
Using SYMCLI to Set Up TimeFinder/Mirror Multiple BCVs and Concurrent BCVs

Abstract

This paper provides an introduction to setting up multiple and concurrent BCVs in the TimeFinder/Mirror environment. The paper focuses on pairing one standard device sequentially with multiple BCVs and pairing one standard device simultaneously with two BCVs.

Copyright © 2005 EMC Corporation. All rights reserved.

EMC believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

THE INFORMATION IN THIS PUBLICATION IS PROVIDED “AS IS”. EMC CORPORATION MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WITH RESPECT TO THE INFORMATION IN THIS PUBLICATION, AND SPECIFICALLY DISCLAIMS IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Use, copying, and distribution of any EMC software described in this publication requires an applicable software license.

EMC², EMC, Symmetrix, Celerra, CLARiiON, CLARAlert, Documentum, HighRoad, Legato, Navisphere, PowerPath, ResourcePak, SnapView/IP, SRDF, TimeFinder, VisualSAN, and where information lives are registered trademarks and EMC Automated Networked Storage, EMC ControlCenter, EMC Developers Program, EMC OnCourse, EMC Proven, EMC Snap, Access Logix, AutoAdvice, Automated Resource Manager, AutoSwap, AVALONidm, C-Clip, Celerra Replicator, Centera, CentraStar, CLARevent, Connectrix, CopyCross, CopyPoint, DatabaseXtender, Direct Matrix, Direct Matrix Architecture, EDM, E-Lab, Enginuity, FarPoint, FLARE, GeoSpan, InfoMover, MirrorView, NetWin, OnAlert, OpenScale, Powerlink, PowerVolume, RepliCare, SafeLine, SAN Architect, SAN Copy, SAN Manager, SDMS, SnapSure, SnapView, StorageScope, SupportMate, SymmAPI, SymmEnabler, Symmetrix DMX, Universal Data Tone, and VisualSRM are trademarks of EMC Corporation. All other trademarks used herein are the property of their respective owners.

Part Number 300-000-073 REV G

Table of Contents

Introduction	4
Purpose and Scope	4
Related Documentation	4
 Practical Uses	 4
 Using Multiple BCVs	 5
 How to Set Up Multiple BCVs.....	 6
 Using Concurrent BCVs	 7
 Example 1: Pairing One Standard Sequentially with Multiple BCVs	 8
 Example 2: Concurrent BCVs	 22

Introduction

TimeFinder®/Mirror software (referred to as “TimeFinder” in this document) works by creating multiple, independently addressable business continuance volumes (BCVs) for independent storage. The BCV is a Symmetrix® device with special attributes created when the Symmetrix is configured. It can function either as an additional mirror to a Symmetrix logical volume or as an independent, host-addressable volume.

Establishing BCV devices as mirror images of active production volumes allows you to run multiple simultaneous business continuance tasks in parallel. The principal device, known as the *standard device*, remains online for regular Symmetrix operation from the original production server. Each BCV contains a unique host address, making it accessible to a separate backup/recovery server. When you establish a BCV as a mirror of a standard device, that relationship is known as a *BCV pair*.

When you establish two BCVs as concurrent mirrors of a single standard, that relationship is known as a *concurrent BCV pair*.

Purpose and Scope

This document describes TimeFinder functionality included in versions of EMC® Solutions Enabler up to version 6.0 running on Symmetrix arrays using Enginuity™ versions up to 5x71. The paper provides an introduction to:

- Pairing one standard device sequentially with multiple BCVs
- Pairing one standard device simultaneously with two BCVs

Related Documentation

The following manuals and white papers provide information related to the concepts in this paper:

- *EMC Solutions Enabler Symmetrix TimeFinder Family CLI Product Guide*
- *Using SYMCLI to Set Up TimeFinder/Mirror BCV Pairs* (P/N 300-000-072)
- *Using SYMCLI to Attach, Query, and Verify with TimeFinder Family Products* (P/N 300-000-075)
- *Using SYMCLI to Perform TimeFinder/Mirror Control Operations* (P/N 300-000-074)
- *Using SYMCLI to Perform SRDF/AR* (P/N 300-000-078)

Practical Uses

Multiple BCVs can satisfy an enterprise’s need for multiple uses of copies of the production data. You can designate one copy for testing, one copy for data warehousing, one copy for backups, one copy for report generation, and so forth. Some enterprises use the multi-BCV feature for improving data access speeds by assigning users to BCV copies of the data instead of the one-and-only production copy, thereby reducing disk contention.

The multi-BCV feature makes it possible to maintain multiple point-in-time copies of a database on BCVs, copies that are ready for use in the event of database corruption. Because copies to and from the BCVs are incremental operations (that is, copying only the changed data), the synchronization process is speedy and efficient. These point-in-time copies are instantly available for recovery operations.

Concurrent BCVs make two identical copies available at any point in time. This is practical when you need to have the same data available for more than one task.

Using Multiple BCVs

A standard device can be paired with multiple BCVs sequentially, provided that those BCVs have not been paired with another standard device. By default, you can establish up to eight BCVs with a standard device and have the standard retain its incremental pairing relationship with each of the eight BCVs.

After you establish a multi-BCV environment for a specific standard device, the Symmetrix system retains changed track information about all BCVs. For example, by performing incremental establish and split operations on the BCVs over a period of time, you can capture progressive historical snapshots of the specified standard device. Later, if necessary, you can have any of these BCVs restore information to the standard device or be incrementally re-established with the standard.

Canceling a BCV ends the existing incremental pairing relationship between the BCV and the standard device. The BCV assumes the SplitNoInc state. You cannot incrementally establish or incrementally restore that BCV pair again until you have performed another full establish operation or a full restore operation. You can explicitly cancel a BCV from a multi-BCV setup by using the `symmir cancel` command.

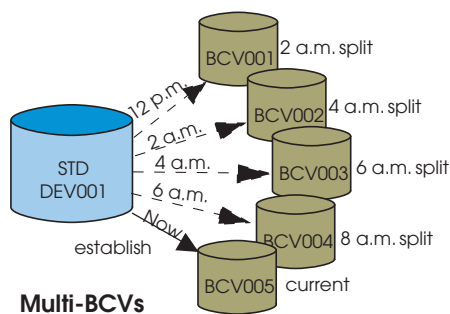


Figure 1. Establishing a Multi-BCV Environment

Figure 1 demonstrates a multi-BCV setup in which BCV001 was established at 12 P.M. and mirrored the activity of DEV001 until 2 A.M. At that time, BCV001 was split, and BCV002 was then established to mirror activity of the standard during the next two hours. Other BCVs mirrored activity at other intervals.

Two environment variables, `SYMCLI_BCV_PAIR_POLICY` and `SYMCLI_MAX_BCV_PAIRS`, affect the SYMCLI behavior in controlling multiple BCVs. `SYMCLI_BCV_PAIR_POLICY` governs the behavior of the $n+1$ BCV, where n represents the maximum number of BCVs that are permitted to have an incremental relationship with the standard device. The default value of n is 8. This value can be set to a number in the range of 1 to 16, depending on the value assigned to the environment variable `SYMCLI_MAX_BCV_PAIRS`. After the maximum allowable number of BCVs has been established with the standard, an attempt to establish the next BCV causes one of the following to occur:

- The oldest BCV is canceled, if `SYMCLI_BCV_PAIR_POLICY` is set to `CANCEL_OLDEST`.
- The newest BCV is canceled, if `SYMCLI_BCV_PAIR_POLICY` is set to `CANCEL_NEWEST`.
- An error message is generated, if `SYMCLI_BCV_PAIR_POLICY` is set to `DONT_CANCEL`.

If you do not set this environment value, the default is `CANCEL_OLDEST`.

If your business needs require that multiple BCV devices interact with the same standard device, SYMCLI allows you to rotate a standard device among multiple BCVs in various ways:

- With one device group, all standards and BCVs are part of that group. By using explicit pairing syntax, such as:

```
symmir -g GroupName -full establish DEV001 BCV 1d BCV001 DEV002 BCV 1d BCV002
```

you can choose to establish one set of standards with a set of BCVs. When it is time to establish those *same* standard devices (DEV001 and DEV002) with a fresh set of BCVs, use a command such as:

```
symmir -g GroupName -full establish DEV001 BCV 1d BCV007 DEV002 BCV 1d BCV008
```

- With multiple device groups, you can rotate standard devices among the device groups using the `symld` command with the `moveall` action. Using `symld moveall` allows you to move all standard devices from a specified source device group to a specified destination device group. The source and destination groups must have the same group type: Regular, RDF1, or RDF2. (Same group type applies only to standard devices; when moving BCVs among groups, the device group types can be different.) The following command moves all the standard devices from group1 to group2:

```
symld -g group1 moveall group2
```

- Device files are an alternative to device groups. You can store paired lists of standard and BCV devices in a text file and use this file wherever it is convenient. However, EMC recommends using device groups rather than device files whenever possible.

If you want to use one device pairing in the group whenever you need to perform an establish or a restore operation, you can specify that relationship using the `symmir attach` command. Specifying an attach assignment sets a BCV preference that TimeFinder recognizes whenever you issue `symmir establish` or `restore` commands without an explicit pair specification or an overriding pairing option.

How to Set Up Multiple BCVs

The usual sequence for setting up a relationship between one standard device and multiple BCVs is to first create a device group, add the standard and BCV devices to the group, and then establish an initial BCV pair. The following sequence outlines a basic setup:

1. Create a device group.
2. Add a standard device to the device group.
3. Add multiple BCVs to the device group.
4. Establish the initial BCV pair explicitly.
5. Split the initial BCV pair and establish a second BCV pair. Split the current pair when you wish to establish another BCV from the device group.

Using Concurrent BCVs

You can have two identical BCV copies available at any point in time by establishing two BCVs as concurrent mirrors of one standard device. Using a concurrent BCV pair like this allows you to easily generate two copies of the same data. When the two BCVs are split from the standard, the BCV's hosts can access the data on either BCV.

You can establish a concurrent BCV pair simultaneously with one command as follows:

```
symmir -g conbcv -full establish DEV001 bcv ld BCV001 DEV001 bcv ld BCV002
```

In the device group named conbcv, BCV001 is paired with the standard DEV001, and BCV002 is also paired with DEV001. While this example pairs both BCVs at the same time, you can also establish one BCV with the standard now and one BCV later. In either case, the standard and its concurrent BCVs become synchronized and stay that way until you split the concurrent BCV pair.

Beginning with Solutions Enabler version 6.0, you can establish concurrent BCVs without specifying the BCVs explicitly. For example, if your device group contains three synchronized BCV pairs and three or more additional unpaired BCVs, you can issue one command with the `-concurrent` option to establish three concurrent BCV pairs. The first standard device in the group is concurrently paired with one of the remaining unpaired BCVs using the “first available” match (that is, the first available BCV that meets pairing criteria such as size and emulation).

```
symmir -g conbcv -full establish -concurrent
```

If a BCV synchronized pair already exists in the device group, then this command adds a second BCV to form the concurrent BCV pair and performs a full copy only on the new BCV being synchronized. If a standard device is not synchronized with a BCV at the time of the command, the standard device becomes fully established with one BCV (ignoring the `-concurrent` option). If prior pairing relationships exist within the device group (for example, the concurrent BCV pair has been split), then re-establishing the pair and omitting the `-full` option results in the two concurrent BCVs being incrementally established.

In *clone emulation mode*, a standard device can have up to eight concurrent BCVs instead of two. Therefore, you can issue the `symmir establish -concurrent` command for the standard device up to eight times, adding one additional BCV each time. For information on clone emulation mode, refer to the white paper *Using SYMCLI to Perform TimeFinder/Mirror Control Operations* (P/N 300-000-074).

You can split both BCVs of a concurrent BCV pair either simultaneously or one at a time. When splitting simultaneously, TimeFinder regulates which BCV it splits first and which it splits second according to varying conditions. The following command performs a simultaneous instant split on the concurrent BCVs paired with the DEV001 standard:

```
symmir -g conbcv split DEV001
```

However, if you want to split the concurrent BCVs independently instead of simultaneously, use the `symmir split` command for each BCV:

```
symmir -g conbcv split DEV001 bcv ld BCV001
symmir -g conbcv split DEV001 bcv ld BCV002
```

The following command displays the status of the background split for both BCVs:

```
symmir -g conbcv query -multi -bg
```

Restoring data from the concurrent BCVs to the standard device is possible only after splitting both BCVs. However, after the split, the BCVs are no longer concurrent.

Example 1: Pairing One Standard Sequentially with Multiple BCVs

The examples show BCVs with physical device names, signifying that the BCVs are addressable by the host on which the standards are visible. While using a single host is convenient for demonstration purposes, usually a second host is used for accessing BCVs for business continuance tasks.

Beginning with Solutions Enabler version 5.0 running on Symmetrix arrays using Enginuity version 5568, all splits are instant splits. Including the `-instant` option on the `symmir split` command line causes SYMCLI to return immediately after the foreground split. Omitting the `-instant` option causes SYMCLI to wait until the background split completes before returning.¹

Creating a device group and adding devices to it are prerequisites for establishing BCV pairs. The following commands create the device group `ProdBgrp` and set environment variables to eliminate the need for typing repetitive options.

```
# symdg create ProdBgrp
```

Setting `DG` prevents you from having to type `-g ProdBgrp` in `symld` and `symmir` commands. Setting `SID` identifies the Symmetrix device number for commands such as `symld` and `symbcv` that may require it. Setting `NOPROMPT` disables prompting for control commands. Setting `LDEV_NAMING` assigns physical device numbers as logical names as opposed to `DEV` or `BCV` type names.

```
# setenv SYMCLI_DG ProdBgrp
# setenv SYMCLI_SID 3264
# setenv SYMCLI_NOPROMPT 1
# setenv SYMCLI_LDEV_NAMING PDEV
```

The following `symcli` command displays currently defined environment variables.

```
# symcli -def
```

Symmetrix Command Line Interface (SYMCLI) Version V5.4-154 (Rev 5.0) built with SYMAPI Version V5.4-154 (Rev 5.0)

Current settings of the SYMCLI environmental variables:

```
SYMCLI_SID          : 3264
SYMCLI_DG           : ProdBgrp
SYMCLI_NOPROMPT     : 1
SYMCLI_LDEV_NAMING  : PDEV
```

The following `symld` command adds standard device 84 to the device group. With the current environment variable settings, there is no need to include `-g ProdBgrp -sid 3264` in the command.

```
# symld add dev 84
```

¹ Another way to make SYMCLI wait for the background split to complete is to set the `SYMAPI_WAIT_FOR_BG_SPLIT` variable in the options file located in the SYMAPI configuration directory. Setting this variable to `TRUE` causes the wait regardless of whether the `symmir split` command includes the `-instant` option.

The following command adds all ungrouped BCVs in the range 88 to 98 to the device group. When using the `symbcv` command as shown here to add multiple BCVs to a device group, you can substitute the syntax `addall` for `associateall`. When adding a single BCV, you can substitute the syntax `add` for `associate`.

```
# symbcv -RANGE 88:98 associateall dev
```

The `symdg show` command displays devices in the group. Note in this display that the logical device names match the physical device names. If the `LDEV_NAMING` environment variable is not set, or if the devices are invisible to the host, the names default to DEV001 for the standard device and BCV001, BCV002, and so forth for the BCVs.

```
# symdg show ProdBgrp
```

Group Name: ProdBgrp

```
Group Type                : REGULAR
Device Group in GNS       : Yes
Valid                     : Yes
Symmetrix ID              : 000000003264
Group Creation Time       : Tue Jan  6 12:08:17 2004
Vendor ID                 : EMC Corp
Application ID            : SYMCLI
```

```
Number of STD Devices in Group : 1
Number of Associated GK's      : 0
Number of Locally-associated VDEV's : 0
Number of Locally-associated BCV's : 17
Number of Remotely-associated BCV's (STD RDF) : 0
Number of Remotely-associated BCV's (BCV RDF) : 0
Number of Remotely-assoc'd RBCV's (RBCV RDF) : 0
```

Standard (STD) Devices (1):

```
{
-----
LdevName          PdevName          Sym   Att.  Sts   Cap
Dev              Dev              Dev
-----
c4t0d0s2          /dev/rdisk/c4t0d0s2    0084      RW    516
}
```

BCV Devices Locally-associated (17):

```
{
-----
LdevName          PdevName          Sym   Att.  Sts   Cap
Dev              Dev              Dev
-----
c4t1d0s2          /dev/rdisk/c4t1d0s2    0088      RW    516
c4t1d1s2          /dev/rdisk/c4t1d1s2    0089      RW    516
c4t1d2s2          /dev/rdisk/c4t1d2s2    008A      RW    516
c4t1d3s2          /dev/rdisk/c4t1d3s2    008B      RW    516
c4t1d4s2          /dev/rdisk/c4t1d4s2    008C      RW    516
c4t1d5s2          /dev/rdisk/c4t1d5s2    008D      RW    516
c4t1d6s2          /dev/rdisk/c4t1d6s2    008E      RW    516
c4t1d7s2          /dev/rdisk/c4t1d7s2    008F      RW    516
c4t2d0s2          /dev/rdisk/c4t2d0s2    0090      RW    516
}
```

```

c4t2d1s2          /dev/rdisk/c4t2d1s2      0091      RW      516
c4t2d2s2          /dev/rdisk/c4t2d2s2      0092      RW      516
c4t2d3s2          /dev/rdisk/c4t2d3s2      0093      RW      516
c4t2d4s2          /dev/rdisk/c4t2d4s2      0094      RW      516
c4t2d5s2          /dev/rdisk/c4t2d5s2      0095      RW      516
c4t2d6s2          /dev/rdisk/c4t2d6s2      0096      RW      516
c4t2d7s2          /dev/rdisk/c4t2d7s2      0097      RW      516
c4t3d0s2          /dev/rdisk/c4t3d0s2      0098      RW      516
}

```

The following command establishes the standard device with BCV 88 as the first pairing in a multi-BCV setup.

```
# symmir -full establish c4t0d0s2 BCV dev 88
```

'Full Establish' operation execution is in progress for device 'c4t0d0s2' paired with BCV device '0088' in device group 'ProdBgrp'. Please wait...

'Full Establish' operation successfully initiated for device 'c4t0d0s2' in group 'ProdBgrp' paired with BCV device '0088'.

The `symmir query` commands display the progress of the synchronization. Another way to monitor progress is to use the `verify` action, shown later in this paper. The ellipsis (.....) indicates where output was omitted.

```
# symmir query
```

```
Device Group (DG) Name: ProdBgrp
DG's Type           : REGULAR
DG's Symmetrix ID   : 000000003264
```

Standard Device			BCV Device			State
Logical	Inv. Sym Tracks	Logical	Sym	Inv. Tracks	STD <=> BCV	
c4t0d0s2	0084 0	c4t1d0s2	0088 *	5818	SyncInProgress	
Total	-----			-----		
MB(s)	0.0			181.8		

.....

```
# symmir query
```

```
Device Group (DG) Name: ProdBgrp
DG's Type           : REGULAR
DG's Symmetrix ID   : 000000003264
```

Standard Device			BCV Device			State
Logical	Inv. Sym Tracks	Logical	Sym	Inv. Tracks	STD <=> BCV	
c4t0d0s2	0084 0	c4t1d0s2	0088 *	5818	Synchronized	
Total	-----			-----		
MB(s)	0.0			0.0		

.....

The `symmir split` command performs an instant split on the BCV pair but waits for the background split to complete before returning to the caller. The `symmir establish` commands that follow will pair the standard device with seven more BCVs in succession. Wait for an establish operation to complete before attempting a split operation.

```
# symmir split
```

```
'Split' operation execution is in progress for device group 'ProdBgrp'.
Please wait...
```

```
'Split' operation successfully executed for device group 'ProdBgrp'.
```

```
# symmir -full establish c4t0d0s2 BCV dev 89
```

```
'Full Establish' operation execution is in progress for device 'c4t0d0s2'
paired with BCV device '0089' in device group 'ProdBgrp'. Please wait...
```

```
'Full Establish' operation successfully initiated for device 'c4t0d0s2'
in group 'ProdBgrp' paired with BCV device '0089'.
```

```
# symmir split
```

```
'Split' operation execution is in progress for device group 'ProdBgrp'.
Please wait...
```

```
'Split' operation successfully executed for device group 'ProdBgrp'.
```

The following interactive UNIX c-shell script establishes the standard device with each of six BCVs in succession, starting with BCV 8A. The `symmir verify` command runs in a loop, checking the status of the BCV pair at 90-second intervals. Upon completion of synchronization, the pair is split and the next BCV is established with the standard.

```
# foreach i (8A 8B 8C 8D 8E 8F)
? symmir -full establish c4t0d0s2 BCV dev $i
? symmir verify -i 90
? symmir split
? end
```

A `symmir query -multi` command displays a list of BCVs with the most recent BCV occupying the top position.

```
# symmir query -multi
```

```
Device Group (DG) Name: ProdBgrp
DG's Type           : REGULAR
DG's Symmetrix ID   : 000000003264
```

Standard Device			BCV Device		State
Logical	Inv. Sym Tracks	Logical	Inv. Sym Tracks	STD <=> BCV	
c4t0d0s2	0084	0 c4t1d7s2	008F *	0	Split
		0 c4t1d6s2	008E *	0	Split
		0 c4t1d5s2	008D *	0	Split

0	c4t1d4s2	008C *	0	Split
0	c4t1d3s2	008B *	0	Split
0	c4t1d2s2	008A *	0	Split
0	c4t1d1s2	0089 *	0	Split
0	c4t1d0s2	0088 *	0	Split
Total (Primary) -----				
Track(s)	0		0	
MB(s)	0.0		0.0	

Legend:

(*): The paired BCV device is associated with this group.

symmir -full establish c4t0d0s2 BCV dev 90

'Full Establish' operation execution is in progress for device 'c4t0d0s2' paired with BCV device '0090' in device group 'ProdBgrp'. Please wait...

'Full Establish' operation successfully initiated for device 'c4t0d0s2' in group 'ProdBgrp' paired with BCV device '0090'.

Because the default setting for MAX_BCV_PAIRS is 8 and BCV_PAIR_POLICY defaults to CANCEL_OLDEST, establishing a ninth BCV will cause the oldest BCV (88) to lose its incremental relationship with standard device 84 and drop off the bottom of the list. The symmir query command shows that BCV 90 was added to the head of the list.

symmir query -multi

Device Group (DG) Name: ProdBgrp
 DG's Type : REGULAR
 DG's Symmetrix ID : 000000003264

Standard Device			BCV Device			State
Logical	Sym	Inv. Tracks	Logical	Sym	Inv. Tracks	STD <=> BCV
c4t0d0s2	0084	0	c4t2d0s2	0090 *	13488	SyncInProg
			c4t1d7s2	008F *	0	Split
			c4t1d6s2	008E *	0	Split
			c4t1d5s2	008D *	0	Split
			c4t1d4s2	008C *	0	Split
			c4t1d3s2	008B *	0	Split
			c4t1d2s2	008A *	0	Split
			c4t1d1s2	0089 *	0	Split
Total (Primary) -----						
Track(s)		0			13488	
MB(s)		0.0			421.5	

Legend:

(*): The paired BCV device is associated with this group.

BCV 88 is now in the SplitNoInc state, as the `syml d list` command below shows.

syml d list

Device Group (DG) Name: ProdBgrp
 DG's Type : REGULAR
 DG's Symmetrix ID : 000000003264

Standard Device Name			Directors	Device		
Logical	Physical	Sym	SA :P DA :IT	Config	Att Sts	Cap (MB)
DEV001	/dev/rdisk/c4t0d0s2	0084 +	16B:0 01A:C0	2-Way Mir	RW	516

Legend for STD devices:

(+): Paired with a BCV device that is associated with this dg.
 (-): Paired with a BCV device that is non-associated with this dg.

BCV Devices associated with this dg:

BCV Device				Standard Device			Status
Logical	Sym	RDF Att.	Inv. Tracks	Logical	Sym	Inv. Tracks	BCV <=> STD
c4t1d0s2	0088	+	0	c4t0d0s2	0084	0	SplitNoInc
c4t1d1s2	0089	+	0	c4t0d0s2	0084	0	Split
c4t1d2s2	008A	+	0	c4t0d0s2	0084	0	Split
c4t1d3s2	008B	+	0	c4t0d0s2	0084	0	Split
c4t1d4s2	008C	+	0	c4t0d0s2	0084	0	Split
c4t1d5s2	008D	+	0	c4t0d0s2	0084	0	Split
c4t1d6s2	008E	+	0	c4t0d0s2	0084	0	Split
c4t1d7s2	008F	+	0	c4t0d0s2	0084	0	Split
c4t2d0s2	0090	+	0	c4t0d0s2	0084	0	Synchronized
Total			-----			-----	
Track(s)			0			0	
MB(s)			0.0			0.0	

The `symmir split` command performs an instant split on the synchronized BCV pair but waits for the background split to complete before returning to the caller.

symmir split

'Split' operation execution is in progress for device group 'ProdBgrp'.
 Please wait...

'Split' operation successfully executed for device group 'ProdBgrp'.

```
# setenv SYMCLI_BCV_PAIR_POLICY CANCEL_NEWEST
# symcli -def
```

Symmetrix Command Line Interface (SYMCLI) Version V5.4-154 (Rev 5.0) built with SYMAPI Version V5.4-154 (Rev 5.0)

Current settings of the SYMCLI environmental variables:

```
SYMCLI_SID          : 3264
SYMCLI_DG           : ProdBgrp
SYMCLI_NOPROMPT     : 1
SYMCLI_LDEV_NAMING  : PDEV
SYMCLI_BCV_PAIR_POLICY: CANCEL_NEWEST
```

When BCV 88 was previously canceled, all record of track changes for this BCV was destroyed. To establish this BCV again requires a full establish operation. An incremental establish operation here would fail.

```
# symmir -full establish c4t0d0s2 BCV dev 88
```

'Full Establish' operation execution is in progress for device 'c4t0d0s2' paired with BCV device '0088' in device group 'ProdBgrp'. Please wait...

'Full Establish' operation successfully initiated for device 'c4t0d0s2' in group 'ProdBgrp' paired with BCV device '0088'.

Because the BCV_PAIR_POLICY is now set to CANCEL_NEWEST, the symmir query command indicates that the most recently established device (BCV 90) dropped off the list and assumed the SplitNoInc state.

```
# symmir query -multi
```

```
Device Group (DG) Name: ProdBgrp
DG's Type           : REGULAR
DG's Symmetrix ID   : 000000003264
```

Standard Device			BCV Device		State
Logical	Sym	Inv. Tracks	Logical	Sym	Inv. Tracks
c4t0d0s2	0084	0	c4t1d0s2	0088 *	0 Synchronized
			c4t1d7s2	008F *	0 Split
			c4t1d6s2	008E *	0 Split
			c4t1d5s2	008D *	0 Split
			c4t1d4s2	008C *	0 Split
			c4t1d3s2	008B *	0 Split
			c4t1d2s2	008A *	0 Split
			c4t1d1s2	0089 *	0 Split
Total (Primary)	-----				-----
MB (s)		0.0			0.0

Legend:

(*): The paired BCV device is associated with this group.

The `symmir split` command performs an instant split on the synchronized BCV pair but waits for the background split to complete before returning to the caller.

```
# symmir split
```

'Split' operation execution is in progress for device group 'ProdBgrp'. Please wait...

'Split' operation successfully executed for device group 'ProdBgrp'.

```
# setenv SYMCLI_BCV_PAIR_POLICY DONT_CANCEL
```

```
# symcli -def
```

Symmetrix Command Line Interface (SYMCLI) Version V5.4-154 (Rev 5.0) built with SYMAPI Version V5.4-154 (Rev 5.0)

Current settings of the SYMCLI environmental variables:

```
SYMCLI_SID           : 3264
SYMCLI_DG            : ProdBgrp
SYMCLI_NOPROMPT      : 1
SYMCLI_LDEV_NAMING   : PDEV
SYMCLI_BCV_PAIR_POLICY: DONT_CANCEL
```

```
# symmir -full establish c4t0d0s2 BCV dev 90
```

'Full Establish' operation execution is in progress for device 'c4t0d0s2' paired with BCV device '0090' in device group 'ProdBgrp'. Please wait...

The maximum number of BCV devices that can be incrementally paired with a standard device has been exceeded

This time the establish operation fails because the policy is `DONT_CANCEL`. BCV 90 remains in the `SplitNoInc` state. To explicitly cancel a BCV pair relationship, use the `symmir cancel` command. The BCV pair must be in the `Split` state for the cancel operation to work. BCV 89 is in the `Split` state.

```
# symmir cancel c4t0d0s2 BCV dev 89
```

'Cancel' operation execution is in progress for device 'c4t0d0s2' paired with BCV device '0089' in device group 'ProdBgrp'. Please wait...

'Cancel' operation successfully executed for device 'c4t0d0s2' in group 'ProdBgrp' paired with BCV device '0089'.

BCV 089 is now in the SplitNoInc state, as the `symld list` command displays.

`symld list`

Device Group (DG) Name: ProdBgrp
 DG's Type : REGULAR
 DG's Symmetrix ID : 000000003264

Standard Device Name			Directors	Device		
Logical	Physical	Sym	SA :P DA :IT	Config	Att Sts	Cap (MB)
DEV001	/dev/rdisk/c4t0d0s2	0084 +	16B:0 01A:C0	2-Way Mir	RW	516

Legend for STD devices:

(+): Paired with a BCV device that is associated with this dg.
 (-): Paired with a BCV device that is non-associated with this dg.

BCV Devices associated with this dg:

BCV Device				Standard Device			Status
Logical	Sym	RDF Att.	Inv. Tracks	Logical	Sym	Inv. Tracks	BCV <=> STD
c4t1d0s2	0088	+	0	c4t0d0s2	0084	0	Split
c4t1d1s2	0089	+	0	c4t0d0s2	0084	0	SplitNoInc
c4t1d2s2	008A	+	0	c4t0d0s2	0084	0	Split
c4t1d3s2	008B	+	0	c4t0d0s2	0084	0	Split
c4t1d4s2	008C	+	0	c4t0d0s2	0084	0	Split
c4t1d5s2	008D	+	0	c4t0d0s2	0084	0	Split
c4t1d6s2	008E	+	0	c4t0d0s2	0084	0	Split
c4t1d7s2	008F	+	0	c4t0d0s2	0084	0	Split
c4t2d0s2	0090	+	0	c4t0d0s2	0084	0	SplitNoInc
c4t2d1s2	0091		0	N/A	N/A	0	NeverEstab
c4t2d2s2	0092		0	N/A	N/A	0	NeverEstab
c4t2d3s2	0093		0	N/A	N/A	0	NeverEstab
c4t2d4s2	0094		0	N/A	N/A	0	NeverEstab
c4t2d5s2	0095		0	N/A	N/A	0	NeverEstab
c4t2d6s2	0096		0	N/A	N/A	0	NeverEstab
c4t2d7s2	0097		0	N/A	N/A	0	NeverEstab
c4t3d0s2	0098		0	N/A	N/A	0	NeverEstab
Total			-----			-----	
Track(s)			0			0	
MB(s)			0.0			0.0	

Legend for BCV devices:

(+): BCV is paired with a member STD device.
 (-): BCV is paired with a non-member STD device.

The following interactive UNIX c-shell script initiates incremental establish operations with the BCVs that currently have a pairing relationship with the standard device. If the incremental establish operation returns an error status because the BCV and the standard do not have a valid incremental pairing relationship, the script then initiates a full establish operation.

```
# setenv SYMCLI_MAX_BCV_PAIRS 16
# foreach i (88 89 8A 8B 8C 8D 8E 8F 90 91 92 93 94 95 96 97)
? symmir establish c4t0d0s2 BCV dev $i
? if ($status != 0) symmir establish -full c4t0d0s2 BCV dev $i
? symmir verify -i 30
? symmir split
? end
```

The following `symmir query` command displays sixteen BCVs that can be incrementally established.

```
# symmir query -multi
```

```
Device Group (DG) Name: ProdBgrp
DG's Type           : REGULAR
DG's Symmetrix ID   : 000000003264
```

Standard Device			BCV Device			State
Logical	Sym	Inv. Tracks	Logical	Sym	Inv. Tracks	STD <=> BCV
c4t0d0s2	0084	0	c4t2d7s2	0097 *	0	Split
		0	c4t2d6s2	0096 *	0	Split
		0	c4t2d5s2	0095 *	0	Split
		0	c4t2d4s2	0094 *	0	Split
		0	c4t2d3s2	0093 *	0	Split
		0	c4t2d2s2	0092 *	0	Split
		0	c4t2d1s2	0091 *	0	Split
		0	c4t2d0s2	0090 *	0	Split
		0	c4t1d7s2	008F *	0	Split
		0	c4t1d6s2	008E *	0	Split
		0	c4t1d5s2	008D *	0	Split
		0	c4t1d4s2	008C *	0	Split
		0	c4t1d3s2	008B *	0	Split
		0	c4t1d2s2	008A *	0	Split
		0	c4t1d1s2	0089 *	0	Split
		0	c4t1d0s2	0088 *	0	Split
Total (Primary)	-----				-----	
MB(s)		0.0				0.0

Legend:

(*): The paired BCV device is associated with this group.

The `symmir establish` command pairs the most recently established BCV (97) with the standard device if no prior pairing preferences were set up using the `attach` action. The `-v` option provides a more detailed output.

```
# symmir establish -v
```

```
'Incremental Establish' operation execution is in progress for device group  
'ProdBgrp'. Please wait...
```

```
SELECTING Standard devices in the group:
```

```
Device: 0084 [SELECTED]
```

```
SELECTING Target devices in the group:
```

```
Device: 0088 [SELECTED]  
Device: 0089 [SELECTED]  
Device: 008A [SELECTED]  
Device: 008B [SELECTED]  
Device: 008C [SELECTED]  
Device: 008D [SELECTED]  
Device: 008E [SELECTED]  
Device: 008F [SELECTED]  
Device: 0090 [SELECTED]  
Device: 0091 [SELECTED]  
Device: 0092 [SELECTED]  
Device: 0093 [SELECTED]  
Device: 0094 [SELECTED]  
Device: 0095 [SELECTED]  
Device: 0096 [SELECTED]  
Device: 0097 [SELECTED]
```

```
PAIRING of Standard and BCV devices:
```

```
Devices: 0084(S) - 0097(B) [PAIRED]
```

```
STARTING a BCV 'INCREMENTAL_ESTABLISH' operation.
```

```
The BCV 'INCREMENTAL_ESTABLISH' operation SUCCEEDED.
```

```
'Incremental Establish' operation successfully initiated for device group  
'ProdBgrp'.
```

The `symmir query` command confirms that the standard is now paired with BCV c4t2d7s2 (97), its most recent partner.

`symmir query`

Device Group (DG) Name: ProdBgrp
 DG's Type : REGULAR
 DG's Symmetrix ID : 000000003264

Standard Device			BCV Device			State
Logical	Sym	Inv. Tracks	Logical	Sym	Inv. Tracks	STD <=> BCV
c4t0d0s2	0084	0	c4t2d7s2	0097 *	0	Synchronized
Total		-----			-----	
MB(s)		0.0			0.0	

Legend:

(*): The paired BCV device is associated with this group.

`symmir split`

'Split' operation execution is in progress for device group 'ProdBgrp'.
 Please wait...

'Split' operation successfully executed for device group 'ProdBgrp'.

Using the `symmir attach` command sets up a preferred pairing relationship between the standard device and BCV 90. The subsequent `symmir query` command with the `-attach` option displays the attachment and also that standard device 84 has a current pairing relationship (Split state) with BCV 97.

`symmir attach c4t0d0s2 BCV dev 90` # `symmir query -attach`

Device Group (DG) Name: ProdBgrp
 DG's Type : REGULAR
 DG's Symmetrix ID : 000000003264

Standard Device		Paired BCV Device		Attached BCV Device		State
Logical	Sym	Logical	Sym	Logical	Sym	STD <=> BCV
c4t0d0s2	0084	c4t2d7s2	0097 *	c4t2d0s2	0090 *	Split

Legend:

(*): The BCV device is associated with this group.

Because of their preferred pairing relationship, the `symmir establish` command pairs the standard device with BCV 90.

```
# symmir establish -v
```

```
'Incremental Establish' operation execution is in progress for device group  
'ProdBgrp'. Please wait...
```

```
SELECTING Standard devices in the group:
```

```
Device: 0084 [SELECTED]
```

```
SELECTING Target devices in the group:
```

```
Device: 0088 [SELECTED]  
Device: 0089 [SELECTED]  
Device: 008A [SELECTED]  
Device: 008B [SELECTED]  
Device: 008C [SELECTED]  
Device: 008D [SELECTED]  
Device: 008E [SELECTED]  
Device: 008F [SELECTED]  
Device: 0090 [SELECTED]  
Device: 0091 [SELECTED]  
Device: 0092 [SELECTED]  
Device: 0093 [SELECTED]  
Device: 0094 [SELECTED]  
Device: 0095 [SELECTED]  
Device: 0096 [SELECTED]  
Device: 0097 [SELECTED]
```

```
PAIRING of Standard and BCV devices:
```

```
Devices: 0084(S) - 0090(B) [PAIRED]
```

```
STARTING a BCV 'INCREMENTAL_ESTABLISH' operation.
```

```
The BCV 'INCREMENTAL_ESTABLISH' operation SUCCEEDED.
```

```
'Incremental Establish' operation successfully initiated for device group  
'ProdBgrp'.
```

The `symmir query` command displays the results in the standard format and that the BCV pair is now in the Synchronized state.

`symmir query`

Device Group (DG) Name: ProdBgrp
 DG's Type : REGULAR
 DG's Symmetrix ID : 000000003264

Standard Device			BCV Device			State
Logical	Sym	Inv. Tracks	Logical	Sym	Inv. Tracks	STD <=> BCV
c4t0d0s2	0084	0	c4t2d0s2	0090 *	0	Synchronized
Total		-----			-----	
MB(s)		0.0			0.0	

Legend:

(*): The paired BCV device is associated with this group.

The `symmir query -attach` command displays a format showing that BCV 90 is the attached BCV and that BCV 90 is the device now having the current pairing relationship (Synchronized state) with standard device 84.

`symmir query -attach`

Device Group (DG) Name: ProdBgrp
 DG's Type : REGULAR
 DG's Symmetrix ID : 000000003264

Standard Device		Paired BCV Device		Attached BCV Device		State
Logical	Sym	Logical	Sym	Logical	Sym	STD <=> BCV
c4t0d0s2	0084	c4t2d0s2	0090 *	c4t2d0s2	0090 *	Synchronized

Legend:

(*): The BCV device is associated with this group.

Example 2: Concurrent BCVs

The following commands that illustrate concurrent BCVs were issued in a separate SYMCLI session on a different host and Symmetrix than the host/Symmetrix used in Example 1. Therefore, the environment variables defined in Example 1 are no longer valid for the new environment.

The following `symdg` command creates a new device group called `conbcv`. The `symldev` commands add two standard devices to the group. The `symlbcv` command associates four BCV devices with the group.

```
# symdg create conbcv
# symldev -g conbcv -sid 184502237 add dev 355
# symldev -g conbcv -sid 184502237 add dev 356
# symlbcv -g conbcv -sid 184502237 -RANGE 35F:362 associateall dev
```

The `symdg show` command displays the group's devices and their assigned logical names.

```
# symdg show conbcv
```

Group Name: conbcv

```
Group Type                : REGULAR
Device Group in GNS       : Yes
Valid                     : Yes
Symmetrix ID              : 000184502237
Group Creation Time       : Tue Jan 13 11:41:32 2004
Vendor ID                 : EMC Corp
Application ID            : SYMCLI
Number of STD Devices in Group : 2
Number of Associated GK's  : 0
Number of Locally-associated VDEV's : 0
Number of Locally-associated BCV's : 4
Number of Remotely-associated BCV's (STD RDF) : 0
Number of Remotely-associated BCV's (BCV RDF) : 0
Number of Remotely-assoc'd RBCV's (RBCV RDF) : 0
```

Standard (STD) Devices (2):

```
{
-----
LdevName          PdevName          Sym   Att. Sts   Cap
Dev              (MB)
-----
DEV001            /dev/rdisk/c4t0d0    0355   RW      4315
DEV002            /dev/rdisk/c4t0d1    0356   RW      4315
}
```

BCV Devices Locally-associated (4):

```
{
-----
LdevName          PdevName          Sym   Att. Sts   Cap
Dev              (MB)
-----
BCV001            N/A                035F   RW      4315
BCV002            N/A                0360   RW      4315
BCV003            N/A                0361   RW      4315
BCV004            N/A                0362   RW      4315
}
```

The `symmir attach` commands create preferred standard/BCV relationships between devices DEV001 and BCV001, and between DEV002 and BCV003.

```
# symmir -g conbcv attach DEV001 bcv ld BCV001 -noprompt
# symmir -g conbcv attach DEV002 bcv ld BCV003 -noprompt
```

The following `symmir establish` command creates concurrent BCV pairs. BCV001 and BCV002 are matched with standard DEV001, while BCV003 and BCV004 are matched with DEV002.

```
# symmir -g conbcv -full establish DEV001 bcv ld BCV001 DEV001 bcv ld BCV002 \
      DEV002 bcv ld BCV003 DEV002 bcv ld BCV004 -noprompt
```

'Full Establish' operation execution is in progress for the device list in device group 'conbcv'. Please wait...

'Full Establish' operation successfully initiated for the device list in device group 'conbcv'.

The `symmir query` with the `-multi` option displays the status of the concurrent BCVs (state is `SyncInProg`).

```
# symmir -g conbcv query -multi
```

```
Device Group (DG) Name: conbcv
DG's Type           : REGULAR
DG's Symmetrix ID    : 000184502237
```

Standard Device			BCV Device			State
Logical	Sym	Inv. Tracks	Logical	Sym	Inv. Tracks	STD <=> BCV
DEV001	0355	0	BCV001	035F *	127806	SyncInProg
		0	BCV002	0360 *	127687	SyncInProg
DEV002	0356	0	BCV003	0361 *	130115	SyncInProg
		0	BCV004	0362 *	130084	SyncInProg
Total (Primary)		-----			-----	
Track(s)		0			257921	
MB(s)		0.0			8060.0	

Legend:

(*): The paired BCV device is associated with this group.

The `symmir verify` command displays a message every five seconds until the BCV pairs are in a Synchronized state.

```
# symmir -g conbcv verify -synched -concurrent -i 5
```

None of the devices in group 'conbcv' are in the 'Synchronized' state.

None of the devices in group 'conbcv' are in the 'Synchronized' state.

Not all devices in group 'conbcv' are in the 'Synchronized' state.

Not all devices in group 'conbcv' are in the 'Synchronized' state.

All devices in group 'conbcv' are in the 'Synchronized' state.

The `symmir query -multi` command confirms that the concurrent BCV pairs are in the Synchronized state.

```
# symmir -g conbcv query -multi
```

```
Device Group (DG) Name: conbcv
DG's Type           : REGULAR
DG's Symmetrix ID   : 000184502237
```

Standard Device			BCV Device			State
Logical	Sym	Inv. Tracks	Logical	Sym	Inv. Tracks	STD <=> BCV
DEV001	0355	0	BCV001	035F *	0	Synchronized
			BCV002	0360 *	0	Synchronized
DEV002	0356	0	BCV003	0361 *	0	Synchronized
			BCV004	0362 *	0	Synchronized
Total (Primary)		-----			-----	
Track(s)		0			0	
MB(s)		0.0			0.0	

Legend:

(*): The paired BCV device is associated with this group.

The `symmir split` command performs an instant split operation on all BCV pairs in the device group. Using the `-instant` option causes SYMCLI to return immediately after the foreground split, while the background split is still in progress.

```
# symmir -g conbcv split -instant -noprompt
```

'Split' operation execution is in progress for device group 'conbcv'.
Please wait...

'Split' operation successfully executed for device group 'conbcv'.

The `symmir query` command displays the concurrent BCVs and the status of the background split (-bg). The Split (bg) state means that the background split is still in progress.

```
# symmir -g conbcv query -multi -bg
```

```
Device Group (DG) Name: conbcv
DG's Type           : REGULAR
DG's Symmetrix ID   : 000184502237
```

Standard Device			BCV Device			State	
Logical	Sym	Inv. Tracks	Logical	Sym	Inv. Tracks	STD <=>	BCV
DEV001	0355	0	BCV002	0360 *	0	Split	(bg)
		0	BCV001	035F *	0	Split	(bg)
DEV002	0356	0	BCV004	0362 *	0	Split	(bg)
		0	BCV003	0361 *	0	Split	(bg)
Total (Primary)		-----			-----		
Track(s)		0			0		
MB(s)		0.0			0.0		

Legend:

(*): The paired BCV device is associated with this group.
 (bg): The paired BCV device is splitting in the background.

The `symmir verify` command checks every five seconds until the background split is complete.

```
# symmir -g conbcv verify -split -concurrent -i 5 -bg
```

None of the devices in group 'conbcv' have finished splitting in the background.

None of the devices in group 'conbcv' have finished splitting in the background.

Not all devices in group 'conbcv' have finished splitting in the background.

All devices in group 'conbcv' have finished splitting in the background.

The following command incrementally re-establishes the preferred BCV pairs that were created earlier with the attach action.

```
# symmir -g conbcv establish -noprompt -v
```

```
'Incremental Establish' operation execution is in progress for device group
'conbcv'. Please wait...
```

SELECTING Standard devices in the group:

```
Device: 0355 [SELECTED]
Device: 0356 [SELECTED]
```

SELECTING BCV devices associated with the group:

```
Device: 035F [SELECTED]
Device: 0360 [SELECTED]
Device: 0361 [SELECTED]
Device: 0362 [SELECTED]
```

PAIRING of Standard and BCV devices:

```
Devices: 0355(S) - 035F(B) [PAIRED]
Devices: 0356(S) - 0361(B) [PAIRED]
```

STARTING a BCV 'INCREMENTAL_ESTABLISH' operation.

The BCV 'INCREMENTAL_ESTABLISH' operation SUCCEEDED.

'Incremental Establish' operation successfully initiated for device group 'conbcv'.

The following query displays the status of the concurrent BCVs and that the attached pairs have been re-established (their state is Synchronized). Those BCVs that were not participants in the previous establish operation are still Split.

```
# symmir -g conbcv query -multi
```

```
Device Group (DG) Name: conbcv
DG's Type           : REGULAR
DG's Symmetrix ID    : 000184502237
```

Standard Device			BCV Device			State
Logical	Sym	Inv. Tracks	Logical	Sym	Inv. Tracks	STD <=> BCV
DEV001	0355	0	BCV001	035F *	0	Synchronized
			BCV002	0360 *	0	Split
DEV002	0356	0	BCV003	0361 *	0	Synchronized
			BCV004	0362 *	0	Split
Total (Primary)		-----			-----	
Track(s)		0			0	
MB(s)		0.0			0.0	

Legend:

(*): The paired BCV device is associated with this group.

The following command incrementally re-establishes the remaining concurrent BCVs.

```
# symmir -g conbcv establish DEV001 bcv ld BCV002 DEV002 bcv ld BCV004 -noprompt
```

'Incremental Establish' operation execution is in progress for the device list in device group 'conbcv'. Please wait...

'Incremental Establish' operation successfully initiated for the device list in device group 'conbcv'.

Querying the device group displays the status of the concurrent BCVs. All are now in a Synchronized state.

```
# symmir -g conbcv query -multi
```

```
Device Group (DG) Name: conbcv
DG's Type           : REGULAR
DG's Symmetrix ID   : 000184502237
```

Standard Device			BCV Device			State
Logical	Sym	Inv. Tracks	Logical	Sym	Inv. Tracks	STD <=> BCV
DEV001	0355	0	BCV002	0360 *	0	Synchronized
		0	BCV001	035F *	0	Synchronized
DEV002	0356	0	BCV004	0362 *	0	Synchronized
		0	BCV003	0361 *	0	Synchronized
Total (Primary)		-----			-----	
Track(s)		0			0	
MB(s)		0.0			0.0	

Legend:

(*): The paired BCV device is associated with this group.