CONFIG 2,DE18,READY
   BCVI021I End of INPUT control statement(s) from SYSIN

   BCVM039I (0002) Process input statement
   BCVM004I READY BCV DE18

5. To set the remote BCV identified with PowerMax/VMAX device number 170 to Not Ready through CUU 4D10 and SRDF group 05:

   CONFIG 20,RMT(4D10,170,05),NR

   BCVI020I Start of INPUT control statement(s) from SYSIN
   BCVI018I (0002) CONFIG 20,RMT(4D10,170,05),NR
   BCVI021I End of INPUT control statement(s) from SYSIN

   BCVM039I (0002) Process input statement
   BCVM004I NOT-READY REMOTE BCV SYMDEV 000170 through 4D10

6. To set remote BCVs identified with PowerMax/VMAX device numbers 170 -173 to Not Ready through CUU 4D10 and SRDF group 05:
**CONFIG 20,RMT(4D10,170-173),NR**

BCVI020I Start of INPUT control statement(s) from SYSIN
BCVI018I (0002) CONFIG 20,RMT(4D10,170-173),NR
BCVI021I End of INPUT control statement(s) from SYSIN

BCVM039I (0002) Process input statement
BCVM004I NOT-READY REMOTE BCV SYMDEV 000170 through 4D10
BCVM080I ACCESS ALLOWED
BCVM069I Security Exit allowed the bypassing of the online state check

BCVI018I (0002) CONFIG 20,RMT(4D10,170-173),NR
BCVI021I End of INPUT control statement(s) from SYSIN

BCVM039I (0002) Process input statement
BCVM004I NOT-READY REMOTE BCV SYMDEV 000171 through 4D10
BCVM069I Security Exit allowed the bypassing of the online state check

BCVM039I (0002) Process input statement
BCVM004I NOT-READY REMOTE BCV SYMDEV 000172 through 4D10
BCVM069I Security Exit allowed the bypassing of the online state check

BCVM039I (0002) Process input statement
BCVM004I NOT-READY REMOTE BCV SYMDEV 000173 through 4D10
BCVM069I Security Exit allowed the bypassing of the online state check

BCVM039I (0003) Process input statement

7. To remove the STD-to-BCV relationship on devices identified with PowerMax/VMAX device numbers EF8-EF9 through CUU 9A10 and SRDF groups 50, 13:

**CONFIG 64,RMT(9A10,EF8-EF9,50.13),DELINC**

BCVI020I Start of INPUT control statement(s) from SYSIN
BCVI018I (0002) CONFIG 64,RMT(9A10,EF8-EF9,50.13),DELINC
BCVI021I End of INPUT control statement(s) from SYSIN

BCVM039I (0002) Process input statement
BCVM004I DELINC REMOTE BCV SYMDEV 000EF8 through 9A10

BCVM039I (0002) Process input statement
BCVM004I DELINC REMOTE BCV SYMDEV 000EF9 through 9A10

BCVM039I (0002) Process input statement
BCVM004I DELINC REMOTE BCV SYMDEV 000EF8 through 9A10

BCVM039I (0002) Process input statement
BCVM004I DELINC REMOTE BCV SYMDEV 000EF9 through 9A10

8. To place a hold on devices identified with PowerMax/VMAX device numbers 1FD-1FF in the PowerMax/VMAX system that contains CUU DE20:

**CONFIG 13,LCL(DE20,1FD-1FF),HOLD**

BCVI020I Start of INPUT control statement(s) from SYSIN
BCVI018I (0002) CONFIG 13,LCL(DE20,1FD-1FF),HOLD
BCVI021I End of INPUT control statement(s) from SYSIN

BCVM039I (0002) Process input statement
BCVM004I HOLD LOCAL BCV SYMDEV 0001FD through DE20
BCVM039I (0002) Process input statement
BCVM004I HOLD LOCAL BCV SYMDEV 0001FE through DE20
BCVM039I (0002) Process input statement
BCVM004I HOLD LOCAL BCV SYMDEV 0001FF through DE20

9. In the following example, after a CONFIG(HOLD) is placed on the device, the STD with which the BCV is paired no longer appears in the QUERY report:

<table>
<thead>
<tr>
<th>CUU</th>
<th>SYM#</th>
<th>CUU</th>
<th>SYM#</th>
<th>ITRK-BCV</th>
<th>ITRK-STD</th>
<th>STATUS</th>
<th>USED</th>
<th>BCV</th>
<th>EMUL</th>
<th>#CYLS</th>
<th>TYPE</th>
<th>SYNC</th>
<th>MODE</th>
</tr>
</thead>
<tbody>
<tr>
<td>0004B0</td>
<td>0004B0</td>
<td>0</td>
<td>0</td>
<td>HOLD-U</td>
<td>000480</td>
<td>3390</td>
<td>1113</td>
<td>R1</td>
<td>YES</td>
<td>CLONE</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0004B1</td>
<td>0004B1</td>
<td>0</td>
<td>0</td>
<td>HOLD-U</td>
<td>000481</td>
<td>3390</td>
<td>1113</td>
<td>R1</td>
<td>YES</td>
<td>CLONE</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Note: The status of “HOLD-U” indicates a user hold, which means that the user issued a CONFIG HOLD command against the device. The status of “HOLD-T” indicates that the BCV is the target of a SNAP.
ESTABLISH

The ESTABLISH command establishes an STD/BCV pair and copies the contents of the STD to the BCV. The BCV becomes Not Ready to the host.

Note the following when specifying devices for the ESTABLISH command:

- You must specify both STDs and BCVs or ranges.

  **Note:** When you specify device ranges, STDs and BCVs are paired together, and the storage system processes them sequentially.

- The STDs and BCVs must have the same device type and size.

  **Note:** CKD STD volumes can only be established to CKD BCV volumes of the same size or larger. FBA STD volumes can only be established to FBA volumes of the same size.

If the BCV was previously split, all tracking of the changed data for the previous STD/BCV pair is discarded.

For more information about the ESTABLISH functionality, see “Synchronizing from STD to BCV: ESTABLISH” on page 37.

Syntax

```
ESTABLISH_sequence #, cuu,cuu [ ,optional_parameters ]
ESTABLISH_sequence #, cuus-cuus, cuup-cuup [ ,optional_parameters ]
ESTABLISH_sequence #, LCL( cuu, symdv#bcv, symdv#std ) [ ,optional_parameters ]
ESTABLISH_sequence #, LCL( cuu, symdv#bcv-symdv#bcv, symdv#std-symdv#std ) [ ,optional_parameters ]
ESTABLISH_sequence #, RMT( cuu, symdv#bcv, symdv#std[ ,srdfgrp ] ) [ ,optional_parameters ]
ESTABLISH_sequence #, RMT( cuu, symdv#bcv-symdv#bcv, symdv#std-symdv#std[ ,srdfgrp ] ) [ ,optional_parameters ]
```

**Where optional_parameters are:**

- [ ,CBCV({Yes |No}) ]
- [ ,CLONE_emulation({Yes |No}) ]
- [ ,MULTIATTach({Yes |No}) ]
- [ ,PROTECTEDBCVEstablish({Yes |No}) ]


\[ , \text{VXWAIT(Yes|No)} \]
\[ , \{ \text{WAIT(} (n\text{nnn}) \mid \text{NOWAIT}) \} \]

**Parameters**

**CBCV((Yes|No))**

**IMPORTANT**

The CBCV parameter is ignored in the clone emulation mode.

Allows to use the BCV as a Concurrent BCV for the STD. The default value is No.

**Note:** “Concurrent BCV” on page 40 describes the Concurrent BCV feature.

A Concurrent BCV ESTABLISH is not allowed if the first pair was attached with a RESTORE command.

If mirror positions are available, you can define a maximum of two BCVs for an STD.

When splitting a BCV that was ESTABLISHed with the CBCV(Y) parameter, INS(Y) must be specified in the SPLIT command to split all devices in the range in one I/O operation.

**CLONEemulation((Yes|No))**

**IMPORTANT**

Under PowerMax OS 5978, HYPERMAX OS 5977 and Enginuity 5876, the CLONEemulation parameter is ignored. All operations are processed in the clone emulation mode.

Enables or disables the clone emulation mode. The default value is No.

**Note:** “Clone emulation” on page 52 describes the clone emulation mode.

Under Enginuity 5773:

- If the BCV is a RAID5- or RAID6-protected device, TF/Mirror uses clone emulation regardless of the value of the CLONEemulation parameter.

- If you set CLONEemulation to Yes and have set the GLOBAL CLONEemulation parameter to REQUEST:
  - TF/Mirror processes the BCV specified in the ESTABLISH statement in the clone emulation mode regardless of the RAID protection type.
  - TF/Mirror uses clone emulation for any RE-ESTABLISH, Incremental RESTORE, or SPLIT command that is executed in the clone emulation mode.

- If you set CLONEemulation to No or do not specify the CLONEemulation parameter, TF/Mirror does not use clone emulation for the ESTABLISH command or any related command if the BCV is a non-RAID5- or RAID6 protected device.
cuu
A device identified with its CUU.
For a RMT operation, if you do not specify the SRDF group (srdfgrp), then cuu must be an R1 device.

cuup
An STD identified with its CUU.

cuup-cuup
A range of STDs identified with their CUUs. All devices in the range must reside in the same storage system. The number of devices in cuus-cuus and cuup-cuup ranges must be the same.

cuus
A BCV identified with its CUU.

cuus-cuus
A range of BCVs identified with their CUUs. All devices in the range must reside in the same storage system. The number of devices in cuus-cuus and cuup-cuup ranges must be the same.

MULTIATTach({Yes | No})
Enables or disables the Multi-Attach feature for this ESTABLISH command:
- Yes—Enables the Multi-Attach feature, unless the GLOBAL MULTIATTach parameter is set to NONE.
- No—Disables the Multi-Attach feature regardless of the GLOBAL MULTIATTach parameter value.
The default value is Yes.

Note: “Multi-attach” on page 56 describes the Multi-Attach feature.

Aliases for MULTIATTach include: MULTA.

PROTECTEDBCVEstablish({Yes | No})

IMPORTANT
The PROTECTEDBCVEstablish parameter is ignored in the clone emulation mode. All operations are processed using the Protected BCV feature.

Enables or disables the Protected BCV feature. The default value is No.

Note: “Protected BCV” on page 41 describes the Protected BCV feature.

Aliases for PROTECTEDBCVEstablish include: PBCVE, PROTECTEDBCVE, PROTBCVE.

LCL
Specifies that the devices are on the local side of an SRDF configuration.
RMT
   Specifies that the devices are on the remote side of an SRDF configuration.

seq#
   A 1-to-128 sequence number that defines the command execution order. Commands with equal sequence numbers are executed in parallel.

srdfgrp
   The SRDF group or the hop list for the RMT operation.
   Each SRDF group is represented by a one- or two-digit value. You can specify a hop list of up to four SRDF groups separated by periods, for example:

   nn.nn.nn.nn

symdv#bcv
   A BCV identified with its PowerMax/VMAX device number.

symdv#bcv—symdv#bcv
   A range of BCVs identified with their PowerMax/VMAX device numbers.

symdv#std
   An STD identified with its PowerMax/VMAX device number.

symdv#std—symdv#std
   A range of STDs identified with their PowerMax/VMAX device numbers.

VXWAIT (Yes|No)

   **Note:** The VXWAIT parameter is supported with PowerMaxOS 5978 and HYPERMAX OS 5977. Under Enginuity 5876 and 5773, the VXWAIT parameter is ignored.

   The VXWAIT parameter fine-tunes TF/Mirror waiting behavior under PowerMaxOS 5978 and HYPERMAX OS 5977 by providing the possibility to override the WAIT|NOWAIT value set for the ESTABLISH/RE-ESTABLISH/RESTORE commands and thus to allow a SPLIT with invalid tracks. The benefit is that the BCV with point-in-time copy of data is available to the host without waiting for tracks to be copied.

   With PowerMaxOS 5978 and HYPERMAX OS 5977, TF/Mirror waits for the action to complete only when the WAIT|NOWAIT parameter is set to WAIT and the VXWAIT parameter is set to Yes. In all other cases, TF/Mirror does not wait for completion.

   The VXWAIT parameter can be used to override the settings made with the GLOBAL command. The default value for this parameter can be predefined using site options.
WAIT\((n\text{nnn})\) | NOWAIT

Determines whether to wait for the action to complete:

- **WAIT**—Wait for the action to complete.
- **NOWAIT**—Pass the command to the storage system and assume that the action is complete.
- **nnnn**—Wait for nnnn minutes and assume that the action is complete. For example, WAIT(10) means a 10-minute wait. A maximum of up to 9999 minutes can be specified.

**Note:** If WaitSync is specified, the time \((nnnn)\) also includes the waiting time allocated to synchronize the BCV mirror.

**IMPORTANT**

With PowerMaxOS 5978 and HYPERMAX OS 5977, TF/Mirror waits for the action to complete only when the WAIT|NOWAIT parameter is set to WAIT and the VXWAIT parameter is set to Yes. In all other cases, TF/Mirror does not wait for completion.

This parameter can be used to override the settings made with the GLOBAL command. The default value for this parameter can be predefined using site options.

**Examples**

1. To copy the contents of CUU C438 to CUU C518:

   **ESTABLISH 111,C518,C438**

   BCVI020I Start of INPUT control statement(s) from SYSIN
   BCVI018I (0001) ESTABLISH 111,C518,C438
   BCVI021I End of INPUT control statement(s) from SYSIN

   BCVM039I (0001) Process input statement
   BCVM004I ESTABLISH STANDARD device C438 to BCV C518
   BCVM080I ACCESS ALLOWED
   BCVM069I Security Exit allowed the bypassing of the online state check

2. To copy the contents of CUUs DE84-DE87 to CUUs DE84-DE87, where BCVs are RAID5, and clone emulation is used:

   **ESTABLISH 111,DP68-DF6B,DE84-DE87**

   BCVI020I Start of INPUT control statement(s) from SYSIN
   BCVI018I (0001) ESTABLISH 111,DP68-DF6B,DE84-DE87
   BCVI021I End of INPUT control statement(s) from SYSIN

   BCVM039I (0001) Process input statement
   BCVM004I ESTABLISH STANDARD device DE84 to BCV DF68
   BCVM080I ACCESS ALLOWED
   BCVM069I Security Exit allowed the bypassing of the online state check
   BCVM140I Command processed via TF/Clone emulation

   BCVM039I (0001) Process input statement
   BCVM004I ESTABLISH STANDARD device DE85 to BCV DF69
   BCVM069I Security Exit allowed the bypassing of the online state check
   BCVM140I Command processed via TF/Clone emulation
3. To copy the contents of the remote device identified with PowerMax/VMAX device number CD to the device identified with the PowerMax/VMAX device number 1FD through CUU DE20, SRDF group E0:

\texttt{ESTABLISH 1,RMT(DE20,01FD,00CD,E0)}

4. To copy the contents of remote devices identified with PowerMax/VMAX device numbers 00CD-00CF to devices identified with PowerMax/VMAX device numbers 01FD-01FF through CUU DE20, SRDF group E0:

\texttt{ESTABLISH 1,RMT(DE20,01FD-1FF,00CD-00CF,E0)}
5. To copy the contents of the device identified with PowerMax/VMAX device number 3D to the devices identified with PowerMax/VMAX device numbers 16A and 16D in the storage system identified with CUU 9A10:

```
BCVI020I Start of INPUT control statement(s) from SYSIN
BCVI018I (0001) ESTABLISH 3,LCL(9A10,16A,03D)
BCVI018I (0002) ESTABLISH 6,LCL(9A10,16D,03D),CBCV(Y)
BCVI021I End of INPUT control statement(s) from SYSIN
```

```
BCVM039I (0001) Process input statement
BCVM004I ESTABLISH LOCAL STANDARD SYMDEV 00003D to BCV SYMDEV 00016A through 9A10
BCVM080I ACCESS ALLOWED
BCVM069I Security Exit allowed the bypassing of the online state check
BCVM039I (0002) Process input statement
BCVM004I ESTABLISH LOCAL STANDARD SYMDEV 00003D to BCV SYMDEV 00016D through 9A10
BCVM069I Security Exit allowed the bypassing of the online state check
```

6. To perform a multi-attach ESTABLISH:

**Note:** In BCVM004I, the BCVs and STDs are separated by a slash (/).

```
BCVI020I Start of INPUT control statement(s) from SYSIN
BCVI018I (0001) GLOBAL MAXRC(8,SETMAX),WAIT,MULTATT(ALL)
BCVI018I (0002) ESTABLISH 10,D02C,CF0C,MULTIATT(Y)
BCVI018I (0003) ESTABLISH 10,D02D,CF0D,MULTIATT(Y)
BCVI018I (0004) ESTABLISH 10,LCL(D02F,15E,00E),MULTIATT(Y)
BCVI018I (0005) ESTABLISH 10,LCL(D02F,15F,00F),MULTIATT(Y)
BCVI018I (0006) ESTABLISH 10,RMT(9401,23C,1B8,07),MULTIATT(Y)
BCVI018I (0007) ESTABLISH 10,RMT(9401,23D,1B9,07),MULTIATT(Y)
BCVM039I (0002) Process input statement
BCVM004I MULTI ESTABLISH devices:
BCVM004I * D02C/CF0C(0002),D02D/CF0D(0003)
BCVM039I (0004) Process input statement
BCVM004I MULTI ESTABLISH LOCAL devices through D02F:
BCVM004I * 00015E/00000E(0004),00015F/00000F(0005)
BCVM039I (0006) Process input statement
BCVM004I MULTI ESTABLISH REMOTE devices through 9401:
BCVM004I * 00023C/0001B8(0006),00023D/0001B9(0007)
```

7. To perform a multi-attach, multihop ESTABLISH operation in the clone emulation mode, where the command is issued remotely through CUU 9A10, through SRDF group 50, and then SRDF group 13, to the devices identified with PowerMax/VMAX device numbers EF8-EF9:

**Note:** In BCVM004I, the BCVs and STDs are separated by a slash (/).

```
BCVI018I (0001) GLOBAL MAXRC(8,SETMAX),MULTA(ALL),CLONE(REQUEST)
BCVI018I (0002) ESTABLISH 2,RMT(9A10,EF8-EF9A,50.13),CLONE(Y)
BCVI021I End of INPUT control statement(s) from SYSIN
BCVM039I (0002) Process input statement
BCVM080I ACCESS ALLOWED
BCVM069I Security Exit allowed the bypassing of the online state check
BCVM004I MULTI ESTABLISH REMOTE devices through 9A10:
BCVM004I * 000EF8-000EFA/000548-00054A(0002)
BCVM140I Command processed via TF/Clone emulation
GLOBAL

The GLOBAL command sets default values for all TF/Mirror commands.

Only one GLOBAL command is allowed. If you use a GLOBAL command, you must specify at least one optional parameter.

**Note:** If you are executing a TimeFinder command from the SRDF Host Component, GLOBAL commands and parameter values set from the SRDF Host Component are not transferred to TF/Mirror. Therefore only default values of the TF/Mirror GLOBAL command parameters are used during execution.

Syntax

GLOBAL

[, ATTach[ (nn, wait_int)]]
[, AUTORElease({Yes|No})]
[, BYPASSPARSEERROR({Yes|No})]
[, CLONEemulation({RAID5BCV|ALLBCV|REQUEST})]
[, CONSistent([MODE({IOSLEVEL|ECA})][, TIMEOUT(nn)]
[,{SYNConly|ALLOWNONSYNC}][, ECACLEAR({SEQLEVEL|CONTROLLER})])]
[, CONVERTFullEstablish({Yes|No})]
[, DEBUG([{CLONE|ECA|FBA}[, SITEOptions[{ALL}][, SRDFA]})]
[, FASTESTablish({Yes|No})]
[, FBAMETA({ALLOWINCOMPLETE|COMPLETEonly})]
[, LCRL1BCV]
[, MAXGrp(nnnn)]
[, MAXRC(n[,{SETMAX|SETZERO}])]
[, MULTBCV({NEW|OLD|REJ})]
[, MULTIATTach({NONE|ALL|EST|REEST})]
[, NOCUU]
[, NOTYPE]
[, PARallel({Yes[,n]|No})]
[, PROTECTEDRESTORE({REQUEST|CLONE|ALLBCV})]
[, TerminateExistingRelationship({Yes|No})]
[, TolerateDesiredState[({NONE|ANY|[|SPLIT|[|RESTORE}])]
[, VXWAIT({Yes|No})]
[,{WAIT|NOWAIT}]
Parameters

**ATTach\[ (nn, wait\_int) \]**

Processes multi-attach requests on ESTABLISH or RESTORE commands before querying the storage system to ensure that the BCVs are attached.

*Note:* “Multi-attach” on page 56 describes the Multi-Attach feature.

**nn**

The number of requests to issue before querying the storage system for status. Valid values are from zero (0) to 64. The default value is 8.

**wait\_int**

The number of seconds to wait before querying for status. Valid values are from zero (0) to 60. The default value is 1.

If you specify ATTach without parameters, TF/Mirror uses the default values.

The default values for this parameter can be predefined using site options.

**AUTORElease\[\{Yes|No\}\]**

*IMPORTANT*

AUTORElease is not required in the clone emulation mode.

Automatically releases a hold on the STD that caused an ESTABLISH or RE-ESTABLISH command to fail and retries the ESTABLISH or RE-ESTABLISH command.

The default is No for TF/Mirror and Yes for SRDF/AR. The default value for this parameter can be predefined using site options.

Under Enginuity 5773, AUTORElease is required for non-clone devices.

**BYPASSPARSEERROR\[\{Yes|No\}\]**

Allows you to ignore parsing errors. The default value is No.

*Note:* BYPASSPARSEERROR requires the MAXRC parameter of the GLOBAL command to be set to 8 or higher.

Aliases for BYPASSPARSEERROR include: BYPPARSEERR, BYPPERROR, BYPPERR.

The default value for this parameter can be predefined using site options.

**CLONEemulation\[\{RAID5BCV|ALLBCV|REQUEST\}\]**

*IMPORTANT*

Under PowerMax OS 5978, HYPERMAX OS 5977 and Enginuity 5876, the CLONEemulation parameter is ignored. All operations are processed in the clone emulation mode.
Sets the clone emulation mode.

**Note:** “Clone emulation” on page 52 describes the clone emulation mode.

Under Enginuity 5773:

**RAID5BCV**

(Default) Only RAID5- or RAID6-protected BCVs are processed in the clone emulation mode.

**ALLBCV**

The clone emulation mode is used for all BCVs, as well as for the following:

- Any ESTABLISH or full RESTORE command
- Any Incremental RESTORE or RE-ESTABLISH or any SPLIT command that is part of a clone emulation session established by a prior related ESTABLISH or full RESTORE

**REQUEST**

The clone emulation mode is used for non-RAID5- and RAID6-protected BCVs, as well as for the following:

- Any operations where initial ESTABLISH or full RESTORE commands contain the CLONE emulation(Y) parameter.
- Any Incremental RESTORE or RE-ESTABLISH or any SPLIT command that is part of a clone emulation session established by a prior related ESTABLISH or full RESTORE.

The default value for this parameter can be predefined using site options.

**CONSistent**

The default values for this parameter can be predefined using site options.

**MODE**

Sets the I/O control mode to ensure consistency for SPLIT operations. The default value is ECA, which is also used when you do not specify MODE.

**TIMEOUT**

Specifies the maximum time (in seconds) that TF/Mirror holds host I/O to process a Consistent SPLIT. The default value is 15. The maximum value is 60.
I/O is released after completion of the Consistent SPLIT or after this timeout, whichever occurs first. A BCVM088W message is issued if the I/O is released before the Consistent SPLIT is completed.

**Note:** “Consistent SPLIT” on page 43 describes the Consistent SPLIT feature.

When the TIMEOUT parameter is specified both in the GLOBAL and in the SPLIT statements, the most restrictive (lowest) value is used.

For example, if the GLOBAL TIMEOUT is set to 15 and the SPLIT TIMEOUT is set to 30, TF/Mirror uses the value of 15. If the GLOBAL TIMEOUT is set to 30 and the SPLIT TIMEOUT is set to 22, TF/Mirror uses the value of 22.

**SYNConly|ALLOWNONSYNC**

Allows (ALLOWNONSYNC) or prohibits (SYNConly, default) a Consistent SPLIT to continue, even though the STD in the split has been identified as an R2 or R21 device in an SRDF relationship with an R1 device that is in the adaptive copy mode.

**Note:** “Consistent SPLIT” on page 43 describes the Consistent SPLIT feature.

Keep in mind that R21 devices must be defined as adaptive copy devices. If you are using an R21 device as an STD in a Consistent SPLIT, you have to specify the ALLOWNONSYNC parameter in the GLOBAL command.

**ECACLEAR({SEQLEVEL|CONTROLLER})**

Sets storage system-level ECA clearing:

- If SEQLEVEL (default) is used, ECA window clearing is deferred until all of the Consistent SPLIT requests at the current sequence level are processed.
- If CONTROLLER is used, TF/Mirror clears ECA for a storage system for an RMT Consistent SPLIT when polling has completed for all BCVs in that storage system.

**Note:** The Consistency Groups for z/OS Product Guide provides more information about storage system-level ECA clearing.

The alias for SEQLEVEL is SEQLVL. The alias for CONTROLLER is CNTRL.

**CONVertFullEstablish({Yes|No})**

**IMPORTANT**

Under PowerMax OS 5978, HYPERMAX OS 5977 and Enginuity 5876, the ConvertFullEstablish parameter is ignored. All operations are processed in the clone emulation mode.

Converts an incremental RE-ESTABLISH to a full ESTABLISH when there is no valid relationship between the STDs and BCVs specified in the RE-ESTABLISH command. The conversion is done using the last known STD.

The default value is No. With the default value, the RE-ESTABLISH command fails.

The default value for this parameter can be predefined using site options.
DEBUG([([CLONE][,ECA][,FBA][,SITEOptions([ALL])][,SRDFA])]

Enables the debug mode.

⚠️ **CAUTION**

Use DEBUG only when directed by Dell EMC Customer Support.

---

CLONE

Provides details for operations processed through the clone emulation mode. This parameter is optional.

---

**Note:** The CLONE option requires the EMCQCAPI DD statement in the TF/Mirror started task JCL, as described in “EMCQCAPI” on page 22.

---

ECA

Provides details while processing an operation where ECA may be active. This parameter is optional.

---

FBA

Issues the BCVI089I message for each FBA device in a Consistent SPLIT. This parameter is optional.

---

SITEOptions([ALL])

Displays the Site Options - TF/Mirror report described in “Viewing site options and overrides” on page 28. Specify SITEOptions without subparameters for basic format and SITEOptions(ALL) for extended format of the report.

---

SRDFA

Provides details while processing a Consistent SPLIT where SRDF/A is active. This parameter is optional.

---

FASTESTablish(Yes|No)

Converts a full ESTABLISH to an incremental RE-ESTABLISH when a valid relationship exists between the STDs and BCVs. The default value is No.

The default value for this parameter can be predefined using site options.

---

FBAMETA({ALLOWINCOMPLETE|COMPLETEonly})

Allows (ALLOWINCOMPLETE) or disallows (COMPLETEonly, default) to specify a partial list of FBA meta group members in a TF/Mirror command, instead of requiring all members of the group to be specified.

The default value for this parameter can be predefined using site options.

---

LCLR1BCV

**IMPORTANT**

This parameter is used for SRDF/AR only.

Indicates that the R1 BCVs are not mounted to the host and are identified with their PowerMax/VMAX device numbers.
**MAXGrp** (nnnn)

**IMPORTANT**
This parameter is used for SRDF/AR only.

The maximum number of SRDF/AR groups. Valid values of nnnn are from 1 to 9999. The default value is 64.

The default value for this parameter can be predefined using site options.

**MAXRC** (n[ , {SETMAX | SETZERO} ])

The maximum allowed return code of a job step to continue processing. The default value is 0 (zero).

**Note:** The default value for this parameter can be predefined using site options.

n is the maximum allowed return value.

- When n is 0 (zero), TF/Mirror treats any message as an error and returns the highest return code encountered as the job step return code. The returned message is an error.

- When n is from 1 to 8:
  - If you receive a return code that is not higher than n, the job step return code is set to 0. TF/Mirror treats any message as a warning and continues processing.
  - If the return code is higher than n, the job step return code is set to that value, and TF/Mirror treats any message as an error and stops processing.

If you receive a return code of 12, that value is greater than any possible MAXRC value, TF/Mirror treats the message as a serious error and stops processing.

SETMAX returns the highest return code observed during execution as the job step return code. If that return code is higher than the value of n, the message returned is an error.

When SETZERO is specified (which is equal to not specifying SETMAX), if the highest internal reason code does not exceed n, then TF/Mirror sets the job step reason code to zero (0).

If you do not specify MAXRC, TF/Mirror always returns the highest return code encountered as the job step return code. The returned message is an error.

Note the following when using MAXRC=8:

- Any action statement that has a return code of 8 or less causes TF/Mirror to set a job step return code of 0.

- If the job scheduler is testing job step return codes to ensure that any process completed before another process is started, use the code MAXRC=7 or less for action statements.

- A return code of 8 is issued for all TF/Mirror error messages that have suffix E. Suffix E indicates that a process did not complete as expected.
You can also use the MAXRC parameter with the TolerateDesiredState parameter of the GLOBAL command to determine whether the current state of devices is taken into consideration during an ESTABLISH, RE-ESTABLISH, RESTORE, or SPLIT operation. If you set TolerateDesiredState and set MAXRC to a value of four (4) or higher, TF/Mirror ignores the current state of any specified devices during ESTABLISH, RE-ESTABLISH, RESTORE, or SPLIT command execution.

MULTBCV ( {NEW | OLD | REJ } )

The action taken when the maximum number of multiple BCVs for an STD is exceeded during processing of an ESTABLISH, RE-ESTABLISH, or RESTORE command.

**Note:** “Multi-BCV” on page 39 describes the Multi-BCV feature.

**NEW**
Delete the most recent incremental relationship.

**OLD**
( Default ) Delete the oldest incremental relationship.

**REJ**
Reject the command.

The default value for this parameter can be predefined using site options.

MULTIATTach ( {NONE | ALL | EST | REEST} )

Determines operations for which the Multi-Attach feature is enabled.

**Note:** “Multi-attach” on page 56 describes the Multi-Attach feature.

**NONE**
Disables the Multi-Attach feature for all attach operations, regardless of whether it is enabled in individual ESTABLISH or RE-ESTABLISH commands.

**ALL**
Enables the Multi-Attach feature for all attach operations. You can disable it in individual ESTABLISH or RE-ESTABLISH commands.

**EST**
Enables the Multi-Attach feature for all ESTABLISH commands. You can disable it in individual ESTABLISH commands.

**REEST**
Enables the Multi-Attach feature for all RE-ESTABLISH commands. You can disable it in individual RE-ESTABLISH commands.

If you specify no value for the GLOBAL MULTIATTach parameter, you can enable the Multi-Attach feature using the MULTIATTach parameter in individual ESTABLISH and RE-ESTABLISH commands.

The default value for this parameter can be predefined using site options.

**NOCUU**
Disables the CUU-BCV search when a QUERY command is executed.
NOTYPE

Disables the type check in a QUERY command to help reduce the processing time.

PARallel({Yes, n | No})

Enables or disables multitasking.

**Note:** “Multitasking” on page 56 describes multitasking.

**No**

(Defaul) Multitasking is disabled.

**Yes**

Multitasking is enabled.

n

The maximum number of subtasks for multitasking (specified with Yes). Valid values are from 1 to 256. If you do not specify a number when you choose Yes, TimeFinder uses the default value of 16.

PROTECTEDRESTORE( {REQUEST | CLONE | ALLBCV})

**Important**

The PROTECTEDRESTORE parameter is ignored in the clone emulation mode. All RESTORE operations are processed using the Protected RESTORE feature described in “Protected RESTORE” on page 49. This means that PROTECTEDRESTORE(Y) is not required in the SPLIT command after a RESTORE operation performed in the clone emulation mode.

Determines whether the PROTECTEDRESTORE(Y) parameter must be specified in a SPLIT command issued after a Protected RESTORE.

**REQUEST**

(Defaul) PROTECTEDRESTORE(Y) is required in the SPLIT command.

**CLONE**

PROTECTEDRESTORE(Y) is not required in the SPLIT command after a RESTORE operation performed in the clone emulation mode.

**ALLBCV**

PROTECTEDRESTORE(Y) is not required in the SPLIT command after any type of Protected RESTORE.

Aliases for PROTECTEDRESTORE include: PROTRESTORE, PROTRSTR, PRSTR, PROT.

The default value for this parameter can be predefined using site options.

TerminateExistingRelationship(Yes | No)

Allows or prohibits termination of an existing STD-to-BCV relationship when establishing a new relationship to the same BCV from another STD.
The TerminateExistingRelationship parameter prevents data corruption on the BCV when attempting to perform an ESTABLISH or a RESTORE on a BCV which already has relationship with another STD.

No

Prevents termination of the existing STD-to-BCV relationship.

Yes

(Defaul) Terminates the existing STD-to-BCV relationship.

The default value for this parameter can be predefined using site options.

TolerateDesiredState[{{NONE|ANY|[EST],[SPLIT],[RESTORE]}}] Determines commands in which the current state of the devices can be ignored if the MAXRC parameter is set to 4 or higher:

ANY

All commands.

EST

ESTABLISH commands.

NONE

(Defaul) No commands.

SPLIT

SPLIT commands.

RESTORE

RESTORE commands.

You can use one or more operands. If you do not specify any operands, the default value of NONE is used.

Note: TolerateDesiredState is ignored for Consistent SPLITs. If you issue TolerateDesiredState(SPLIT) or TolerateDesiredState(ANY) in the same job step as a Consistent SPLIT, you receive the BCVI127W message.

The default value for this parameter can be predefined using site options.

VXWAIT (Yes | No)

Note: The VXWAIT parameter is supported with PowerMaxOS 5978 and HYPERMAX OS 5977. Under Enginuity 5876 and 5773, the VXWAIT parameter is ignored.
The VXWAIT parameter fine-tunes TF/Mirror waiting behavior under PowerMaxOS 5978 and HYPERMAX OS 5977:

- For ESTABLISH/RE-ESTABLISH/RESTORE commands, it overrides the WAIT|NOWAIT value, making the BCV with point-in-time copy of data available to the host without waiting for tracks to be copied.
- For SPLIT commands, it determines whether TF/Mirror allows a SPLIT when the STD and BCV pair is not synchronized.

**IMPORTANT**
With PowerMaxOS 5978 and HYPERMAX OS 5977, TF/Mirror waits for the action to complete only when the WAIT|NOWAIT parameter is set to WAIT and the VXWAIT parameter is set to Yes. In all other cases, TF/Mirror does not wait for completion.

**WAIT|NOWAIT**
Determines whether TF/Mirror waits for the action to complete or continues after passing the command to the storage system.

**WAIT**
(Default) Wait for the action to complete.

**NOWAIT**
Pass the command to the storage system and assume that the action is complete.

**IMPORTANT**
With PowerMaxOS 5978 and HYPERMAX OS 5977, TF/Mirror waits for the action to complete only when the WAIT|NOWAIT parameter is set to WAIT and the VXWAIT parameter is set to Yes. In all other cases, TF/Mirror does not wait for completion.

The default wait times are 10 minutes for a SPLIT command and 120 minutes for ESTABLISH, RE-ESTABLISH, and RESTORE commands.

This setting can be overridden by the WAIT parameter in individual commands.

If there are multiple commands with the WAIT parameters at a sequence level, all wait conditions must be satisfied before the next sequence level can be executed.

The default value for this parameter can be predefined using site options.

**Examples**

1. To make TF/Mirror wait for completion of any command with the maximum return code of 8 before proceeding to the next command:

   ```
   GLOBAL WAIT, MAXRC(8, SETMAX)
   ```

2. The make TF/Mirror continue processing after passing a command to the storage system with a 5-second timeout for Consistent SPLITs:

   ```
   GLOBAL NOWAIT, CONS(TIMEOUT(5))
   ```
3. To set the maximum return code to 8 and make the following feature settings:
   - Use the clone emulation mode as requested in a particular command,
   - Use the Multi-Attach feature for ESTABLISH, RE-ESTABLISH, or RESTORE commands, and
   - Automatically release any STD holds:
     \[ \text{GLOBAL} \ \text{MAXRC}(8, \text{SETMAX}), \text{CLONE} (\text{REQUEST}), \text{MULTA} (\text{ALL}), \text{AUTOREL} (\text{Y}) \]

4. To make the following settings:
   - Set the maximum return code to 8 and reject commands if the maximum number of multiple BCVs is exceeded while ignoring the current state of devices,
   - Convert any RE-ESTABLISH operations that do not have a valid pair to a full ESTABLISH, and
   - Use the Multi-Attach feature for RE-ESTABLISH operations only:
     \[ \text{GLOBAL} \ \text{MAXRC}(8, \text{SETMAX}), \text{CONVERTFULL} \text{ESTABLISH} (\text{Y}), \text{MULTIATTACH} (\text{REEST}), \text{TDS} (\text{ALL}), \text{MULTBCV} (\text{REJ}) \]
QUERY

The QUERY command displays the current status of BCVs in the storage system. For more information about the QUERY functionality, see “TF/Mirror queries” on page 71.

Syntax

QUERY seq#, cuu[, optional_parameters]
QUERY seq#, LCL(cuu)[, optional_parameters]
QUERY seq#, RMT(cuu[, srdfgrp])[, optional_parameters]

Where optional_parameters are:

[,{ALL|count}[, symdv#]]
[,,EXtended(Y[, SBCV][, SNAPHold])]

Parameters

ALL
Displays all BCVs in the storage system. This is the default value.

Note: If the gatekeeper has a PowerMax/VMAX device number that is higher than that of the BCV in the storage system, no BCVs are displayed, unless you use the ALL parameter. By default, it only lists BCVs with PowerMax/VMAX device numbers higher than that of the gatekeeper.

count
The number of BCVs to display.

If not specified, all BCVs are displayed, starting with the device identified with the cuu.

cuu
An STD or BCV to start the query, identified with its CUU. BCVs at or above this CUU are displayed.

For a RMT operation with no SRDF group specified, the cuu must be an R1 device.

EXtended(Yes[,SBCV][,SNAPHold])
Enables the extended query mode.

Note: “TF/Mirror queries” on page 71 describes TF/Mirror queries.

SBCV
Starts the display with the BCV identified with the symdv#.

SNAPHold
Shows whether a volume is the source (HOLD-S) or the target (HOLD-T) of a snap.
The default value for this parameter can be predefined using site options.

**LCL**

Specifies that the devices are on the local side of an SRDF configuration.

**RMT**

Specifies that the devices are on the remote side of an SRDF configuration.

**seq#**

A 1-to-128 sequence number that defines the command execution order. Commands with equal sequence numbers are executed in parallel.

**srdfgrp**

The SRDF group or the hop list for the RMT operation.

Each SRDF group is represented by a one- or two-digit value. You can specify a hop list of up to four SRDF groups separated by periods, for example:

```
nn.nnn.nnn
```

**symdv#**

A device to start the query, identified with its PowerMax/VMAX device number.

In extended queries, `symdv#` normally refers to an STD because the Extended Format Query report is sorted by STD. However, the `symdv#` value does not have to be an STD. If you use the SBCV keyword (Extended(Y,SBCV), `symdv#` refers to the BCV with which to start the display.

**Note:** By default, the device to start the query is that identified with the PowerMax/VMAX device number of the gatekeeper. If this PowerMax/VMAX device number is higher than the last BCV in the remote storage system, no BCVs are displayed. If there is no gatekeeper with a PowerMax/VMAX device number lower than those of the BCVs to be displayed, the value of 0000 can be specified.
Examples

1. To display all BCVs starting with CUU 831F:

   QUERY 1,831F

   BCVI020I Start of INPUT control statement(s) from SYSIN
   BCVI021I End of INPUT control statement(s) from SYSIN

   BCVM039I (0001) Process input statement
   BCVM047I All control statements processed, highest RC 0

2. To display 3 BCVs starting with CUU 8330:

   QUERY 3,8330,3

   BCVI020I Start of INPUT control statement(s) from SYSIN
   BCVI021I End of INPUT control statement(s) from SYSIN

   BCVM047I All control statements processed, highest RC 0

3. To display 8 BCVs starting with PowerMax/VMAX device number 690, with CUU 881F used as the gatekeeper:

   QUERY 1,881F,8,690

   BCVI020I Start of INPUT control statement(s) from SYSIN
   BCVI021I End of INPUT control statement(s) from SYSIN

   BCVM047I All control statements processed, highest RC 0
4. To display 4 local BCVs starting with PowerMax/VMAX device number 47A, with CUU 831F used as the gatekeeper:

```
QUERY 1, LCL(831F), 4, 47A
```

```
BCVM039I (0002) Process input statement
BCVM004I QUERY status through device 831F, MICRO-CODE level 5x76 type VMAX-2, S/N 0001926-00215
BCVM003I ...BCV... ...STD... ACTION LAST PROT MIRROR BCV
BCVM003I CUU SYM# CUU SYM# ITRK-BCV ITRK-STD STATUS USED BCV EMUL #CYLS TYPE SYNC MODE
BCVM003I ---- 00047A1 ---- 00046A1 0 0 INUSE EST 00047A 3390 1113 NONE RD6/CLONE
BCVM003I ---- 00047B1 ---- 00046B1 0 0 INUSE EST 00047B 3390 1113 NONE RD6/CLONE
BCVM003I ---- 00047C1 ---- 00046C1 0 0 INUSE EST 00047C 3390 1113 NONE RD6/CLONE
BCVM003I ---- 00047D1 ---- 00046D1 0 0 INUSE EST 00047D 3390 1113 NONE RD6/CLONE
```

5. To display all BCVs in the remote storage system reached through gatekeeper CUU 831F and SRDF group 0D:

```
QUERY 2, RMT(831F, 0D), ALL
```

```
BCVM039I (0001) Process input statement
BCVM004I QUERY status on device 831F, MICRO-CODE level 5x76 type VMAX-2, S/N 0009876-54321
BCVM003I ...BCV... ...STD... ACTION LAST PROT MIRROR BCV
BCVM003I CUU SYM# CUU SYM# ITRK-BCV ITRK-STD STATUS USED BCV EMUL #CYL TYPE SYNC MODE
BCVM003I ---- 000615r ---- 000515r 0 0 AVAIL 000615 3390 3339 NONE RD6/CLONE
BCVM003I ---- 000616r ---- 000516r 0 0 AVAIL 000616 3390 3339 NONE RD6/CLONE
BCVM003I ---- 000617r ---- 000517r 0 0 AVAIL 000617 3390 3339 NONE RD6/CLONE
BCVM003I ---- 000618r ---- 000518r 0 0 AVAIL 000618 3390 3339 NONE RD6/CLONE
BCVM003I ---- 000619r ---- 000519r 0 0 AVAIL 000619 3390 3339 NONE RD6/CLONE
BCVM003I ---- 00061Ar ---- 00051Ar 0 0 AVAIL 00061A 3390 3339 NONE RD6/CLONE
BCVM003I ---- 00061Br ---- 00051Br 0 0 AVAIL 00061B 3390 3339 NONE RD6/CLONE
BCVM003I ---- 00061Cr ---- 00051Cr 0 0 AVAIL 00061C 3390 3339 NONE RD6/CLONE
BCVM003I ---- 00061Dr ---- 00051Dr 0 0 AVAIL 00061D 3390 3339 NONE RD6/CLONE
BCVM003I ---- 00061Er ---- 00051Er 0 0 AVAIL 00061E 3390 3339 NONE RD6/CLONE
BCVM003I ---- 00061Fr ---- 00051Fr 0 0 AVAIL 00061F 3390 3339 NONE RD6/CLONE
```

6. To display extended information about 3 STDs starting with PowerMax/VMAX device number 0FC2, with CUU 831F used as the gatekeeper:

```
QUERY 1, 831F, 4, FC2, EX(Y)
```

```
BCVM039I (0001) Process input statement
BCVM004I QUERY status on device 831F, MICRO-CODE level 5x76 type VMAX-2, S/N 0001234-56789
BCVM003I SYMDV# LAST TIME-FROM BCV BCV STD PAIR LAST TRACK TRACK TRACK
BCVM003I STD BCV BCV ...SPLIT.... CNFG STATE STATE STATE ACTION COUNT1 COUNT2 COUNT3
BCVM003I 000FC2 000FD2 000FD2 000.00:00:11 MIRR-C SYNC AVAIL AVAIL N/A 0 N/A
BCVM003I 000FC3 000FD3 000FD3 000.00:00:10 MIRR-C SYNC AVAIL AVAIL N/A 0 N/A
BCVM003I 000FC4 000FD4 000FD4 MIRR-C ATTACH AVAIL INUSE EST N/A 0 N/A
BCVM003I 000FC5 000FD5 000FD5 MIRR-C ATTACH AVAIL INUSE EST N/A 0 N/A
BCVM047I All control statements processed, highest RC 0
```
7. To display extended information, including information about whether a volume is a SNAP volume, on all STD and BCV pairs in the storage system, with CUU 8320 used as the gatekeeper:

```
QUERY 1,8320,EX(Y,SBCV,SNAPH)
```

BCVI020I Start of INPUT control statement(s) from SYSIN
BCVI018I (0001) QUERY 1,8320,EX(Y,SBCV,SNAPH)
BCVI021I End of INPUT control statement(s) from SYSIN

```
BCVM039I (0001) Process input statement
BCVM004I QUERY status on device 8320, MICRO-CODE level 5x76 type VMAX-2, S/N 0001234-56789
BCVM073I SYMDV#.. LAST TIME-FROM BVC BCV STD PAIR LAST TRACK TRACK TRACK
BCVM073I STD BVC BVC ...SPLIT... CNPG STATE STATE STATE ACTION COUNT1 COUNT2 COUNT3
BCVM073I 00006B 00040B 00044B 005.00:02:19 NONE-6 NR AVAIL AVAIL N/A 0 N/A
BCVM073I 00006C 00040C 00044C 005.00:02:19 NONE-6 NR AVAIL AVAIL N/A 0 N/A
BCVM073I 00006D 00040D 00044D 005.00:02:19 NONE-6 NR AVAIL AVAIL N/A 0 N/A
BCVM073I 00006E 00040E 00044E 005.00:02:19 NONE-6 NR AVAIL AVAIL N/A 0 N/A
BCVM073I 00006F 00040F 00044F 005.00:02:19 NONE-6 NR AVAIL AVAIL N/A 0 N/A
```

8. To display extended information about the 8 devices starting with the STD identified with PowerMax/VMAX device number 20C, with CUU 8320 used as the gatekeeper:

```
QUERY 1,8320,8,20C,EX(Y)
```

BCVI020I Start of INPUT control statement(s) from SYSIN
BCVI018I (0001) QUERY 1,8320,8,20C,EX(Y)
BCVI021I End of INPUT control statement(s) from SYSIN

```
BCVM039I (0001) Process input statement
BCVM004I QUERY status through device 8320, MICRO-CODE level 5x76 type VMAX-2, S/N 0001234-56789
BCVM073I STD BVC BVC ...SPLIT... CNPG STATE STATE STATE ACTION COUNT1 COUNT2 COUNT3
BCVM073I 00020C 00021C 00021C 005.00:37:09 NONE-C SYNC AVAIL INUSE EST N/A 0 N/A
BCVM073I 00020D 00021D 00021D 000.01:01:47 MIRR-C SYNC AVAIL AVAIL N/A 0 N/A
BCVM073I 00020E 00021E 00021E 000.01:01:46 MIRR-C HOLD-U AVAIL AVAIL N/A 0 N/A
BCVM073I 000210 000220 000220 000.01:01:45 MIRR-C SYNC AVAIL AVAIL N/A 0 N/A
BCVM073I 000211 000221 000221 000.01:01:44 MIRR-C SYNC AVAIL AVAIL N/A 0 N/A
BCVM073I 000212 000222 000222 MIRR-C ATTACH AVAIL INUSE EST N/A 0 N/A
BCVM073I 000213 000223 000223 MIRR-C ATTACH AVAIL INUSE EST N/A 0 N/A
```

9. To display extended information about 8 remote devices starting with the BCV identified with PowerMax/VMAX device number 100C, reached through gatekeeper CUU 8320 and SRDF group 0D:

```
QUERY 1,RMT(8320,0D),8,100C,EX(Y,SBCV)
```

BCVI020I Start of INPUT control statement(s) from SYSIN
BCVI018I (0001) QUERY 1,RMT(8320,0D),8,100C,EX(Y,SBCV)
BCVI021I End of INPUT control statement(s) from SYSIN

```
BCVM039I (0001) Process input statement
BCVM004I QUERY status through device 8320, MICRO-CODE level 5x76 type VMAX-2, S/N 0001234-56789
BCVM073I STD BVC BVC ...SPLIT... CNPG STATE STATE STATE ACTION COUNT1 COUNT2 COUNT3
BCVM073I 000C0C 00100C 00100C 005.00:37:09 NONE-C SYNC AVAIL AVAIL N/A 0 N/A
BCVM073I 000C0D 00100D 00100D 005.00:37:09 NONE-C SYNC AVAIL AVAIL N/A 0 N/A
BCVM073I 000C0E 00100E 00100E 005.00:37:09 NONE-C SYNC AVAIL AVAIL N/A 0 N/A
BCVM073I 000C0F 00100F 00100F 005.00:37:09 NONE-C SYNC AVAIL AVAIL N/A 0 N/A
BCVM073I 000C10 001010 001010 MIRR-C ATTACH AVAIL INUSE EST N/A 0 N/A
BCVM073I 000C11 001011 001011 MIRR-C ATTACH AVAIL INUSE EST N/A 0 N/A
BCVM073I 000C12 001012 001012 MIRR-C ATTACH AVAIL INUSE EST N/A 0 N/A
BCVM073I 000C13 001013 001013 MIRR-C ATTACH AVAIL INUSE EST N/A 0 N/A
```

All control statements processed, highest RC 0
10. To display 8 remote BCVs starting with PowerMax/VMAX device number 5E0, reached through gatekeeper CUU 851F and SRDF groups F0 and 52:

```
QUERY 1,RMT(851F,F0.52),8,5E0
```

BCVI020I Start of INPUT control statement(s) from SYSIN
BCVI018I (0001) QUERY 1,RMT(851F,F0.52),8,5E0
BCVI021I End of INPUT control statement(s) from SYSIN

```
BCVM039I (0001) Process input statement
BCVM004I QUERY status through device 851F, MICRO-CODE level 5x76 type VMAX-2, S/N 0009876-54121
BCVM003I ...BCV... ...STD... ACTION LAST PROT MIRROR BCV
BCVM004I CUU SYM#  CUU SYM#  ITRK-BCV  ITRK-STD STATUS USED  BCV  EMUL  #CYLS TYPE  SYNC  MODE
BCVM003I ---- 0005E0r ---- 0001A2r 0 0 AVAIL 0005E0 3390 1113 MIRR RD5/CLONE
BCVM003I ---- 0005E1r ---- 0001A3r 0 0 AVAIL 0005E1 3390 1113 MIRR RD5/CLONE
BCVM003I ---- 0005E2r ---- 0001A4r 0 0 AVAIL 0005E2 3390 1113 MIRR RD5/CLONE
BCVM003I ---- 0005E3r ---- 0001A5r 0 0 AVAIL 0005E3 3390 1113 MIRR RD5/CLONE
BCVM003I ---- 0005E4r ---- 0001A6r 0 0 AVAIL 0005E4 3390 1113 MIRR RD5/CLONE
BCVM003I ---- 0005E5r ---- 0001A7r 0 0 AVAIL 0005E5 3390 1113 MIRR RD5/CLONE
BCVM003I ---- 0005E6r ---- 0001A8r 0 0 AVAIL 0005E6 3390 1113 MIRR RD5/CLONE
BCVM003I ---- 0005E7r ---- 0001A9r 0 0 INUSE 0005E7 3390 1113 MIRR RD5/CLONE

BCVM047I All control statements processed, highest RC 0
```
RE-ESTABLISH

The RE-ESTABLISH command copies the contents of the STD in a previously split STD/BCV pair to the BCV. The BCV becomes Not Ready to the host.

The RE-ESTABLISH command reassigns a BCV to the STD to which it was assigned before the SPLIT. Any data written to the BCV while it was split from the STD is overwritten on the BCV. The BCV receives its updates from the STD.

⚠️ CAUTION ⚠️

If there was any changed data on the BCV, the data is lost.

The BCV must be offline to all hosts to which it is attached.

For more information about the RE-ESTABLISH functionality, see “Re-synchronizing from STD to BCV: RE-ESTABLISH” on page 47.

Syntax

RE-ESTABLISH  seq#, cuus  
[ [, optional_parameters] ]

RE-ESTABLISH  seq#, cuus-cuus  
[ [, optional_parameters] ]

RE-ESTABLISH  seq#, LCL(cuu, symdv#bcv)  
[ [, optional_parameters] ]

RE-ESTABLISH  seq#, LCL(cuu, symdv#bcv=symdv#bcv)  
[ [, optional_parameters] ]

RE-ESTABLISH  seq#, RMT(cuu, symdv#bcv[, srdgrp])  
[ [, optional_parameters] ]

RE-ESTABLISH  seq#, RMT(cuu, symdv#bcv-symdv#bcv [, srdgrp])  
[ [, optional_parameters] ]

Where optional_parameters are:

[ [, CBCV({Yes|No}) ] ]

[ [, CLONEemulation({Yes|No}) ] ]

[ [, MULTIATTach({Yes|No}) ] ]

[ [, PROTECTEDBCVEstablish({Yes|No}) ] ]

[ [, VXWAIT(Yes|No) ] ]

[ [, {WAIT[(nnnn)]]|NOWAIT} ]]

Parameters

CBCV({Yes|No})

**IMPORTANT**

The CBCV parameter is ignored in the clone emulation mode.
Allows to use the BCV as a Concurrent BCV for the STD. The default value is No.

**Note:** “Concurrent BCV” on page 40 describes the Concurrent BCV feature.

A Concurrent BCV ESTABLISH is not allowed if the first pair was attached with a RESTORE command.

If mirror positions are available, you can specify a maximum of two BCVs for an STD.

**CLONEemulation**

```
{{Yes | No}}
```

**IMPORTANT**

Under PowerMaxOS 5978, HYPERMAX OS 5977 and Enginuity 5876, the CLONEemulation parameter is ignored. All operations are processed in the clone emulation mode.

Enables or disables the clone emulation mode. The default value is No.

**Note:** “Clone emulation” on page 52 describes the clone emulation mode.

Under Enginuity 5773, RE-ESTABLISH always uses the clone emulation mode with RAID5- and RAID6-protected BCVs. For other BCVs, RE-ESTABLISH commands use the clone emulation mode only if they are in a clone emulation session already established by an ESTABLISH or a full RESTORE command.

**cuu**

A device identified with its CUU.

For a RMT operation, if you do not specify the SRDF group (srdfgrp), then cuu must be an R1 device.

**cuus**

A BCV identified with its CUU.

**cuus-cuus**

A range of BCVs identified with their CUUs. All devices in the range must reside in the same storage system. The number of devices in cuus-cuus and cuup-cuup ranges must be the same.

**LCL**

Specifies that the devices are on the local side of an SRDF configuration.

**MULTIATTach**

```
{{Yes | No}}
```

Enables or disables the Multi-Attach feature for this RE-ESTABLISH command.

- **Yes**—Enables the Multi-Attach feature, unless the GLOBAL MULTIATTach parameter is set to NONE.

- **No**—Disables the Multi-Attach feature regardless of the GLOBAL MULTIATTach parameter value.
The default value is Yes.

Note: “Multi-attach” on page 56 describes the Multi-Attach feature.

Aliases for MULTIATTach include: MULTA.

PROTECTEDBCVEstablish({Yes | No})

Enables or disables the Protected BCV feature. The default value is No.

Note: “Protected BCV” on page 41 describes the Protected BCV feature.

IMPORTANT

The PROTECTEDBCVEstablish parameter is ignored in the clone emulation mode. All operations are processed using the Protected BCV feature.

Aliases for PROTECTEDBCVEstablish include: PBCVE, PROTECTEDBCVE, PROTBCVE.

RMT

Specifies that the devices are on the remote side of an SRDF configuration.

seq#

A 1-to-128 sequence number that defines the command execution order. Commands with equal sequence numbers are executed in parallel.

srdfgrp

The SRDF group or the hop list for the RMT operation.

Each SRDF group is represented by a one- or two-digit value. You can specify a hop list of up to four SRDF groups separated by periods, for example:

nn.nn.nn.nn

symdv#bcv

A BCV identified with its PowerMax/VMAX device number.

symdv#bcv-symdv#bcv

A range of BCVs identified with their PowerMax/VMAX device numbers.

VXWAIT (Yes | No)

Note: The VXWAIT parameter is supported with PowerMaxOS 5978 and HYPERMAX OS 5977. Under Enginuity 5876 and 5773, the VXWAIT parameter is ignored.

The VXWAIT parameter fine-tunes TF/Mirror waiting behavior under PowerMaxOS 5978 and HYPERMAX OS 5977 by providing the possibility to override the WAIT|NOWAIT value set for the ESTABLISH/RE-ESTABLISH/RESTORE commands and thus to allow a SPLIT with invalid tracks. The benefit is that the BCV with point-in-time copy of data is available to the host without waiting for tracks to be copied.
With PowerMaxOS 5978 and HYPERMAX OS 5977, TF/Mirror waits for the action to complete only when the WAIT|NOWAIT parameter is set to WAIT and the VXWAIT parameter is set to Yes. In all other cases, TF/Mirror does not wait for completion.

The VXWAIT parameter can be used to override the settings made with the GLOBAL command. The default value for this parameter can be predefined using site options.

**WAIT\[(nnnn)\] | NOWAIT**

Determines whether to wait for the action to complete:

- **WAIT**—Wait for the action to complete.
- **NOWAIT**—Pass the command to the storage system and assume that the action is complete.
- **nnnn**—Wait for nnnn minutes and assume that the action is complete. For example, WAIT(10) means a 10-minute wait. A maximum of up to 9999 minutes can be specified.

**Note:** If WaitSync is specified, the time \((nnnn)\) also includes the waiting time allocated to synchronize the BCV mirror.

**IMPORTANT**

With PowerMaxOS 5978 and HYPERMAX OS 5977, TF/Mirror waits for the action to complete only when the WAIT|NOWAIT parameter is set to WAIT and the VXWAIT parameter is set to Yes. In all other cases, TF/Mirror does not wait for completion.

This parameter can be used to override the settings made with the GLOBAL command. The default value for this parameter can be predefined using site options.

**Examples**

1. To copy the contents of the STD previously paired with CUU 7B35 (the BCV) to its BCV without waiting for the action to complete:

   **RE-ESTABLISH 1,7B35,NOWAIT**

   BCVI020I Start of INPUT control statement(s) from SYSIN
   BCVI018I (0001) GLOBAL MAXRC(8,SETMAX)
   BCVI018I (0002) RE-ESTABLISH 1,7B35,NOWAIT
   BCVI021I End of INPUT control statement(s) from SYSIN

   BCVM039I (0002) Process input statement
   BCVM004I RE-ESTABLISH BCV 7B35
   BCVM069I Security Exit allowed the bypassing of the online state check
   BCVM140I Command processed via TF/Clone emulation
   BCVM047I All control statements processed, highest RC 0
2. To copy the contents of STDs previously paired with CUUs 7B35-7B38 (the BCVs) to their BCVs and wait for the action to complete:

BCVI020I Start of INPUT control statement(s) from SYSIN
BCVI018I (0001) RE-ESTABLISH 3,7B35-7B38
BCVI021I End of INPUT control statement(s) from SYSIN

BCVM039I (0001) Process input statement
BCVM004I RE-ESTABLISH BCV 7B35
BCVM069I Security Exit allowed the bypassing of the online state check
BCVM140I Command processed via TF/Clone emulation

BCVM039I (0001) Process input statement
BCVM004I RE-ESTABLISH BCV 7B36
BCVM069I Security Exit allowed the bypassing of the online state check
BCVM140I Command processed via TF/Clone emulation

BCVM039I (0001) Process input statement
BCVM004I RE-ESTABLISH BCV 7B37
BCVM069I Security Exit allowed the bypassing of the online state check
BCVM140I Command processed via TF/Clone emulation

BCVM039I (0001) Process input statement
BCVM004I RE-ESTABLISH BCV 7B38
BCVM069I Security Exit allowed the bypassing of the online state check
BCVM140I Command processed via TF/Clone emulation

BCVM047I All control statements processed, highest RC 0

3. To copy the contents of STDs previously paired with local PowerMax/VMAX device numbers 690-693 (the BCVs) to their BCVs through gatekeeper CUU 881F:

RE-ESTABLISH 1,LCL(881F,690-693)

BCVI020I Start of INPUT control statement(s) from SYSIN
BCVI018I (0001) RE-ESTABLISH 1,LCL(881F,690-693)
BCVI021I End of INPUT control statement(s) from SYSIN

BCVM039I (0001) Process input statement
BCVM004I RE-ESTABLISH LOCAL BCV SYMDEV 000690 through 881F
BCVM069I Security Exit allowed the bypassing of the online state check
BCVM140I Command processed via TF/Clone emulation

BCVM039I (0001) Process input statement
BCVM004I RE-ESTABLISH LOCAL BCV SYMDEV 000691 through 881F
BCVM069I Security Exit allowed the bypassing of the online state check
BCVM140I Command processed via TF/Clone emulation

BCVM039I (0001) Process input statement
BCVM004I RE-ESTABLISH LOCAL BCV SYMDEV 000692 through 881F
BCVM069I Security Exit allowed the bypassing of the online state check
BCVM140I Command processed via TF/Clone emulation

BCVM039I (0001) Process input statement
BCVM004I RE-ESTABLISH LOCAL BCV SYMDEV 000693 through 881F
BCVM069I Security Exit allowed the bypassing of the online state check
BCVM140I Command processed via TF/Clone emulation

BCVM047I All control statements processed, highest RC 0
4. To copy the contents of STDs previously paired with remote PowerMax/VMAX device numbers 690-693 (the BCVs) to their BCVs through gatekeeper CUU 881F and SRDF group 2F:

```
RE-ESTABLISH 1,RMT(851F,690-693,2F)
```

```
BCVI020I Start of INPUT control statement(s) from SYSIN
BCVI018I (0001) RE-ESTABLISH 1,RMT(851F,690-693,2F)
BCVI021I End of INPUT control statement(s) from SYSIN

BCVM039I (0001) Process input statement
BCVM004I RE-ESTABLISH REMOTE BCV SYMDEV 000690 through 851F
BCVM069I Security Exit allowed the bypassing of the online state check
BCVM140I Command processed via TF/Clone emulation

BCVM039I (0001) Process input statement
BCVM004I RE-ESTABLISH REMOTE BCV SYMDEV 000691 through 851F
BCVM069I Security Exit allowed the bypassing of the online state check
BCVM140I Command processed via TF/Clone emulation

BCVM039I (0001) Process input statement
BCVM004I RE-ESTABLISH REMOTE BCV SYMDEV 000692 through 851F
BCVM069I Security Exit allowed the bypassing of the online state check
BCVM140I Command processed via TF/Clone emulation

BCVM039I (0001) Process input statement
BCVM004I RE-ESTABLISH REMOTE BCV SYMDEV 000693 through 851F
BCVM069I Security Exit allowed the bypassing of the online state check
BCVM140I Command processed via TF/Clone emulation

BCVM047I All control statements processed, highest RC 0
```

5. To copy the contents of STDs previously paired with remote PowerMax/VMAX device numbers 5F0-5FF (the BCVs) to their BCVs through gatekeeper CUU 85B0 using the Multi-Attach feature (because an SRDF group is not specified, the SRDF group of the gatekeeper is used):

```
RE-ESTABLISH 1,RMT(85B0,5F0-5FF),MULTATT(YES)
```

```
BCVI020I Start of INPUT control statement(s) from SYSIN
BCVI018I (0001) RE-ESTABLISH 1,RMT(85B0,5F0-5FF),MULTATT(YES)
BCVI021I End of INPUT control statement(s) from SYSIN

BCVM039I (0001) Process input statement
BCVM069I Security Exit allowed the bypassing of the online state check
BCVM004I MULTI RE-ESTABLISH REMOTE devices through 85B0:
BCVM004I   0005F0-0005FF(0001)
BCVM140I Command processed via TF/Clone emulation

BCVM047I All control statements processed, highest RC 0
6. To copy the contents of STDs previously paired with remote PowerMax/VMAX device numbers 5F0-5FF (the BCVs) to their BCVs through gatekeeper CUU 851F and SRDF groups F0 and 52 using the Multi-Attach and Protected BCV features (the BCV and the BCV mirror are reestablished with the STD):

GLOBAL MULTA(ALL)
RE-ESTABLISH 1,RMT(851F,5E0-5EF,F0.52),PBCVE(Y)

7. The following example illustrates issuing RE-ESTABLISH commands on held devices and receiving BCVI46E error messages that indicate that the operation was denied because the devices are in the HOLD status:

GLOBAL MULTA(ALL)
RE-ESTABLISH 1,RMT(851F,5E0-5EF,F0.52),PBCVE(Y)

GLOBAL MULTA(ALL)
RE-ESTABLISH 1,LCL(861F,4B0-4B3)

GLOBAL MULTA(ALL)
RE-ESTABLISH 1,LCL(861F,4B0-4B3)
8. The following example illustrates mixing ESTABLISH and RE-ESTABLISH commands:

BCVI018I (0001) GLOBAL MAXRC(8,SETMAX),MULTA(ALL)
BCVI018I (0002) ESTABLISH 1,LCL(831F,3AA,3BA)
BCVI018I (0003) ESTABLISH 1,LCL(831F,3AB,3BB)
BCVI018I (0004) RE-ESTABLISH 1,LCL(831F,200)
BCVI018I (0005) RE-ESTABLISH 1,LCL(831F,201)
BCVI021I End of INPUT control statement(s) from SYSIN

BCVM039I (0002) Process input statement  
BCVM080I ACCESS ALLOWED  
BCVM069I Security Exit allowed the bypassing of the online state check  
BCVM004I MULTI ESTABLISH LOCAL devices through 831F:  
BCVM004I 0003AA/0003BA(0002),0003AB/0003BB(0003)  
BCVM140I Command processed via TF/Clone emulation

BCVM039I (0004) Process input statement  
BCVM080I ACCESS ALLOWED  
BCVM069I Security Exit allowed the bypassing of the online state check  
BCVM004I MULTI RE-ESTABLISH LOCAL devices through 831F:  
BCVM004I 0200(0004),0201(0005)  
BCVM140I Command processed via TF/Clone emulation

BCVM047I All control statements processed, highest RC 0
RESTORE

The RESTORE command assigns the BCV to an STD and copies data from the BCV to the STD. The BCV becomes *Not Ready* to the host.

⚠️ **CAUTION**

If there was any changed data on the STD, the data is lost. For more information about the RESTORE functionality, see “Synchronizing from BCV to STD: RESTORE” on page 48.

With PowerMaxOS 5978 and HYPERMAX OS 5977, to process a RESTORE command immediately after a SPLIT, you must specify WAIT and VXWAIT(YES) in the SPLIT command.

Syntax

**Full RESTORE:**

```
RESTORE seq#, cuus, cuup, VERIFY(volser|FBADEV) [,optional_parameters]
```

```
RESTORE seq#, LCL(cuu, sym#bcv, sym#std), VERIFY(volser|FBADEV) [,optional_parameters]
```

```
RESTORE seq#, LCL(cuu, sym#bcv-sym#bcv, sym#std-sym#std), VERIFY(FBADEV) [,optional_parameters]
```

**Incremental RESTORE:**

```
RESTORE seq#, cuus [,optional_parameters]
```

```
RESTORE seq#, cuus-cuus [,optional_parameters]
```

```
RESTORE seq#, LCL(cuu, symdv#bcv) [,optional_parameters]
```

```
RESTORE seq#, LCL(cuu, symdv#bcv-symdv#bcv) [,optional_parameters]
```

```
RESTORE seq#, RMT(cuu, symdv#bcv[, srdgrp]) [,optional_parameters]
```

```
RESTORE seq#, RMT(cuu, symdv#bcv-symdv#bcv[, srdgrp]) [,optional_parameters]
```

Where *optional_parameters* are:

```
[,CLONEemulation({Yes|No})]
```

```
[,PROTECTEDRESTORE({Yes|No})]
```

```
[,R1SYNC({Yes|No})]
```

```
[,VXWAIT(Yes|No)]
```

```
[,{WAIT[(nnnn)]|NOWAIT}]
```
Parameters

CLONEemulation({Yes|No})

IMPORTANT
Under PowerMax OS 5978, HYPERMAX OS 5977 and Enginuity 5876, the CLONEemulation parameter is ignored. All operations are processed in the clone emulation mode.

Enables or disables the clone emulation mode. The default value is No.

Note: “Clone emulation” on page 52 describes the clone emulation mode.

cuu
A device identified with its CUU.
For a RMT operation, if you do not specify the SRDF group (srdfgrp), then cuu must be an R1 device.

cuup
An STD identified with its CUU.

cuus
A BCV identified with its CUU.

cuus-cuus
A range of BCVs identified with their CUUs. All devices in the range must reside in the same storage system. The number of devices in cuus-cuus and cuup-cuup ranges must be the same.

LCL
Specifies that the devices are on the local side of an SRDF configuration.

PROTECTEDRESTORE({Yes|No})

IMPORTANT
The PROTECTEDRESTORE parameter is ignored in the clone emulation mode. All operations are processed using the Protected RESTORE feature. This means that PROTECTEDRESTORE(Y) is not required in the SPLIT command after a RESTORE operation performed in the clone emulation mode.

Enables or disables the Protected RESTORE feature.
When the PROTECTEDRESTORE parameter is set to Yes, the BCV is protected against host writes in its STD until the next SPLIT command.
The default value is No.

Note: “Protected RESTORE” on page 49 describes the Protected RESTORE feature.

Aliases for PROTECTEDRESTORE include: PROTRESTORE, PROTRSTR, PRSTR, PROT.
R1SYNC({Yes|No})

**IMPORTANT**

A RESTORE to an R2 STD with the R1SYNC(y) parameter causes an error message.

Enables or disables automatic R2 to R1 synchronization. The default value is No.

If you specify Yes, the R1 device enters the TNR (Target Not Ready) state and the R2 device has invalid tracks for the R1 device.

**RMT**

Specifies that the devices are on the remote side of an SRDF configuration.

**seq#**

A 1-to-128 sequence number that defines the command execution order. Commands with equal sequence numbers are executed in parallel.

**srdfgrp**

The SRDF group or the hop list for the RMT operation.

Each SRDF group is represented by a one- or two-digit value. You can specify a hop list of up to four SRDF groups separated by periods, for example:

```
nn.nn.nn.nn
```

**symdv#bcv**

A BCV identified with its PowerMax/VMAX device number.

**symdv#bcv-symdv#bcv**

A range of BCVs identified with their PowerMax/VMAX device numbers.

**symdv#std**

An STD identified with its PowerMax/VMAX device number.

**symdv#std-symdv#std**

A range of STDs identified with their PowerMax/VMAX device numbers.

**VERIFY(volser|FBADEV)**

**IMPORTANT**

This is a required parameter for a full RESTORE. For an incremental RESTORE, VERIFY is ignored.

Ensures that the volume specified by volser is the same as the STD specified by cuup.

If the specified BCV is an FBA device, specify VERIFY(FBADEV).

**VXWAIT(Yes|No)**

**Note:** The VXWAIT parameter is supported with PowerMaxOS 5978 and HYPERMAX OS 5977. Under Enginiuity 5876 and 5773, the VXWAIT parameter is ignored.
The VXWAIT parameter fine-tunes TF/Mirror waiting behavior under PowerMaxOS 5978 and HYPERMAX OS 5977 by providing the possibility to override the WAIT|NOWAIT value set for the ESTABLISH/RE-ESTABLISH/RESTORE commands and thus to allow a SPLIT with invalid tracks. The benefit is that the BCV with point-in-time copy of data is available to the host without waiting for tracks to be copied.

With PowerMaxOS 5978 and HYPERMAX OS 5977, TF/Mirror waits for the action to complete only when the WAIT|NOWAIT parameter is set to WAIT and the VXWAIT parameter is set to Yes. In all other cases, TF/Mirror does not wait for completion.

The VXWAIT parameter can be used to override the settings made with the GLOBAL command. The default value for this parameter can be predefined using site options.

```
WAIT[(nnnn)]|NOWAIT
```

Determines whether to wait for the action to complete:

- **WAIT**—Wait for the action to complete.
- **NOWAIT**—Pass the command to the storage system and assume that the action is complete.
- **nnnn**—Wait for nnnn minutes and assume that the action is complete. For example, WAIT(10) means a 10-minute wait. A maximum of up to 9999 minutes can be specified.

**Note:** If WaitSync is specified, the time (nnnn) also includes the waiting time allocated to synchronize the BCV mirror.

**IMPORTANT**

With PowerMaxOS 5978 and HYPERMAX OS 5977, TF/Mirror waits for the action to complete only when the WAIT|NOWAIT parameter is set to WAIT and the VXWAIT parameter is set to Yes. In all other cases, TF/Mirror does not wait for completion.

This parameter can be used to override the settings made with the GLOBAL command. The default value for this parameter can be predefined using site options.

**Examples**

1. To copy data from the BCV identified with CUU DE18 to the STD identified with CUU DE10, with volser USK272, waiting for the action to complete (note that clone emulation is being used, so the RESTORE is protected):

```
RESTORE 1,DE18,DE10,WAIT,VERIFY(USK272)
```

```
BCVI020I Start of INPUT control statement(s) from SYSIN
BCVI018I (0001) RESTORE 1,DE18,DE10,WAIT,VERIFY(USK272)
BCVI021I End of INPUT control statement(s) from SYSIN
BCVM039I (0001) Process input statement
BCVM004I RESTORE BCV DE18 to DE10
BCVM080I ACCESS ALLOWED
BCVM069I Security Exit allowed the bypassing of the online state check
BCVM069I Security Exit allowed the bypassing of the online state check
```
BCVM070I Security Exit allowed the bypassing of the WTOR on a Full Restore
BCVM139I BCV DE18 is TF/Clone, Restore is PROTECTED
BCVM140I Command processed via TF/Clone emulation
BCVM047I All control statements processed, highest RC 0

2. To copy changed tracks from the BCVs identified with CUUs DF21 and DF22:

RESTORE 1,DF21-DF22

BCVI020I Start of INPUT control statement(s) from SYSIN
BCVI018I (0001) RESTORE 1,DF21-DF22
BCVI021I End of INPUT control statement(s) from SYSIN

BCVM039I (0001) Process input statement
BCVM004I RESTORE BCV DF21
BCVM080I ACCESS ALLOWED
BCVM069I Security Exit allowed the bypassing of the online state check
BCVM071I Security Exit allowed the bypassing of the WTOR on a Partial Restore

BCVM039I (0001) Process input statement
BCVM004I RESTORE BCV DF22
BCVM069I Security Exit allowed the bypassing of the online state check
BCVM071I Security Exit allowed the bypassing of the WTOR on a Partial Restore

BCVM047I All control statements processed, highest RC 0

3. To copy changed tracks from the remote BCV identified with PowerMax/VMAX device number 01CC in the clone emulation mode, reached through CUU C120 and SRDF group 70:

RESTORE 1,RMT(C120,01CC,70)

BCVI020I Start of INPUT control statement(s) from SYSIN
BCVI018I (0001) RESTORE 1,RMT(C120,01CC,70)
BCVI021I End of INPUT control statement(s) from SYSIN

BCVM039I (0001) Process input statement
BCVM004I RESTORE REMOTE BCV SYMDEV 0001CC through C120
BCVM080I ACCESS ALLOWED
BCVM069I Security Exit allowed the bypassing of the online state check
BCVM071I Security Exit allowed the bypassing of the WTOR on a Partial Restore
BCVM139I BCV 0001CC is TF/Clone, Restore is PROTECTED
BCVM140I Command processed via TF/Clone emulation
BCVM047I All control statements processed, highest RC 0

4. To copy changed tracks from the remote BCVs identified with PowerMax/VMAX device numbers 1D4, 1D5, and 1D6 in the clone emulation mode, reached through CUU C120 and SRDF group E0:

RESTORE 1,RMT(C120,01D4-1D6,E0)

BCVI020I Start of INPUT control statement(s) from SYSIN
BCVI018I (0001) RESTORE 1,RMT(C120,01D4-1D6,E0)
BCVI021I End of INPUT control statement(s) from SYSIN

BCVM039I (0001) Process input statement
BCVM004I RESTORE REMOTE BCV SYMDEV 0001D4 through C120
BCVM080I ACCESS ALLOWED
BCVM069I Security Exit allowed the bypassing of the online state check
BCVM071I Security Exit allowed the bypassing of the WTOR on a Partial Restore
BCVM139I BCV 0001D4 is TF/Clone, Restore is PROTECTED
BCVM140I Command processed via TF/Clone emulation
BCVM047I (0001) Process input statement
5. To copy changed tracks from the local PowerMax/VMAX device numbers 1FD-1FF, reached through CUU DF60:

```
RESTORE 1,LCL(DF60,01FD-01FF)
```

```
BCVI020I Start of INPUT control statement(s) from SYSIN
BCVI018I (0001) RESTORE 1,LCL(DF60,01FD-1FF)
BCVI021I End of INPUT control statement(s) from SYSIN
BCVM039I (0001) Process input statement
BCVM004I RESTORE LOCAL BCV SYMDEV 0001FD through DF60
BCVM080I ACCESS ALLOWED
BCVM069I Security Exit allowed the bypassing of the online state check
BCVM071I Security Exit allowed the bypassing of the WTOR on a Partial Restore

BCVM039I (0001) Process input statement
BCVM004I RESTORE LOCAL BCV SYMDEV 0001FE through DF60
BCVM069I Security Exit allowed the bypassing of the online state check
BCVM071I Security Exit allowed the bypassing of the WTOR on a Partial Restore

BCVM039I (0001) Process input statement
BCVM004I RESTORE LOCAL BCV SYMDEV 0001FF through DF60
BCVM069I Security Exit allowed the bypassing of the online state check
BCVM071I Security Exit allowed the bypassing of the WTOR on a Partial Restore

BCVM047I All control statements processed, highest RC 0

6. To copy data from the remote BCVs identified with PowerMax/VMAX device numbers 18D and 18E, reached through CUU DE20 and SRDF group E0, in the Protected RESTORE mode:

```
RESTORE 4,RMT(DE20,18D-18E,E0),PROTRSTR(Y)
```

```
BCVI020I Start of INPUT control statement(s) from SYSIN
BCVI018I (0001) RESTORE 4,RMT(DE20,18D-18E,E0),PROTRSTR(Y)
BCVI021I End of INPUT control statement(s) from SYSIN
BCVM039I (0001) Process input statement
BCVM004I RESTORE REMOTE BCV SYMDEV 00018D through DE20
BCVM080I ACCESS ALLOWED
BCVM069I Security Exit allowed the bypassing of the online state check
BCVM071I Security Exit allowed the bypassing of the WTOR on a Partial Restore

BCVM039I (0001) Process input statement
BCVM004I RESTORE REMOTE BCV SYMDEV 00018E through DE20
BCVM069I Security Exit allowed the bypassing of the online state check
BCVM071I Security Exit allowed the bypassing of the WTOR on a Partial Restore

BCVM047I All control statements processed, highest RC 8
SPLIT

The SPLIT command splits the BCV from the STD to stop the mirroring process between the two devices. After the STD/BCV pair is split, the changed tracks for both devices are logged, and the BCV becomes Ready to the host.

Normally, all devices in the range to be SPLIT must be ATTACHED.

After the split occurs, you may resume the STD/BCV relationship between the two devices. In this case, only changed tracks are synchronized.

**Note:** Only RAID10 head devices can be used. Do not include member devices in the ranges.

For more information about the SPLIT functionality, see “Stopping synchronization: SPLIT” on page 42.

**Syntax**

SPLIT seq#,cuus[,VOLID(volser[,E])]
    [,optional_parameters]

SPLIT seq#,cuus-cuus
    [,optional_parameters]

SPLIT seq#,LCL(cuu,symdv#bcv)
    [,optional_parameters]

SPLIT seq#,LCL(cuu,symdv#bcv-symdv#bcv)
    [,optional_parameters]

SPLIT seq#,RMT(cuu,symdv#bcv[,srdfgrp])
    [,optional_parameters]

SPLIT seq#,RMT(cuu,symdv#bcv-symdv#bcv[,srdfgrp])
    [,optional_parameters]

Where **optional_parameters** are:

    [,BCVRefresh({Yes|No})]
    [,ChangedOnly({Yes|No})]
    [,CLONEemulation({Yes|No})]
    [,CONSistent({GLOBAL[{(ECA|IOSLevel})]|LOCAL[{(ECA|IOSLevel})}])]
    [,STDcuu(cuu)][,TIMEOUT(nn)]
    [,FORCE|NOFORCE]
    [,HOLD]
    [,INSTant({Yes|No})]
    [,NR]
    [,PROTECTEDRESTORE({Yes|No})]
    [,R2Sync({Yes|No})]
    [,VXWAIT(Yes|No)]
    [,{WAIT[(nnnn)]|NOWAIT}]
    [,WaitSync(WTO({Yes|No}))]
Parameters

BCVRefresh({Yes|No})

IMPORTANT
The BCVRefresh parameter is ignored in the clone emulation mode. A BCVREFRESH(Y) included in a SPLIT command results in a BCVM138W message and a return code of four (4) when no MAXRC is specified.

Enables or disables the Reverse SPLIT mode.

**Note:** “Reverse SPLIT” on page 45 describes the Reverse SPLIT mode.

The default value for this parameter can be predefined using site options.

ChangedOnly({Yes|No})
Synchronizes only changed tracks on the BCV mirrors.

**Note:** This is a legacy parameter. Under currently supported operating environment levels, all SPLITs are incremental and setting the ChangedOnly parameter to No generates a BCVI101W message.

CLONEemulation({Yes|No})
Enables or disables the clone emulation mode. The default value is No.

**Note:** “Clone emulation” on page 52 describes the clone emulation mode.

IMPORTANT
Under PowerMaxOS 5978, HYPERMAX OS 5977 and Enginuity 5876, the CLONEemulation parameter is ignored. All operations are processed in the clone emulation mode.

CONSistent({GLOBAL[{(ECA|IOSLevel)}]|LOCAL[(BYPassonlinecheck[,{ECA|IOSLevel}])] [,STDcuu(cuu)] [,TIMEOUT(nn)]

Performs a Consistent SPLIT by halting the host I/O to the STDs of the STD/BCV pairs for the duration of the SPLIT operation.

**Note:** “Consistent SPLIT” on page 43 describes Consistent SPLITs.

You can specify the RMT parameter to perform a Consistent SPLIT in a remote storage system.

BYPassonlinecheck
Bypasses the online/offline status check for STDs on other LPARs. This means that there is no dependent I/O from other LPARs to the volume(s) in a local (single-system) Consistent SPLIT.

**Note:** “Online/offline status check” on page 55 describes the online/offline status check.

**ECA**

Sets the I/O control mode to ECA. ECA is the default I/O control mode.

**Note:** For information about the ECA mode, see the *Consistency Groups for z/OS Product Guide*.

**IOSLEVEL**

Sets the I/O control mode to IOSLEVEL.

**Note:** ECA is used if available, even if IOSLEVEL is specified on a Consistent SPLIT.

**GLOBAL**

Specifies a multisystem Consistent SPLIT.

**LOCAL**

Specifies a single-system Consistent SPLIT.

**STDcuu (cuu)**

Helps determine the CUU of the STD when ranges of devices are split.

When the relationship between the STD and BCVs is not symmetric, TF/Mirror cannot determine the STDcuu that is paired with the BCV. In this case, the STDcuu parameter is needed. When specified with LOCAL and RMT keywords that specify a range, the STDcuu references the first STD in the range.

The STDcuu parameter is optional for a remote Consistent SPLIT and for any other form of Consistent SPLITs.

If you do specify the STDcuu parameter, TF/Mirror validates the specification. If the STDcuu parameter does not reference the correct R1, TF/Mirror issues the BCVI120W message (or BCVI120E, depending on the MAXRC parameter value of the GLOBAL command) and substitutes the correct R1.

**TIMEOUT (nn)**

Sets the timeout (in seconds) for the SPLIT operation.

The default value is 15 seconds.

When both the GLOBAL and the SPLIT timeouts are set, the most restrictive (lowest) value is used. For example, if the GLOBAL timeout is set to 15 and the SPLIT timeout is set to 30, TF/Mirror uses the value of 15. If the GLOBAL timeout is set to 22 and the SPLIT timeout is set to 30, TF/Mirror uses the value of 22.
cuu
A device identified with its CUU.

For an RMT operation, if you do not specify the SRDF group (srdfgrp), then cuu must be an R1 device in the Synchronous mode.

If the R1 device is in semi-synchronous or adaptive copy mode, TF/Mirror issues the BCV1119W message (or BCV1119E, depending on the MAXRC parameter value of the GLOBAL command).

cuus
A BCV identified with its CUU.

cuus-cuus
A range of BCVs identified with their CUUs. All devices in the range must reside in the same storage system. The number of devices in cuus-cuus and cuup-cuup ranges must be the same.

FORCE | NOFORCE
Allows to force a SPLIT of a STD/BCV pair when STD/BCV invalid tracks exist.

- FORCE—The pair can be split.
- (Default) NOFORCE—The pair cannot be split.

HOLD
Places a hold on the BCV, thus making it unavailable for TF/Mirror commands from all hosts.

INSTant({Yes | No})
By default, all split commands for a range of devices and SPLITs with the same sequence number are processed as multi-instant split (in one I/O operation). If a multi-instant split is not needed, specify INSTant(No).

The default value for this parameter can be predefined using site options.

LCL
Specifies that the devices are on the local side of an SRDF configuration.

NR
Makes the BCV Not Ready after the split.

In this state, the BCV cannot be brought online or accessed. This prevents duplicate volser at IPL if you do not relabel the device after the split.

PROTECTEDRESTORE({Yes | No})

IMPORTANT
The PROTECTEDRESTORE parameter is ignored in the clone emulation mode. All operations are processed using the Protected RESTORE feature. This means that PROTECTEDRESTORE(Y) is not required in the SPLIT command after a RESTORE operation performed in the clone emulation mode.
Enables the Protected RESTORE feature.

**Note:** “Protected RESTORE” on page 49 describes the Protected RESTORE feature.

This parameter is required for a SPLIT command to succeed after a Protected RESTORE if you set the GLOBAL PROTECTEDRESTORE parameter to REQUEST or did not specify the GLOBAL PROTECTEDRESTORE parameter.

Aliases for PROTECTEDRESTORE include: PROTRESTORE, PROTRSTR, PRSTR, PROT.

**R2Sync** (Yes | No)

Enables or disables synchronization of data from the R1 BCV to the R2 device after the split. The default value is Yes.

- If you specify Yes, the SRDF link is resumed.
- If you specify No, the SRDF link remains suspended.

The default value for this parameter can be predefined using site options.

**RMT**

Specifies that the devices are on the remote side of an SRDF configuration.

**seq#**

A 1-to-128 sequence number that defines the command execution order. Commands with equal sequence numbers are executed in parallel.

**srdfgrp**

The SRDF group or the hop list for the RMT operation.

Each SRDF group is represented by a one- or two-digit value. You can specify a hop list of up to four SRDF groups separated by periods, for example:

nn.nn.nn.nn

**IMPORTANT**

You can specify up to 2 hops for remote Consistent SPLITs.

**symdv#bcv**

A BCV identified with its PowerMax/VMAX device number.

**symdv#bcv-symdv#bcv**

A range of BCVs identified with their PowerMax/VMAX device numbers.

**VOLID**(volser, [E])

Changes the volser of the BCV after the pair is split. The E option also updates the VVDS, index VTOC, and DSCB information to reflect the new volser.

VOLID is allowed only for a single-CUU SPLIT when the WAIT parameter is specified in the SPLIT command or is implied by the GLOBAL command.
The VOLID parameter is invalid for FBA cuus.

**Note:** In the clone emulation mode, you can use the NOWAIT parameter with the VOLID parameter.

**VXWAIT** *(Yes | No)*

**Note:** The VXWAIT parameter is supported with PowerMaxOS 5978 and HYPERMAX OS 5977. Under Enginuity 5876 and 5773, the VXWAIT parameter is ignored.

The VXWAIT parameter fine-tunes TF/Mirror behavior under PowerMaxOS 5978 and HYPERMAX OS 5977. For SPLIT commands, it determines whether TF/Mirror allows a SPLIT when the STD and BCV pair is not synchronized.

- Yes—SPLIT is not allowed.
- No—SPLIT is allowed.

**IMPORTANT**

With PowerMaxOS 5978 and HYPERMAX OS 5977, TF/Mirror waits for the synchronization to complete only when the WAIT|NOWAIT parameter is set to WAIT and the VXWAIT parameter is set to Yes. In all other cases, TF/Mirror does not wait for completion.

This parameter can be used to override the settings made with the GLOBAL command. The default value for this parameter can be predefined using site options.

**WAIT** *(<nnnn>) | NOWAIT*

Determines whether to wait for the action to complete:

- **WAIT**—Wait for the action to complete.
- **NOWAIT**—Pass the command to the storage system and assume that the action is complete.
- **nnnn**—Wait for <nnnn> minutes and assume that the action is complete. For example, WAIT(10) means a 10-minute wait. A maximum of up to 9999 minutes can be specified.

**Note:** If WaitSync is specified, the time *(nnnn)* also includes the waiting time allocated to synchronize the BCV mirror.

**IMPORTANT**

With PowerMaxOS 5978 and HYPERMAX OS 5977, TF/Mirror waits for the synchronization to complete only when the WAIT|NOWAIT parameter is set to WAIT and the VXWAIT parameter is set to Yes. In all other cases, TF/Mirror does not wait for completion.
This parameter can be used to override the settings made with the GLOBAL command. The default value for this parameter can be predefined using site options.

\[ \text{WaitSync}\{\text{WTO}([\text{Yes}|\text{No}]})\]  
Waits for the BCV mirror positions (M2, M3, or M4) to synchronize (M1-M4).

The optional parameter WTO issues a WTO to route code 11 specifying the start and stop points. The BCVM060I message indicates the start of the synchronization. The BCVM061I message indicates the end of the synchronization.

When the WaitSync parameter is specified in the SPLIT command, the waiting time is determined as follows:

- If the WAIT parameter (specified in the same SPLIT command or the GLOBAL command) has a value of \( n \), the system waits \( n \) minutes for the background SPLIT action to complete and another \( n \) minutes for the synchronization to complete.
- If the WAIT parameter is specified with no value, the system waits 10 minutes for the background SPLIT action to complete and 60 minutes for the synchronization to complete.

**Examples**

1. To split the BCV identified with CUU DF21:

   ```plaintext
   SPLIT 1,DF21
   ```

   ```plaintext
   BCVI020I Start of INPUT control statement(s) from SYSIN
   BCVI018I (0001) SPLIT 1,DF21
   BCVI021I End of INPUT control statement(s) from SYSIN
   ```

   ```plaintext
   BCVM039I (0001) Process input statement
   BCVM004I INSTANT SPLIT BCV DF21
   BCVM047I All control statements processed, highest RC 0
   ```

2. To perform an Instant SPLIT of the BCVs identified with CUUs DF60-DF62 (note that the clone emulation mode is used because the devices are RAID5):

   ```plaintext
   SPLIT 1,DF60-DF62,INS(Y)
   ```

   ```plaintext
   BCVI020I Start of INPUT control statement(s) from SYSIN
   BCVI018I (0001) SPLIT 1,DF60-DF62,INS(Y)
   BCVI021I End of INPUT control statement(s) from SYSIN
   ```

   ```plaintext
   BCVM039I (0001) Process input statement
   BCVM004I * MULTI INSTANT-SPLIT BCVs:
   BCVM004I * DF60-DF62(0001)
   BCVM140I Command processed via TF/Clone emulation
   BCVM047I All control statements processed, highest RC 0
   ```
3. To split the remote BCV identified with PowerMax/VMAX device number 27C, reached through CUU C121 and SRDF group 06 (note that the clone emulation mode is used):

*SPLIT 1,RMT(C121,27C,06)*

BCVI020I Start of INPUT control statement(s) from SYSIN
BCVI018I (0001) SPLIT 1,RMT(C121,27C,06)
BCVI021I End of INPUT control statement(s) from SYSIN

BCVM039I (0001) Process input statement
BCVM004I INSTANT SPLIT REMOTE BCV SYMDEV 00027C through C121
BCVM140I Command processed via TF/Clone emulation
BCVM047I All control statements processed, highest RC 0

4. To split the BCV identified with CUU DE18 and change the volser to TST27A (note that the clone emulation mode is used):

*SPLIT 1,DE18,VOLID(TST27A)*

BCVI020I Start of INPUT control statement(s) from SYSIN
BCVI018I (0001) SPLIT 1,DE18,VOLID(TST27A)
BCVI021I End of INPUT control statement(s) from SYSIN

BCVM039I (0001) Process input statement
BCVM004I INSTANT SPLIT BCV DE18
BCVM140I Command processed via TF/Clone emulation
BCVM038I CLIP VOLID(TST27A) complete on BCV DE18
BCVM047I All control statements processed, highest RC 0

5. To split the BCV identified with CUU DE19, change the volser to TST27B and update the VTOC, IXVTOC, and VVDS information (note that the clone emulation mode is used):

*SPLIT 1,DE19,WAIT,VOLID(TST27B,E)*

BCVI020I Start of INPUT control statement(s) from SYSIN
BCVI018I (0001) SPLIT 1,DE19,WAIT,VOLID(TST27B,E)
BCVI021I End of INPUT control statement(s) from SYSIN

BCVM039I (0001) Process input statement
BCVM004I INSTANT SPLIT BCV DE19
BCVM140I Command processed via TF/Clone emulation
BCVM038I CLIP VOLID(TST27B) complete on BCV DE19
BCVM038I VTOC, IXVTOC, and VVDS updated
BCVM047I All control statements processed, highest RC 0

6. To perform an Instant SPLIT of remote BCVs identified with PowerMax/VMAX device numbers 1FD-1FF, reached through CUU C122 and SRDF group E0:

*SPLIT 1,RMT(C122,01FD-01FF,E0),INS(Y)*

BCVI020I Start of INPUT control statement(s) from SYSIN
BCVI018I (0001) SPLIT 1,RMT(C122,0001FD-0001FF,E0),INS(Y)
BCVI021I End of INPUT control statement(s) from SYSIN

BCVM039I (0001) Process input statement
BCVM004I * MULTI INSTANT-SPLIT REMOTE BCVs through C122:
BCVM004I * 0001FD-0001FF(0001)
BCVM047I All control statements processed, highest RC 0
7. To perform an Instant SPLIT of local BCVs identified with PowerMax/VMAX device numbers 18D-18F, reached through CUU C122, where the BCV (M1) is synchronized from the BCV mirror (M2):

**SPLIT 1, LCL(C122, 18D-018F), INS(Y), BCVFRESH(Y)**

BCVI0201 Start of INPUT control statement(s) from SY SIN
BCVI0181 (0001) SPLIT 1, LCL(C122, 18D-018F), INS(Y), BCVFRESH(Y)
BCVI0211 End of INPUT control statement(s) from SY SIN

BCVM039I (0001) Process input statement
BCVM0041 * MULTI INSTANT-SPLIT LOCAL BCVs through C122:
BCVM0041 * 00018D-00018F(0001)
BCVM047I All control statements processed, highest RC 0

8. To perform a multisystem Consistent SPLIT of the remote BCV identified with PowerMax/VMAX device number 01CC, reached through gatekeeper CUU DE02 and SRDF group 06, without waiting for the action to complete (note that the clone emulation mode is used):

**SPLIT 1, RMT(DE20, 01CC, 6), CONS(GLOBAL, TIMEOUT(5), STDCUU(DF60)), NOWAIT**

BCVI0201 Start of INPUT control statement(s) from SY SIN
BCVI0181 (0001) SPLIT 1, RMT(DE20, 01CC, 6), CONS(GLOBAL, TIMEOUT(5), STDCUU(DF60)), NOWAIT
BCVI0211 End of INPUT control statement(s) from SY SIN
BCVI082I (0001) Symm 0001903-00097, ECA detected and enabled
BCVM039I (0001) Process input statement
BCVM0041 CONSISTENT-SPLIT REMOTE BCV SYMDEV 0001CC through DE20
BCVM140I Command processed via TF/Clone emulation

BCVM047I All control statements processed, highest RC 0

9. To split the local BCV identified with PowerMax/VMAX device number 01D4, reached through gatekeeper CUU DE22, without resuming the SRDF link after the split (note that the clone emulation mode is used):

**SPLIT 1, LCL(DE22, 01D4), R2SYNC(N)**

BCVI0201 Start of INPUT control statement(s) from SY SIN
BCVI0181 (0001) SPLIT 1, LCL(DE22, 01D4), R2SYNC(N)
BCVI0211 End of INPUT control statement(s) from SY SIN

BCVM039I (0001) Process input statement
BCVM0041 INSTANT SPLIT LOCAL BCV SYMDEV 0001D4 through DE22
BCVM140I Command processed via TF/Clone emulation

BCVM047I All control statements processed, highest RC 0

10. To perform an Instant SPLIT of remote BCVs identified with PowerMax/VMAX device numbers EF8-EF9, reached through gatekeeper CUU 9A10 and SRDF groups 50 and 13:

**SPLIT 7, RMT(9A10, EF8-EFA, 50.13), INS(Y)**

BCVI0201 Start of INPUT control statement(s) from SY SIN
BCVI0181 (0001) SPLIT 7, RMT(9A10, EF8-EFA, 50.13), INS(Y)
BCVI0211 End of INPUT control statement(s) from SY SIN
BCVM039I (0001) Process input statement
BCVM0041 * MULTI INSTANT-SPLIT REMOTE BCVs through 9A10:
BCVM0041 * 000EF8-000EFA(0001)
BCVM047I All control statements processed, highest RC 0
11. To perform a multisystem Consistent SPLIT of remote BCVs identified with PowerMax/VMAX device numbers 16A-16C, reached through gatekeeper CUU DE20 and SRDF group 50, using the ECA I/O control mode:

```
SPLIT 2,RMT(DE20,16A-16C,50),INS(Y),CONS(GLOBAL(ECA))
```

```
BCVI020I Start of INPUT control statement(s) from SYSIN
BCVI018I (0001) SPLIT 2,RMT(DE20,16A-16C,50),INS(Y),CONS(GLOBAL(ECA))
BCVI021I End of INPUT control statement(s) from SYSIN
BCVI082I (0001) Symm 0001903-00344, ECA detected and enabled
BCVM039I (0001) Process input statement
BCVM004I * CONSISTENT-SPLIT REMOTE BCVs through DE20:
BCVM004I * 00016A-00016C(0001)
BCVM047I All control statements processed, highest RC 0
```

12. To perform a multisystem Consistent SPLIT of remote BCVs identified with PowerMax/VMAX device numbers 5F0-5FF, reached through gatekeeper CUU 8510 and SRDF group F2, using the ECA I/O control mode (note that since SRDF/A was detected, it is suspended and resumed to ensure consistency):

```
SPLIT 1,RMT(8510,5F0-5FF,F2),INS(Y),CONS(GLOBAL(ECA))
```

```
BCVI020I Start of INPUT control statement(s) from SYSIN
BCVI018I (0001) SPLIT 1,RMT(8510,5F0-5FF,F2),INS(Y),CONS(GLOBAL(ECA))
BCVI082I (0001) Symm 0001926-00313, ECA detected and enabled
BCVI121I ECA bypassed for SRDF/A Remote Consistent Split
BCVM039I (0001) Process input statement
BCVM004I * CONSISTENT-SPLIT REMOTE BCVs through 8510:
BCVM004I * 0005F0-0005FF(0001)
BCVM140I Command processed via TF/Clone emulation
BCVM133I SRDFA Suspend successful for RA Group F2, CUU 8510, Symm 0001926-00313
BCVM133I SRDFA Resume successful for RA Group F2, CUU 8510, Symm 0001926-00313
BCVM047I All control statements processed, highest RC 0
```

13. To perform a simultaneous Consistent SPLIT of remote BCVs identified with PowerMax/VMAX device numbers 5F0-5FF and 690-69F (Concurrent SRDF/A), reached through gatekeeper CUU 8510 and SRDF groups F2 and 2F respectively (note that because MSC is not active, consistency cannot be guaranteed):

```
SPLIT 1,RMT(8510,5F0-5FF,F2),INS(Y),CONS(GLOBAL(ECA))
SPLIT 1,RMT(8510,690-69F,2F),INS(Y),CONS(GLOBAL(ECA))
```

```
BCVI020I Start of INPUT control statement(s) from SYSIN
BCVI018I (0002) SPLIT 1,RMT(8510,5F0-5FF,F2),INS(Y),CONS(GLOBAL(ECA))
BCVI018I (0003) SPLIT 1,RMT(8510,690-69F,2F),INS(Y),CONS(GLOBAL(ECA))
BCVI021I End of INPUT control statement(s) from SYSIN
BCVI082I (0002) Symm 0001926-00313, ECA detected and enabled
BCVI121I ECA bypassed for SRDF/A Remote Consistent Split
BCVM039I (0002) Process input statement
BCVM004I * CONSISTENT-SPLIT REMOTE BCVs through 8510:
BCVM004I * 0005F0-0005FF(0002)
BCVM140I Command processed via TF/Clone emulation
BCVM133I SRDFA Suspend successful for RA Group F2, CUU 8510, Symm 0001926-00313
BCVM133I SRDFA Resume successful for RA Group 2F, CUU 8510, Symm 0001926-00313
BCVM047I All control statements processed, highest RC 0
```
14. To perform a Consistent SPLIT of remote BCVs identified with PowerMax/VMAX device numbers 4E0-4EF (Cascaded R21 configuration), reached through gatekeeper CUU 8510 and SRDF groups F0 and 52 (note that since ECA is sent on the R21, the GLOBAL parameter CONS(ALLOWNONSYNC) is required; the BCVM080I messages are issued because of DEBUG(ECA)):

GLOBAL CONS(ALLOWNONSYNC), MAXRC(4, SETMAX), DEBUG(ECA)
SPLIT 1, RMT(8510, 4E0-4EF, 00.52), INS(Y), CONS(GLOBAL(ECA))

BCVI020I Start of INPUT control statement(s) from SYSIN
BCVI018I (0001) GLOBAL CONS(ALLOWNONSYNC), MAXRC(4, SETMAX), DEBUG(ECA)
BCVI018I (0002) SPLIT 1, RMT(8510, 4E0-4EF, 00.52), INS(Y), CONS(GLOBAL(ECA))
BCVI021I End of INPUT control statement(s) from SYSIN

BCVI133W (0002) BCV 04E0, Consistent Split allowed for (ADCOPY-Disk)
BCVI133W (0002) BCV 0004E1, Consistent Split allowed for (ADCOPY-Disk)
BCVI133W (0002) BCV 0004E2, Consistent Split allowed for (ADCOPY-Disk)
BCVI133W (0002) BCV 0004E3, Consistent Split allowed for (ADCOPY-Disk)
BCVI133W (0002) BCV 0004E4, Consistent Split allowed for (ADCOPY-Disk)
BCVI133W (0002) BCV 0004E5, Consistent Split allowed for (ADCOPY-Disk)
BCVI133W (0002) BCV 0004E6, Consistent Split allowed for (ADCOPY-Disk)
BCVI133W (0002) BCV 0004E7, Consistent Split allowed for (ADCOPY-Disk)
BCVI133W (0002) BCV 0004E8, Consistent Split allowed for (ADCOPY-Disk)
BCVI133W (0002) BCV 0004E9, Consistent Split allowed for (ADCOPY-Disk)
BCVI133W (0002) BCV 0004EA, Consistent Split allowed for (ADCOPY-Disk)
BCVI133W (0002) BCV 0004EB, Consistent Split allowed for (ADCOPY-Disk)
BCVI133W (0002) BCV 0004EC, Consistent Split allowed for (ADCOPY-Disk)
BCVI133W (0002) BCV 0004ED, Consistent Split allowed for (ADCOPY-Disk)
BCVI133W (0002) BCV 0004EE, Consistent Split allowed for (ADCOPY-Disk)
BCVI133W (0002) BCV 0004EF, Consistent Split allowed for (ADCOPY-Disk)

BCVM039I (0002) Process input statement
BCVM004I * CONSISTENT-SPLIT REMOTE BCVs through 8510:
BCVM004I * 0004E0-0004EF(0002)
BCVM140I Command processed via TF/Clone emulation
BCVM080I Window Set, Symm 0001926-00313, Duration 0019, Interval 0001, Run# 0001, through 8510
BCVM080I Start device 0000B2, Last 0000C1
BCVM080I Window Crl, Symm 0001926-00313, Run# 0001, through 8510
BCVM080I Start device 0000B2, Last 0000C1

BCVM047I All control statements processed, highest RC 4
USEREXIT

The USEREXIT command allows you to pass control to a user-written program during TF/Mirror processing.

The program specified is linked to using the LINK macro while passing the parameters \( p1 \) and \( p2 \).

The exits are executed on their sequence level in the order in which they are specified in SYSIN.

When returning control, the exit routine must specify the exit return code in R15. This return code is checked against the MAXRC parameter value specified in the GLOBAL command.

Syntax

USEREXIT \( \text{seq#}, \text{load-module-name}, p1, p2 \)

Parameters

\( \text{load-module-name} \)

The user-written program to which control is to be passed during TF/Mirror processing.

\( p1 \)

4-digit (2-byte) hexadecimal value.

\( p2 \)

4-digit (2-byte) hexadecimal value.

\( \text{seq#} \)

A 1-to-128 sequence number that defines the command execution order. Commands with equal sequence numbers are executed in parallel.

Example

To pass TF/Mirror control to a user-written program called WAITBKUP:

USEREXIT 1, WAITBKUP, 008A, C001
SRDF/AR commands

In addition to commands and parameters listed in this section, SRDF/AR uses the following parameters of the TF/Mirror GLOBAL command:

- AUTORElease
- DEBUG
- FASTESTablish
- LCLR1BCV
- MAXRC
- MAXGrp
- MULTIATTach

For more information about these settings, see “GLOBAL” on page 130.

ADD

Creates a new process and determines the type of SRDF/AR environment for the process.

Syntax

```
ADD seq#, process-name, HOP_TYPE({SINGLE|MULTI})
[, MAXDEV(nnnnn)]
[, MAXGrp(nnnn)]
[, MAXMSG(nnnn)]
[, SYSTEM({LOCAL[({ECA|IOSLEVEL})]|GLOBAL[(ECA)])})]
```

Parameters

- **HOP_TYPE({SINGLE|MULTI})**
  The type of the SRDF/AR environment for the process:
  - SINGLE—A single-hop environment.
  - MULTI—A multihop environment.
  This is a required parameter with no default value.

  **Note:** Information about specific devices is provided in the MODIFY DEFINE command, as described in “MODIFY DEFINE” on page 175.

- **MAXDEV(nnnnn)**
  The maximum number of SRDF/AR devices per a storage system. Valid values of nnnnn are from 1 to 64000. The default value is 8192.

- **MAXGrp(nnnn)**
  The maximum number of SRDF/AR groups. Valid values of nnnn are from 1 to 9999. The default value is 64.

- **MAXMsg(nnnn)**
  The maximum number of SRDF/AR message buffers. Valid values of nnnn are from 1 to 9999. The default value is 256.
process-name

The name of the process. The value cannot exceed 20 characters.

If a process with the specified name already exists, the statement fails.

seq#

A 1-to-128 sequence number that defines the command execution order.
Commands with equal sequence numbers are executed in parallel.

SYSTEM({LOCAL[{ECA, IOSLEVEL}] } | GLOBAL[(ECA)] })

Whether the SRDF/AR environment includes one or more LPARs:

- LOCAL—A single-LPAR environment.
- GLOBAL—A multi-LPAR environment.

You can also set the I/O control mode to ECA or IOSLEVEL.

Note: For information about ECA and IOSLEVEL modes, see the Consistency Groups for z/OS Product Guide.

DELETE

Removes the process control blocks from the system.

Before using the DELETE command, stop the process with the STOP command, as described in “MODIFY STOP” on page 189. Otherwise the DELETE command fails.

Syntax

DELETE seq#, process-name[,FORCE]

Parameters

FORCE

Forces deletion of a process when the process cannot be stopped using the MODIFY STOP command.

Issue the DELETE command with the FORCE parameter while the STOP command has not finished.

process-name

The name of the process. The value cannot exceed 20 characters.

If a process with the specified name does not exist, the statement fails.

seq#

A 1-to-128 sequence number that defines the command execution order.
Commands with equal sequence numbers are executed in parallel.
MODIFY DEFINE

Defines process attributes and devices.

The number of DEFINE statements for a process is unlimited. If multiple storage systems are needed to create a restartable remote image, specify at least one DEFINE action for each storage system.

**Note:** To change the configuration, you have to delete and then redefine the required configuration.

For multiple MODIFY DEFINE statements, the last setting of each parameter except DEVice_List is used.

Multiple MODIFY DEFINE statements are allowed at a single sequence level. Ensure that all ADD and DELETE commands are executed prior to issuing a MODIFY DEFINE command.

**Syntax**

```
MODIFY seq#,process-name,DEFINE,
DEVice_List(
 {SRCR1BCV(sbcv_cuu_list|sbcv_symdv#_list)|
 LCLR1BCV(sbcv_symdv#_list)},
 {SRCSTD(sstd_cuu_list[,srdfgrp|gatekeeper_cuu)]|
 LCLSTD(sstd_symdv#/list[,srdfgrp|gatekeeper_cuu])},
 TGTBCV(tbcv_symdv#/list))
[,BCVSTATE({READY|NR})]
[,BYPassTargetBCVNotReady({Yes|No})]
[,BYPTBCV({Yes|No})]
[,BYPTONL({Yes|No})]
[,CYCle(cycle-time[,nmmm])]
[,CYCle_OverFlow({IMMEDIATE|NEXT})]
[,DEBUG([STATus],[STATusE],[ITRks],[IOSLevel],[DISPCmd],[ECA])]
[,ITRKSYNC([nmm],[mmm])]
[,MAXDEV(nnnnn)]
[,MSGOPT({BUFFERed|SYSOUT(ddname)|WTO([nn])})]
[,POLICY(Pn[,...])]
[,REEST(count,initial_wait,interval)]
[,REEST_TGT(count,initial_wait,interval)]
[,RESUME(count,initial_wait,interval)]
[,TARGETBCVMIRRORSYNCATEND({PROCESS|CYCLE|NO})]
[,TIMEOUT(time,[{CONTinue|TERminate({IMMEDIATE|CYCLE})}])]
```
Parameters

BCVSTATE ( {READY | NR} )

Note: The BCVSTATE parameter is obsolete and ignored.

Controls the status of the R1 BCVs after Step 2: Consistent SPLIT of R1 BCVs.

READY

(Default) The R1 BCVs are made Ready.

NR

The R1 BCVs remain in the NR_BCV status. They will be made Ready at Step 4: Check if R1 BCVs are Ready.

BYPassTargetBCVNotReady ( {Yes | No} )

Whether to leave BCVs paired to R2 STDs in the Not Ready state after Step 9: SPLIT BCVs paired to R2 STDs.

Note: This parameter applies to clone emulation BCVs only.

No

(Default) The BCVs are not left in the Not Ready state.

Yes

The BCVs are left in the Not Ready state.

BYPTBCV ( {Yes | No} )

Whether to bypass the following steps in the SRDF/AR cycle:

- Step 1B: Ensure BCVs paired to R2 STDs are offline
- Step 6: Wait for mirror sync on BCVs paired to R2 STDs
- Step 7: RE-ESTABLISH BCVs paired to R2 STDs
- Step 8: Wait for TF/Mirror sync of R2 STDs
- Step 9: SPLIT BCVs paired to R2 STDs
- Step 10b: Wait for mirror sync on BCVs paired to R2 STDs
- Step 11b: Wait for mirror sync on BCVs paired to R2 STDs

Yes

The RE-ESTABLISH and SPLIT operations are not performed.

No

(Default) The RE-ESTABLISH and SPLIT operations are performed.

Specifying BYPTBCV(Y) in the initial definition of a SRDF/AR process means there are no BCVs paired to R2 STDs in the SRDF/AR configuration. This is SRDF/AR configuration choice made by the installation. In this case, it is not possible to start a process using BYPTBCV(N).
BYPTONL({Yes|No})

Whether to bypass the online/offline status check for BCVs paired to R2 STDs. The default value is No.

CYCle(cycle-time[,nnnn])

Sets the cycle time and optionally the number of cycles to perform.

cycle-time

The period to start the cycle of the process (for example, every 4 hours) in the hh:mm:ss format.

Note: If the actual cycle duration exceeds cycle-time, the CYCle_OverFlow parameter determines the system behavior.

nnnn

The number of cycles to perform (for example, 3 cycles). Valid values are from 0 to 9999. If \( n \) is zero (0) or not specified, the process runs until a STOP command is issued.

CYCle_OverFlow({IMMEDiate|NEXT})

Determines the system behavior when the actual cycle duration exceeds the cycle time set with the CYCle parameter.

IMMEDiate

(Default) A new cycle is started immediately after the current cycle ends. The start time of the new cycle is the basis for the next calculated cycle duration.

NEXT

The system waits until the next calculated cycle time to start a new cycle.

DEBUG([STATus][,STATusE][,ITRks][,IOSLevel][,DISPCmd][,ECA])

Enables the debugging mode.

STATus

Displays SRDF/AR status information.

STATusE

Displays SRDF/AR status details.

InvalidTRacks

Displays invalid tracks.

IOSLevel

Displays held IOS level information.

DISPlayCommand

Displays the SRDF/AR statement used to define all current processes.

ECA

Displays ECA information.
DEVice_List(
{SRCR1BCV(sbcv_cuu_list|sbcv_symdv#_list)|
LCLR1BCV(sbcv_symdv#_list)),
{SRCSTD(sstd_cuu_list|[srdfgrp],gatekeeper_cuu)|
LCLSTD(sstd_symdv#_list,[srdfgrp],gatekeeper_cuu)),
TGTBCV(tbcv_symdv#_list)
)

Defines devices to be used in the process:

- The R1 BCVs
- The STDs paired to R1 BCVs
- The BCVs paired to R2 STDs

**Note:** Device lists may not span storage systems. With multiple storage systems, an individual DEFINE DEVice_List statement for each storage system is required.

SRCR1BCV(sbcv_cuu_list|sbcv_symdv#_list)|LCLR1BCV(sbcv_symdv#_list)

Identifies R1 BCVs with their CUUs or PowerMax/VMAX device numbers. If the LCLR1BCV parameter of the TF/Mirror GLOBAL command is set, PowerMax/VMAX device numbers are required.

You can specify a single device or a range of devices using a hyphen. If you specify a range, All devices in the range must reside in the same storage system.

**Note:** For multihop configurations, SRCR1BCV identifies BCVs at Site B.

(SRCSTD(sstd_cuu_list|[srdfgrp],gatekeeper_cuu)|
LCLSTD(sstd_symdv#_list,[srdfgrp],gatekeeper_cuu)),

Identifies STDs paired with R1 BCVs with their CUUs or PowerMax/VMAX device numbers.

You can specify a single device or a range of devices using a hyphen. If you specify a range, All devices in the range must reside in the same storage system.

**Note:** For multihop configurations, SRCSTD or LCLSTD identify R1 devices at the production Site A.

gatekeeper_cuu

A gatekeeper device in the production storage system identified with its CUU.

**Note:** A device from sstd_cuu_list cannot be specified as gatekeeper_cuu when IOSLEVEL is requested.

srdfgroup

The SRDF group to be used to route requests to R2 STDs and BCVs paired to them.
By default, the SRDF group of R1 BCVs is used. This parameter overrides the
default setting.

In multihop configurations, the srdfgroup value is ignored, and the default
SRDF group is used.

_TGTBCV(tbcv_symdv#_list)_

Identifies BCVs paired to R2 STDs with their PowerMax/VMAX device numbers.
You can specify a single device or a range of devices using a hyphen. If you
specify a range, All devices in the range must reside in the same storage
system.

**Note:** For multihop configurations, TGTBCV identifies BCVs at Site C.

_INTRKSYNC([nnn][,mmm])_

Sets limits for the invalid track check before a pause is issued. This parameter is
optional.

_\_nnn_

The number of times to perform an invalid track check. Valid values are from 1
to 999. The default value is 60.

_\_mmm_

The number of successive checks on which an invalid track is detected before
issuing a Verify. If a Verify does not resolve the situation, SRDF/AR issues a
pause. Valid values are from 1 to 999.

**Note:** When the invalid track count is not one for each BCV with any invalid
tracks, the value of the normal parameter is used.

The default value for this parameter can be predefined using site options.

_MAXDEV(nnnnnn)_

See “MAXDEV(nnnnn)” on page 173.

_MESSAGEOPT({BUFFERED|SYSOUT(ddname)|WTO([nn])})_

Sets message handling options.

_BUFFERED_

_(Default) Holds messages in the memory buffer._

**Note:** You can display the messages using the MODIFY QUERY(MSG)
statement.

_SystemOUT(ddname)_

Routes messages to SYSOUT identified by the _ddname_ of the process.

**Note:** You specify the process DD name in the SCF PROC at startup.
WTO\((nn)\)  
Issues messages as WTO (Write To Operator) messages using the route code \(nn\). Valid values are from 1 to 28. If \(nn\) is not specified, the default value of 11 is used.

**Note:** You can specify both WTO\((nn)\) and SYSOUT at the same time.

POLICY\(Pn[,...]\)  
Applies policies to the process.

- With a single-hop configuration: POLICY can be used at Step 5: RE-ESTABLISH R1 BCVs, if the STDs are not SRDF, or if SRDF, Static SRDF and unprotected.
- In all other cases: the STD is an SRDF device, so it must be Static SRDF and unprotected.

\(n\)  
The policy number. Valid policy numbers are as follows:

- 1—Wait for drive replacement (pause).
- 2—Proceed with resiliency.
- 3—Do not tolerate a mirror failure for a Protected BCV.
- 4—Tolerate mirror failures for a Protected BCV.

**Note:** P4 can make a huge difference in the resiliency of the SRDF/AR process.

- 5—Wait for BCV mirror synchronization.
- 6—Do not wait for BCV mirror synchronization.
- 7—Skip Step 3: Wait for SRDF sync of R1s with R2s in case of a drive failure at Step 2: Consistent SPLIT of R1 BCVs.
- 8—Use Protected BCV ESTABLISH.
- 9—No R2 replacement.
- 10—Allow R2 replacement.
- 11—Target BCVs only.
- 12—N/A
- 13—Protected R2s must always be protected.
- 14—Protected R2s can run unprotected. (Tolerate a mirror failure.)
- 15—N/A
- 16—N/A

The default values are 1, 3, 5, 9, and 13.

The default values for this parameter can be predefined using site options.

**process-name**  
The name of the process. The value cannot exceed 20 characters.

If a process with the specified name does not exist, the statement fails.
**REEST**(\(\text{count, initial\_wait, interval}\))

Controls issuing concurrent RE-ESTABLISH requests to the storage system where R1 devices reside.

\(\text{count}\)

The number of concurrent RE-ESTABLISH requests. Valid values are from 0 to 65535. The default value is 8. A \text{count} of 0 disables the option.

\(\text{initial\_wait}\)

The time (in seconds) to wait prior to issuing RE-ESTABLISH requests. Valid values are from 0 to 999. The default value is 5.

\(\text{interval}\)

The time (in seconds) to wait after issuing each RE-ESTABLISH request. Valid values are from 1 to 999. The default value is 1.

After \(\text{count}\) RE-ESTABLISH commands have been issued, a wait for \(\text{interval}\) occurs and the status of each of the devices just processed is checked. Once attached, the next \(\text{count}\) devices is processed in the same way until all of the devices have been processed successfully.

The default value for this parameter can be predefined using site options.

**REEST\_TGT**(\(\text{count, initial\_wait, interval}\))

Controls issuing concurrent RE-ESTABLISH requests to the storage system where R2 devices reside.

\(\text{count}\)

The number of concurrent RE-ESTABLISH requests. Valid values are from 0 to 65535. The default value is 8. A \text{count} of 0 disables the option.

\(\text{initial\_wait}\)

The time (in seconds) to wait prior to issuing RE-ESTABLISH requests. Valid values are from 0 to 999. The default value is 5.

\(\text{interval}\)

The time (in seconds) to wait after issuing each RE-ESTABLISH request. Valid values are from 1 to 999. The default value is 1.

After \(\text{count}\) RE-ESTABLISH commands have been issued, a wait for \(\text{interval}\) occurs and the status of each of the devices just processed is checked. Once attached, the next \(\text{count}\) devices is processed in the same way until all of the devices have been processed successfully.

The default value for this parameter can be predefined using site options.

**RESUME**(\(\text{count, initial\_wait, interval}\))

Controls issuing concurrent RESUME requests to the storage system where R1 BCVs reside. The RESUME requests are issued at Step 2: Consistent SPLIT of R1 BCVs to propagate the data from R1 BCVs to R2 STDs.
count

The number of concurrent RESUME requests. Valid values are from 0 to 65535. The default value is 0.

initial_wait

The time (in seconds) to wait prior to issuing RESUME requests. Valid values are from 1 to 999. The default value is 30.

interval

The time (in seconds) to wait before issuing the next count of RESUME requests. Valid values are from 1 to 999. The default value is 1.

After all the Consistent SPLIT commands have been issued, a wait of initial_wait seconds occurs. The RESUME requests are then issued in groups of count with a wait of interval seconds for each next group.

The maximum time waiting for the attached state is 10 times the interval_wait. If this is exceeded, the cycle stops with the appropriate messages.

The default value for this parameter can be predefined using site options.

seq#

A 1-to-128 sequence number that defines the command execution order. Commands with equal sequence numbers are executed in parallel.

TARGETBCVMIRRORSYNCATEND({PROCESS | CYCLE | NO})

Determines when to run the mirror synchronization check for the BCVs paired to R2 STDs. The purpose is to detect a drive failure affecting a BCV mirror before the end of a cycle or process.

PROCESS

(Default) Run the check before termination of the SRDF/AR process (as Step 11b: Wait for mirror sync on BCVs paired to R2 STDs).

CYCLE

Run the check at the end of every cycle (as Step 10b: Wait for mirror sync on BCVs paired to R2 STDs).

NO

Do not run the check (neither as Step 10b nor as Step 11b).

Additionally, the mirror status check is run as Step 0 when the first cycle starts executing.

Note: You can abbreviate this parameter as TGTBCVMSYNCATEND or TGTBCVMSE.

TIMEOUT(time[, {CONTinue|TERMinate({IMMEDiate|CYCLE})}])

Sets the I/O processing timeout during Step 2: Consistent SPLIT of R1 BCVs and the timeout action.

time

The maximum duration of time (in seconds) to hold I/O processing. Valid values are from 1 to 255. The default value is 15 seconds.
CONTinue

Continues the process based on the MAXRC parameter value in the TF/Mirror GLOBAL command.

TERMinate

(Default) Terminates the process:

IMMEDIATE

(Defa ult) The process is terminated immediately.

CYCLE

The process is terminated at the end of the current cycle.

If the specified time is exceeded, no operations are performed on the BCVs paired with R2 STDs. The BCVs contain consistent images of the last successfully completed cycle. RESTART is not available under this condition. The process must have a new START action issued.

The default value for this parameter can be predefined using site options.

MODIFY EXPORT

Exports SRDF/AR syntax of a defined process (the internal control blocks) to SYSOUT or a preallocated dataset (sequential, PDS member, or GDG). The result can be used to redefine the configuration.

The generated syntax has some differences from typical SRDF/AR syntax. For example, all device specifications use PowerMax/VMAX device numbers.

The EXPORT command must be issued only on the host where the SRDF/AR task is started.

Ensure that all ADD and DELETE commands are executed prior to issuing a MODIFY EXPORT command.

Syntax

MODIFY seq#, process-name, EXPORT (SYNTAX (ddname[, {RANGE | NORANGE} ]))

Parameters

ddname

The DD name that specifies SYSOUT or a preallocated dataset for the exported syntax.

process-name

The name of the process. The value cannot exceed 20 characters.

If a process with the specified name does not exist, the statement fails.

NORANGE

Places individual MODIFY DEFINE statements for each set of devices when generating the syntax.

RANGE

(Defa ult) Forces use of device ranges, whenever possible, when generating the syntax.
**seq#**

A 1-to-128 sequence number that defines the command execution order. Commands with equal sequence numbers are executed in parallel.

**MODIFY PAUSE**

Pauses the process and issues the BCVA058A message.

*Note:* You can also use the SAR,PAUSE command of ResourcePak Base to pause a process. For more information, see the ResourcePak Base for z/OS Product Guide.

If the *step#* value is specified, the pause occurs before the next API call for the step. When specified without the *step#* value, the pause occurs just before the next API call.

Pause is deferred during a Consistent SPLIT to prevent loss of consistency for the affected cycle. For example, a PAUSE issued during **Step 2: Consistent SPLIT of R1 BCVs** is deferred until the Consistent SPLIT processing has completed.

Ensure that all ADD and DELETE commands are executed prior to issuing a MODIFY PAUSE command.

**Syntax**

```
MODIFY seq#,process-name,PAUSE[(step#)]
```

**Parameters**

*process-name*

The name of the process. The value cannot exceed 20 characters.

If a process with the specified name does not exist, the statement fails.

*seq#*

A 1-to-128 sequence number that defines the command execution order. Commands with equal sequence numbers are executed in parallel.

*step#*

The number of the step at which to pause the process.
MODIFY QUERY

Displays information for the process. The process can be active or stopped.

Multiple QUERY actions are allowed on a single sequence level. Ensure that all ADD
and DELETE commands are executed prior to issuing a MODIFY QUERY command.

Syntax

MODIFY seq#,process-name,QUERY({STATUS|DEVICES|MSG})

Parameters

DEVICES
Display devices defined for the process.

MSG
Lists the message buffers associated with the process. After a MSG display, the
message queue is emptied.

process-name
The name of the process. The value cannot exceed 20 characters.

If a process with the specified name does not exist, the statement fails.

seq#
A 1-to-128 sequence number that defines the command execution order.
Commands with equal sequence numbers are executed in parallel.

STATUS
Displays process status details, such as cycle step, cycles performed, changed
track accumulation, current settings, and so on.
MODIFY RESTART

Restarts a stopped process.

The restart point is determined by the previous STOP command. If the STOP command contained the FORCE keyword, restart may not be possible.

Only one RESTART action is allowed on a single sequence. Ensure that all ADD and DELETE commands are executed prior to issuing a MODIFY RESTART command.

**Note:** You can also use the SAR,RESTART command of ResourcePak Base to restart a process. For more information, see the ResourcePak Base for z/OS Product Guide.

Syntax

```
MODIFY seq#,process-name,RESTART[,WAIT]
```

Parameters

- **process-name**
  The name of the process. The value cannot exceed 20 characters.
  If a process with the specified name does not exist, the statement fails.

- **seq#**
  A 1-to-128 sequence number that defines the command execution order.
  Commands with equal sequence numbers are executed in parallel.

- **WAIT**
  Waits for completion of the MODIFY RESTART command while the SRDF/AR process is active.
MODIFY START

Starts the process at the beginning of the cycle.

Only one START action is allowed at a sequence level. Ensure that all ADD and DELETE commands are executed prior to issuing a MODIFY START command.

**Note:** You can also use the SAR,START command of ResourcePak Base to start a process. For more information, see the *ResourcePak Base for z/OS Product Guide.*

You can use MODIFY START to override the settings made with previous MODIFY DEFINE statements.

**Syntax**

```
MODIFY seq#,process-name,START,
[,BCVSTATE({READY|NR})]
[,BYPTBCV({Yes|No})]
[,BYPTONL({Yes|No})]
[,CYCle(cycle-time[,nnnn])]
[,CYCle_OverFlow({IMMED|NEXT})]
[,DEBUG([STATus][,STATusE][,ITRks][,IOSLevel][,DISPCmd][,ECA])]
[,MAXDEV(nnnnn)]
[,MSGOPT({BUFFERed|SYSOUT(ddname)|WTO([nn]))}]
[,REEST(count,initial_wait,interval)]
[,REEST_TGT(count,initial_wait,interval)]
[,TIMEOUT(time,{CONTinue|TERMinate({IMMEDiate|CYCLE})})]
[,RESUME(count,initial_wait,interval)]
[,WAIT]
```

**Parameters**

- **BCVSTATE({READY|NR})**
  
  See “BCVSTATE({READY|NR})” on page 176.

- **BYPTBCV({Yes|No})**
  
  See “BYPTBCV({Yes|No})” on page 176.

- **BYPTONL({Yes|No})**
  
  See “BYPTONL({Yes|No})” on page 177.

- **CYCle(cycle-time[,nnnn])**
  
  See “CYCle(cycle-time[,nnnn])” on page 177.

- **CYCle_OverFlow({IMMED|NEXT})**
  
  See “CYCle_OverFlow({IMMEDiate|NEXT})” on page 177.
DEBUG([STATus][,STATusE][,ITRks][,IOSLevel][,DISPCmd][,ECA])
   See “DEBUG([STATus][,STATusE][,ITRks][,IOSLevel][,DISPCmd][,ECA])” on page 177.

MAXDEV(nnnnn)
   See “MAXDEV(nnnnn)” on page 173.

MSGOPT({BUFFERed|SYSOUT(ddname)|WTO[(nn)]})
   See “MSGOPT({BUFFERed|SYSOUT(ddname)|WTO[(nn)]})” on page 179.

process-name
   The name of the process. The value cannot exceed 20 characters.
   If a process with the specified name does not exist, the statement fails.

REEST(count,initial_wait,interval)
   See “REEST(count,initial_wait,interval)” on page 181.

REEST_TGT(count,initial_wait,interval)
   See “REEST_TGT(count,initial_wait,interval)” on page 181.

seq#
   A 1-to-128 sequence number that defines the command execution order.
   Commands with equal sequence numbers are executed in parallel.

TIMEOUT(time,[CONTinue|TERMinate([IMMEDiate|CYCLE])])
   See “TIMEOUT(time,[CONTinue|TERMinate([IMMEDiate|CYCLE])])” on page 182.

RESUME(count,initial_wait,interval)
   See “RESUME(count,initial_wait,interval)” on page 181.

WAIT
   See “WAIT” on page 186.
MODIFY STOP

Stops the process.

When specified without additional parameters, the process is stopped when the current cycle step completes.

Only one STOP action is allowed at a sequence level. Ensure that all ADD and DELETE commands are executed prior to issuing a MODIFY STOP command.

**Note:** You can also use the SAR,STOP command of ResourcePak Base to stop a process. For more information, see the *ResourcePak Base for z/OS Product Guide*.

**Syntax**

```
MODIFY seq#,process-name, STOP[({NORMAL|FORCE|IMMEDiate|STEP[(step#)]})]
```

**Parameters**

**FORCE**

The process is stopped immediately, regardless of the cycle state.

**IMMEDiate**

The process is stopped immediately. The process must be active for the STOP to take effect.

**NORMAL**

*(Default)* The process is stopped at the normal end of the cycle.

**process-name**

The name of the process. The value cannot exceed 20 characters.

If a process with the specified name does not exist, the statement fails.

**seq#**

A 1-to-128 sequence number that defines the command execution order. Commands with equal sequence numbers are executed in parallel.

**STEP[(step#)]**

Stops the process at the current step or at the step specified as `step#`, providing that the step has not yet occurred in the SRDF/AR cycle. Otherwise, the SRDF/AR cycle stops at the end of the current cycle.

For example:

- If you specify STOP(STEP(03)) and SRDF/AR is at step 2 in the SRDF/AR cycle, SRDF/AR stops at the end of step 3.
- If you specify STOP(STEP(03)) and SRDF/AR is at step 4 or higher in the SRDF/AR cycle, SRDF/AR stops at the end of the current cycle.
CHAPTER 7
Message and Error Codes

This chapter covers the following topics:

- Message codes ................................................................. 192
- User abend codes ............................................................. 192
- DOIO error codes ............................................................. 193
- TF/Mirror reason codes ...................................................... 193
- EXTENTS reason codes ...................................................... 198
Message and Error Codes

Message codes

TF/Mirror messages are described in *Mainframe Enablers Message Guide*.

The messages you receive from TF/Mirror can be returned with an informational suffix (I), with a warning suffix (W), or with an error suffix (E). The message class you receive with these messages depends on the return code encountered. The values that can be returned as the job step return code range from zero (0) to eight (8). Zero would be the least serious; for example, informational messages. Eight would be an error.

You can also receive a return code of 12. A return code of 12 a serious error, but is returned with the error suffix E.

The TF/Mirror message prefixes are listed in Table 32.

**Table 32** Message prefixes

<table>
<thead>
<tr>
<th>Prefix</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BCVA</td>
<td>SRDF/AR (SAR) messages</td>
</tr>
<tr>
<td>BCVG</td>
<td>BCV group messages</td>
</tr>
<tr>
<td>BCVI</td>
<td>Initialization messages (from the TFINIT module)</td>
</tr>
<tr>
<td>BCVM</td>
<td>Mainline messages (from the EMCTF module)</td>
</tr>
<tr>
<td>BCVX</td>
<td>Exit messages (from the EMCTFX01 module)</td>
</tr>
</tbody>
</table>

User abend codes

**Table 33** lists user abend codes that are issued if an error occurs before the message system is initialized.

**Table 33** Abend codes

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>U0001</td>
<td>The SYSOUT DD statement was not specified in the JCL.</td>
</tr>
<tr>
<td>U0002</td>
<td>The OPEN request failed for the SYSOUT DD statement.</td>
</tr>
<tr>
<td>U0806</td>
<td>SCF address space not found.</td>
</tr>
</tbody>
</table>
DOIO error codes

Figure 33 shows the format for DOIO error codes.

Figure 33  DOIO error code format

TF/Mirror reason codes

PowerMaxOS 5978, HYPERMAX OS 5977, Enginuity 5876
clone emulation

The hexadecimal reason code (yy) is converted to a decimal number (xxx) and contained in the appropriate EQCAxxxE message, indicated by the BCVM144I message.

For example:

BCVM013E RESTORE failed on BCV xxxxxx, reason code yy
BCVM144I - REFER TO EQCAxxxE JOBLOG MESSAGE

Enginuity 5773

Table 34 describes TF/Mirror reason codes.

Table 34  TF/Mirror reason codes (page 1 of 5)

<table>
<thead>
<tr>
<th>Error code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>The STD does not exist.</td>
</tr>
<tr>
<td>02</td>
<td>The STD is configured as a BCV.</td>
</tr>
<tr>
<td>03</td>
<td>For an attach operation: The STD already has a BCV or a spare mirror. For a detach operation: The STD does not have an active BCV mirror.</td>
</tr>
<tr>
<td>04</td>
<td>The STD has four active mirrors.</td>
</tr>
<tr>
<td>05</td>
<td>The BCV is not the device which initiated the ESTABLISH command.</td>
</tr>
<tr>
<td>06</td>
<td>The device specified as a BCV in the call is not a BCV.</td>
</tr>
<tr>
<td>07</td>
<td>The BCV is already established to another STD.</td>
</tr>
<tr>
<td>07A</td>
<td>This version of TF/Mirror is not supported with the current level of the operating environment. Use TF/Mirror 5.6 with Level Set 1 or a higher version of TF/Mirror.</td>
</tr>
<tr>
<td>09</td>
<td>The STDs and BCVs are not the same size.</td>
</tr>
<tr>
<td>0A</td>
<td>Invalid request options.</td>
</tr>
<tr>
<td>0B</td>
<td>The STDs and BCVs are not of the same emulation type.</td>
</tr>
</tbody>
</table>
### Table 34 TF/Mirror reason codes (page 2 of 5)

<table>
<thead>
<tr>
<th>Error code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0D</td>
<td>The STD is not ready or is write-disabled.</td>
</tr>
<tr>
<td>0E</td>
<td>The command attempted to establish a ready BCV paired to R2 STD.</td>
</tr>
<tr>
<td>0F</td>
<td>This BCV was not previously attached to this STD, cannot re-establish.</td>
</tr>
<tr>
<td>10</td>
<td>ESTABLISH in progress. Poll later for ending status.</td>
</tr>
<tr>
<td>11</td>
<td>The STD is busy. Reissue the SPLIT/RESTORE command.</td>
</tr>
<tr>
<td>13</td>
<td>The STDs and BCVs are the same device.</td>
</tr>
<tr>
<td>14</td>
<td>Invalid storage system type.</td>
</tr>
<tr>
<td>15</td>
<td>The STD has open concurrent copy sessions.</td>
</tr>
<tr>
<td>16</td>
<td>The BCV has open concurrent copy sessions.</td>
</tr>
<tr>
<td>17</td>
<td>The BCV is not in a ready state or is write-disabled, or (on a SPLIT) the BCV mirror is not ready.</td>
</tr>
<tr>
<td>1A</td>
<td>The BCV does not have enough available dynamic mirrors to perform the requested ESTABLISH operation.</td>
</tr>
<tr>
<td>1B</td>
<td>The STD already has two attached dynamic mirrors (such as BCV, hot spare, or dynamic SRDF mirror). Operating environment limits the number of dynamic mirrors to two.</td>
</tr>
<tr>
<td>1C</td>
<td>Regular SPLIT not supported.</td>
</tr>
<tr>
<td>1D</td>
<td>SYSTEM_TIME_OVERRUN - no resources to process request, try later.</td>
</tr>
<tr>
<td>20</td>
<td>Internal error.</td>
</tr>
<tr>
<td>21</td>
<td>The STD and BCVs do not comprise an STD/BCV pair.</td>
</tr>
<tr>
<td>22</td>
<td>Not enough resources (write-pending slots) to execute the SPLIT process. Try again later.</td>
</tr>
<tr>
<td>23</td>
<td>The BCV mirror is not fully synchronized with the STD mirror(s).</td>
</tr>
<tr>
<td>24</td>
<td>Rejected because STD would become invalid.</td>
</tr>
<tr>
<td>25</td>
<td>The BCV has four mirrors.</td>
</tr>
<tr>
<td>26</td>
<td>SDDF is not enabled, therefore a differential SPLIT operation is not possible.</td>
</tr>
<tr>
<td>27</td>
<td>The STDs and BCVs have a different number of meta members.</td>
</tr>
<tr>
<td>28</td>
<td>One device in the pair is a meta device and the other is not.</td>
</tr>
<tr>
<td>2A</td>
<td>The BCV is in transient state and the requested operation is rejected. For a Concurrent BCV, the first ESTABLISH may not have completed.</td>
</tr>
<tr>
<td>2B</td>
<td>The STD is in transient state and the requested operation is rejected.</td>
</tr>
<tr>
<td>2C</td>
<td>Attempting a regular SPLIT on a meta volume where not all the members are established.</td>
</tr>
<tr>
<td>2D</td>
<td>The pair has different RAD flag in flags 2.</td>
</tr>
<tr>
<td>30</td>
<td>Illegal TF/Mirror command.</td>
</tr>
<tr>
<td>31</td>
<td>Configuration upgrade in progress.</td>
</tr>
</tbody>
</table>
## Table 34 TF/Mirror reason codes (page 3 of 5)

<table>
<thead>
<tr>
<th>Error code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>Cannot use internal BCV (DRVs) devices.</td>
</tr>
<tr>
<td>33</td>
<td>Cannot use STD for internal BCV.</td>
</tr>
<tr>
<td>34</td>
<td>Cannot create GST record for DAS.</td>
</tr>
<tr>
<td>35</td>
<td>BCV assigned to SNAP session or is held.</td>
</tr>
<tr>
<td>36</td>
<td>Moving the BCV mirror away would leave invalid tracks.</td>
</tr>
<tr>
<td>37</td>
<td>A Reverse SPLIT was requested on a device with only one mirror.</td>
</tr>
<tr>
<td>38</td>
<td>BCV has write pendings (not synchronized with its mirror).</td>
</tr>
<tr>
<td>39</td>
<td>The pair has different physical block sizes. Low level format difference.</td>
</tr>
<tr>
<td>3D</td>
<td>Destination device not ready.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> This code comes from TF/Clone Mainframe Snap Facility. TF/Mirror invokes TF/Clone Mainframe Snap Facility syscalls through the TF/Mirror CONFIG command.</td>
</tr>
<tr>
<td>40</td>
<td>BCV is not on this host adapter.</td>
</tr>
<tr>
<td>41</td>
<td>A background SPLIT is in progress, BCV cannot be used. SR device is not a BCV.</td>
</tr>
<tr>
<td>42</td>
<td>A background SPLIT is in progress, STD device cannot be used. Device already set or released.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> This code comes from TF/Clone Mainframe Snap Facility. TF/Mirror invokes TF/Clone Mainframe Snap Facility syscalls through the TF/Mirror CONFIG command.</td>
</tr>
<tr>
<td>43</td>
<td>Not enough concurrent copy slots exist to keep track of yet another BCV. Device is an established BCV.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> This code comes from TF/Clone Mainframe Snap Facility. TF/Mirror invokes TF/Clone Mainframe Snap Facility syscalls through the TF/Mirror CONFIG command.</td>
</tr>
<tr>
<td>44</td>
<td>Illegal to force an Instant SPLIT.</td>
</tr>
<tr>
<td></td>
<td>Device is an active EMCSNAP device.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> This code comes from TF/Clone Mainframe Snap Facility. TF/Mirror invokes TF/Clone Mainframe Snap Facility syscalls through the TF/Mirror CONFIG command.</td>
</tr>
<tr>
<td>45</td>
<td>SNAP session (File SMMF) exists for this BCV. Syscall 812C - illegal modifier.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> This code comes from TF/Clone Mainframe Snap Facility. TF/Mirror invokes TF/Clone Mainframe Snap Facility syscalls through the TF/Mirror CONFIG command.</td>
</tr>
</tbody>
</table>
Table 34 TF/Mirror reason codes (page 4 of 5)

<table>
<thead>
<tr>
<th>Error code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>46</td>
<td>The pair specified are meta devices with differing stripe sizes. The device has CCOPY (Protection Session) sessions established.</td>
</tr>
<tr>
<td>47</td>
<td>The STD has an active file level SNAP session.</td>
</tr>
<tr>
<td>48</td>
<td>Timeout.</td>
</tr>
<tr>
<td>50</td>
<td>In SymmVault mode, no SMMF operations are allowed.</td>
</tr>
<tr>
<td>51</td>
<td>SymmVault BCV cannot receive any data through an ESTABLISH after a track was protected.</td>
</tr>
<tr>
<td>52</td>
<td>SymmVault STD cannot receive any data through a RESTORE after a track was protected.</td>
</tr>
<tr>
<td>53</td>
<td>STD cannot be a virtual device or a log device.</td>
</tr>
<tr>
<td>55</td>
<td>If STD is striped, SMMF operations are on the HEAD of the device.</td>
</tr>
<tr>
<td>56</td>
<td>If BCV is striped, SMMF operations are on the HEAD of the device.</td>
</tr>
<tr>
<td>60</td>
<td>Moving mirror is not ready.</td>
</tr>
<tr>
<td>61</td>
<td>Moving mirror is not write-enabled.</td>
</tr>
<tr>
<td>62</td>
<td>Moving mirror is not valid.</td>
</tr>
<tr>
<td>63</td>
<td>Syccall 813D: Invalid number of entries.</td>
</tr>
<tr>
<td>65</td>
<td>Syccall 813D: Instant SPLIT flag not set.</td>
</tr>
<tr>
<td>66</td>
<td>The STD is assigned to a TF/Clone Mainframe Snap Facility session or is held.</td>
</tr>
<tr>
<td>67</td>
<td>Syccall 812D: Some of the data mirrors of the RAID group have invalid tracks.</td>
</tr>
<tr>
<td>68</td>
<td>Syccall 8125: No SPLIT request exists for the other BCV concurrently established.</td>
</tr>
<tr>
<td>69</td>
<td>Syccall 8125: RESTORE not allowed against a PPRC R2 STD.</td>
</tr>
<tr>
<td>70</td>
<td>Syccall 812D: Some of the data mirrors of a RAID group have invalid tracks.</td>
</tr>
<tr>
<td>71</td>
<td>Syccall 8125: ESTABLISH or RESTORE not allowed for a PPRC BCV.</td>
</tr>
<tr>
<td>72</td>
<td>Syccall 8125: ESTABLISH or RESTORE not allowed if SNOW is active and R1 is ready.</td>
</tr>
<tr>
<td>74</td>
<td>Syccall 8125: RESTORE is not allowed if R2 is STD and write-enabled and if the R1 is ready.</td>
</tr>
<tr>
<td>76</td>
<td>A virtual RESTORE has been aborted, leaving the BCV inconsistent and with aborted tracks.</td>
</tr>
<tr>
<td>77</td>
<td>BCV is a RAID-protected device.</td>
</tr>
<tr>
<td>78</td>
<td>Cannot split while STD is being attached.</td>
</tr>
<tr>
<td>79</td>
<td>Cannot perform Protected RESTORE while BCV has IVTOC tracks.</td>
</tr>
<tr>
<td>F0</td>
<td>Storage system completed the request although the BCV shows not in use.</td>
</tr>
<tr>
<td>F1</td>
<td>Retry count was exceeded for code 22.</td>
</tr>
</tbody>
</table>
Table 34  TF/Mirror reason codes (page 5 of 5)

<table>
<thead>
<tr>
<th>Error code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>F2</td>
<td>I/O error on SPLIT.</td>
</tr>
<tr>
<td>F3</td>
<td>Retry count was exceeded for code 11.</td>
</tr>
<tr>
<td>F8</td>
<td>Clone session detected in error.</td>
</tr>
</tbody>
</table>
EXTENTS reason codes

Table 35 describes reason codes for the EXTENTS program.

**Table 35** EXTENTS reason codes

<table>
<thead>
<tr>
<th>R15</th>
<th>Reason</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td></td>
<td>Successful</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>Illegal mask specified</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>No matching datasets found</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>Unknown function code</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>Overflow, unable to return all matching datasets</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>UCB not found</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>Dataset not found on volume</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>DSCB not type 1 or type 4</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>Tracks allocated = 0</td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>Extents not available due to HSM migrate</td>
</tr>
<tr>
<td>A</td>
<td></td>
<td>Caller is not APF-authorized</td>
</tr>
<tr>
<td>B</td>
<td></td>
<td>Extents program logic error</td>
</tr>
<tr>
<td>C</td>
<td></td>
<td>Entry type = GDG base</td>
</tr>
<tr>
<td>8</td>
<td></td>
<td>SVC26 - catalog management return code</td>
</tr>
<tr>
<td>12</td>
<td>8</td>
<td>Dataset is not cataloged</td>
</tr>
<tr>
<td>?</td>
<td></td>
<td>Locate failed - the reason code contains the locate return code</td>
</tr>
<tr>
<td>16</td>
<td></td>
<td>EXTWA failed validation</td>
</tr>
</tbody>
</table>