
Using SYMCLI to Perform TimeFinder/Mirror Control Operations

Abstract

This paper describes how to use TimeFinder/Mirror software to create and split a BCV pair, to retrieve original BCV data from a fixed BCV mirror, and to perform a BCV action while making the BCV or the standard device temporarily inaccessible to the host.

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 300-000-074 REV J

Table of Contents

Introduction	4
Purpose and Scope	4
Related Documentation	4
Practical Uses	4
Establishing a BCV Pair	6
Splitting a BCV Pair	7
Splitting When a BCV Has Two Mirrors	8
Reverse Splitting When a BCV Has Two Mirrors	9
Protected BCV Establish When a BCV Has Two Mirrors.....	10
Restoring a BCV Pair	11
Protected BCV Restore.....	11
Using the -not_ready Option to Prevent Host Access	12
Using a Composite Group to Control a Set of BCV Pairs That Spans Multiple Symmetrix Arrays.....	13
Instant Control Response for Establish and Restore Operations	15
Clone Emulation.....	15
Example 1: Split Operations	17
Example 2: Using the split -not_ready Command.....	26
Example 3: Using a Composite Group.....	30
Appendix A: Clone Emulation Mapping.....	34

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*.

The *device group* and the *composite group* are entities that you can create and use to manage and control BCV pairs. Your host’s SYMAPI¹ database file or the GNS-managed global repository stores information about a device group or composite group and the devices contained in the group.

Beginning with Solutions Enabler version 5.4, you can create a composite group to control a set of BCV pairs that spans multiple Symmetrix arrays. A composite group provides greater flexibility than a device group, which can define devices only on a single Symmetrix array. Descriptions in this paper of operations on device groups are also relevant for composite groups.

Purpose and Scope

This document describes TimeFinder functionality included in versions of EMC[®] Solutions Enabler up through version 6.0 running on Symmetrix arrays using Enginuity[™] versions 5x65 to 5x67, 5568, 5669, 5670, and 5x71.

Related Documentation

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

- *Using SYMCLI to Perform TimeFinder/Snap Operations* (P/N 300-001-381)
- *Using SYMCLI to Perform TimeFinder/Clone Operations* (P/N 300-001-382)
- *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 Set Up TimeFinder/Mirror Multiple BCVs and Concurrent BCVs* (P/N 300-000-073)
- *Using SYMCLI to Perform Consistent Splits with TimeFinder Family Products* (P/N 300-000-283)

Practical Uses

Establishing BCV pairs make it practical for you to access the copied data on a BCV at any later point in time without interfering with business operations. Any time you split one of the BCVs from the standard device, the BCV has data that is available for backup, testing, analysis, or snapping (making instant copies).

Regular split has been traditional for TimeFinder since its inception. Applications attempting to access the standard or the BCV would sometimes experience a short wait during a regular split because of the time it took to complete the regular split in full. Once the split was complete, no further overhead was incurred. However, beginning with Enginuity version 5568, *any* split operation is an instant split.

¹ Symmetrix API

Instant split reduces the wait period during a split to a short time by decomposing the process into two phases: a *foreground split* and a *background split*. When splitting large numbers of heavily accessed volumes at one time using instant split, SYMCLI cuts the application wait times from several seconds (as was common with regular split) to a fraction of a second. During an instant split, the system performs the splitting of the mirrors *after* a foreground split is declared, and the system makes the BCV pair accessible almost immediately to their respective hosts for processing.

You can configure a BCV to have two local mirrors to guard against data loss through physical drive failure of one of the BCV mirrors. *Reverse split* is applicable for a BCV that is configured to have two local mirrors and is generally used to recover from an unsuccessful *restore* operation. When data is restored from the BCV to the standard device, any writes that occur while the standard and BCV are joined result in the alteration of the original copy of data on the BCV's primary mirror. If you need that original copy of BCV data again later, you can restore it to the BCV's primary mirror from the BCV's secondary mirror using a reverse split. For example, whenever logical corruption is reintroduced to a database during a recovery process (following a BCV restore), both the standard device and the primary BCV mirror are left with corrupted data. In this case, a reverse split can restore the original BCV data from a BCV's secondary mirror to its primary mirror.

Protected BCV establish is also applicable for a BCV that is configured to have two local mirrors. It allows you to simultaneously move both mirrors of a BCV when establishing that BCV to a standard device as part of a BCV pair. Under normal circumstances, a BCV's secondary (fixed) mirror does not acquire the latest data until the BCV is split from the standard device. Then the primary (moving) mirror copies its changed data to the secondary mirror. Protected BCV establish allows both BCV mirrors to be moved to the standard device when the BCV pair is established, thus eliminating any wait period after the split operation for the secondary mirror to acquire the data and ensuring immediate protection of the data on the BCV in case of a disk failure.

Restoring a BCV pair is useful if you want to copy data from the BCV to its paired standard device. For example, if the results from running a new application on the BCV were satisfactory, you could port the data and the new application to the standard device.

Using the `-not_ready` option with the *split* and *restore* actions enables you to manipulate Windows 2000 disk signatures such that BCVs and standards do not present duplicate signatures to a Windows 2000 host. This option also facilitates BCV sharing among multiple cooperating TimeFinder processes.

Beginning with Solutions Enabler version 5.4, you can use a composite group to control a set of BCV pairs that spans multiple Symmetrix arrays. This feature is useful in multi-Symmetrix configurations where you want to copy data that spans more than one Symmetrix array.

Beginning with Solutions Enabler version 6.0, you can control specific BCV pairs within a composite group instead of having to operate on the entire group. This allows composite groups the same granularity as device groups, although with the added scope of being able to span multiple Symmetrix arrays.

Beginning with Solutions Enabler version 6.0, TimeFinder can initiate the pre-copying of data to the RAID-5 BCV by automatically mapping TimeFinder/Mirror commands to existing TimeFinder/Clone commands when it encounters a RAID-5 BCV. If non-RAID-5 BCVs exist in a mixed Symmetrix environment with RAID-5 BCVs, you need to explicitly set clone emulation mode.

Establishing a BCV Pair

A business continuance sequence first involves establishing the BCV device as a mirror of a standard Symmetrix device. While the BCV is mirroring the standard device, the BCV is inaccessible to its host (“Not Ready” in Figure 1). You can split the BCV from the standard device at any time, making the BCV accessible again to its host. After the split, the BCV contains valid data and is available for business continuance tasks or restoring (copying) data back to the standard device if data was lost there.

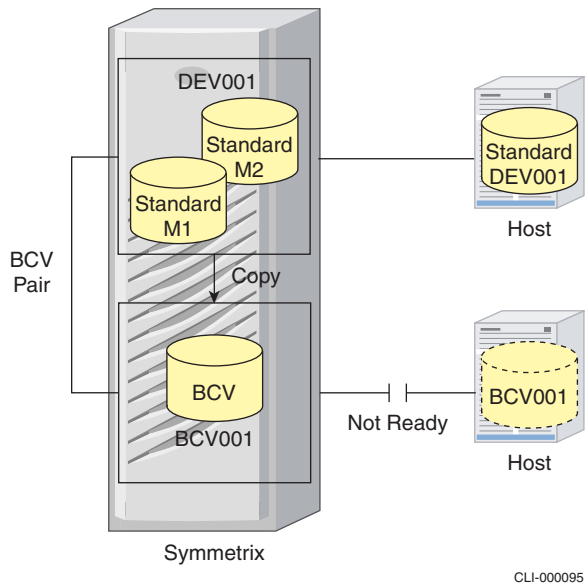


Figure 1. Establishing a BCV Pair

Symmetrix allows up to four mirrors for each logical volume. The mirror positions are commonly designated M1, M2, M3, and M4. A single BCV can be the second, third, or fourth mirror of the standard device. In Figure 1, because standard device DEV001 is configured with two mirrors, BCV001 functions as the third mirror. A host logically views the Symmetrix M1/M2 mirrored devices as a single device.

To assign a BCV as a mirror of a standard Symmetrix device, use the `symmir establish` command.² One method of establishing a BCV pair is to allow the standard/BCV device-pairing algorithm to create BCV pairs from multiple devices within a device group (ProdBgrp, for example):

```
symmir -g ProdBgrp -full establish
```

With this method, TimeFinder first checks for any `attach` assignments (specifying a preferred BCV match from among multiple BCVs in a device group). TimeFinder then checks if there are any previous (current) pairing relationships among the devices. If either of these previous conditions exists, TimeFinder uses these assignments. If not, the algorithm pairs standards and BCVs of equal sizes.

Other methods of establishing BCV pairs that override any current pairing relationships involve using the `-exact` option, the `-opt` option, or explicit device names on the command line. For details about establishing BCV pairs using these other options, refer to the white paper *Using SYMCLI to Set Up TimeFinder/Mirror BCV Pairs* (P/N 300-000-072).

² The `symmir restore` command also creates a BCV pair when you need to restore (copy) data from the BCV to the standard. For more details, refer to the section “Restoring a BCV Pair.”

In the SYMAPI options file you can set a `SYMAPI_SYNC_DIRECTION` variable that allows SYMCLI to accept only establish operations and reject any attempt to perform a restore operation, or vice versa. The default is `BOTH`. To restrict SYMCLI to establish operations, set the variable to `ESTABLISH`:

```
SYMAPI_SYNC_DIRECTION = ESTABLISH
```

Setting the variable to `ESTABLISH` may be useful if, for example, you have a host on which you normally perform only establish operations and you want to block any attempt on that host to perform restores.

Splitting a BCV Pair

Splitting a BCV pair is a TimeFinder action that reassigns to the BCV its original independent host address and allows its contents to be used for business continuance tasks. When splitting a BCV, the system needs to perform housekeeping tasks that may require a few seconds on a busy Symmetrix array. These tasks involve a series of steps that result in separation of the BCV from its paired standard:

1. I/O is suspended to the standard device.
2. Write pending tracks to the standard device that have not yet been written out to the BCV are duplicated in cache memory to be written to the BCV.
3. The BCV is split from the standard device.
4. The BCV device status is changed to Ready.

With a regular split (prior to Enginuity version 5568), the standard device did not resume front-end I/O activity until the entire split sequence completed.

During an *instant split*, SYMCLI decomposes the splitting process into two phases: a foreground split and a background split. When the `symmir split` command is issued, the system executes the foreground split almost instantaneously and returns a successful status to the host. This instantaneous execution allows I/O operations to the standard and BCV (for example, host applications such as databases) to continue immediately. At the same time, the background split continues to split the BCV pair until the split is complete. If a host application tries to access a data track on a BCV or standard that has not completed splitting, the Symmetrix array splits the track being accessed and completes the host I/O request.

Beginning with Enginuity version 5568, all splits are instant splits. If you omit the `-instant` option on the `symmir split` command line, SYMCLI waits for the background split to complete before returning. If you include the `-instant` option, SYMCLI returns immediately after the foreground split, allowing you to perform other TimeFinder operations while the BCV pair is splitting in the background. However, the following restrictions on performing TimeFinder operations apply while the system is processing a background split on a BCV pair:

- You cannot restore any BCV to the standard being split,
- You cannot establish (full or incremental) the standard with another BCV device,
- If you incrementally re-establish the splitting pair, any tracks that are still splitting in the background are marked invalid on the BCV device and have to be copied.

The following operation performs an instant split on all BCV pairs in the group named `ProdBgrp` and allows SYMCLI to return while the background split is in progress. The `symmir query` command checks the progress of the background split (`-bg`):

```
symmir -g ProdBgrp split -instant
symmir -g ProdBgrp query -bg
```

The SYMAPI configuration directory contains an options file that allows you to set a parameter that causes your script or application calls to wait for the background split operation to complete before returning to

your script or program. Setting the following parameter to `TRUE` causes the wait to occur regardless of whether the `symmir split` command includes the `-instant` option:

```
SYMAPI_WAIT_FOR_BCV_BG_SPLIT = TRUE
```

If you want to remove a default option from the options file, you can either remove the line entry from the file or comment the line by adding a pound (#) sign at the beginning of the line entry. For a description of the options file, refer to the *EMC Solutions Enabler Symmetrix Base Management CLI Product Guide*.

The TimeFinder product family also provides a consistent-split implementation of instant split that allows you to split off a consistent, DBMS-restartable BCV copy of your database without having to quiesce or shut down the database first. The consistent split is accomplished with negligible impact on your production environment. This implementation, called Engenuity Consistency Assist (ECA), performs consistent split across multiple, heterogeneous hosts without the PowerPath® support that was required prior to Solutions Enabler version 5.1. For more information about consistent split, refer to the white paper *Using SYMCLI to Perform Consistent Splits with TimeFinder Family Products* (P/N 300-000-283).

Splitting When a BCV Has Two Mirrors

As mentioned previously, you can configure a BCV to have two mirrors to guard against data loss through physical drive failure of one of the BCV mirrors. In a BCV establish and split, the BCV's primary mirror (also called the *moving mirror*) is established with the standard device while its secondary mirror (also called the *fixed mirror*) remains as it was before the establish operation. The fixed mirror is updated with the new data only *after* the BCV is split from the standard device.

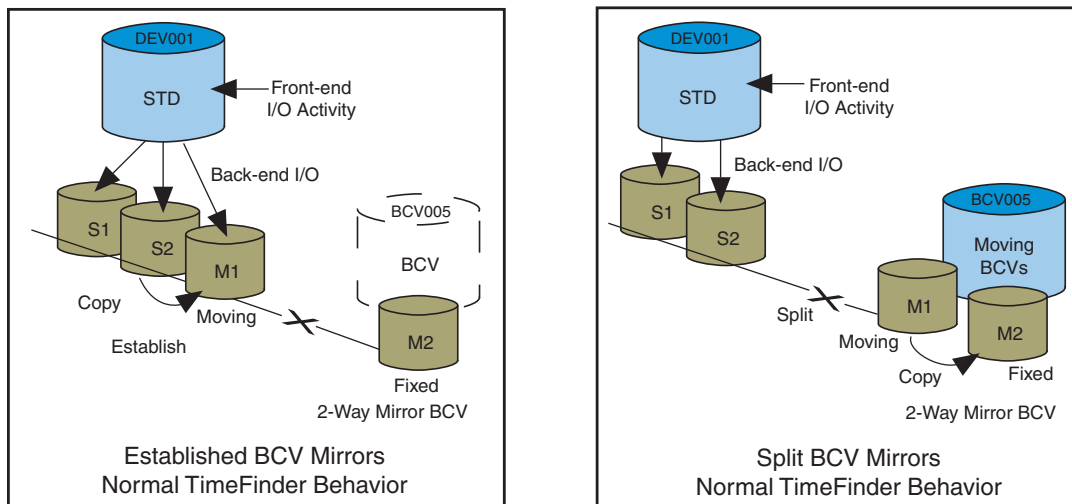


Figure 2. Behavior of BCV Mirrors During a Normal Split

In Figure 2, M1 and M2 are BCV005's mirrors, and S1 and S2 are the DEV001 standard mirrors. When the BCV is paired with a standard device (as BCV005 is paired with DEV001 in Figure 2), the BCV's moving mirror "moves" to join the mirrors of the standard. The BCV's fixed mirror stays behind and is updated from the moving mirror only *after* the BCV splits from the standard.

The BCV's moving mirror does not copy to its fixed mirror until the background split completes. Then it copies only tracks on the moving mirror that changed while the two mirrors were separated.

Reverse Splitting When a BCV Has Two Mirrors

When a BCV is configured to have two mirrors and you want to synchronize the two BCV mirrors after a split, the system normally copies data from the BCV's moving mirror to its fixed mirror. The `symmir split` command with the `-reverse` option, however, reverses the normal copy process between them. A reverse split operation causes the system to copy data from the fixed mirror to the moving mirror, allowing you to revert to a copy of the data that was on the BCV *before* it was synchronized with the standard.

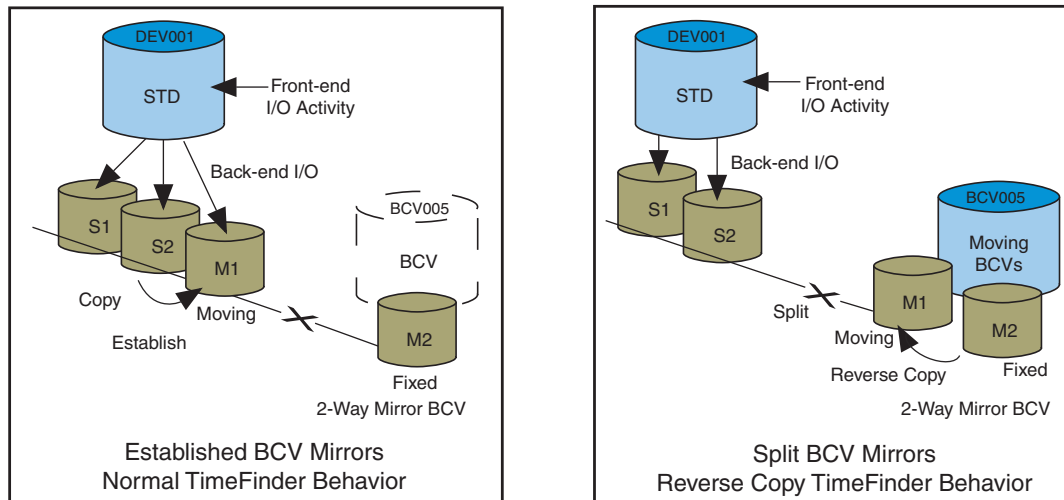


Figure 3. Behavior of BCV Mirrors During a Reverse Split

Reverse splits are not permitted if the moving mirror and fixed mirror were not fully synchronized when the moving mirror was established or restored with the standard, or if there were Write Pending tracks to the BCV at the time of the establish or restore operation. You can use the `-reverse` option with establish and restore operations to check whether both BCV mirrors are synchronized before initiating either operation. If the BCV mirrors are not fully synchronized, SYMCLI rejects the establish or restore operation. To retry the operation at specified time intervals until these conditions are met, you can use the `-i` option. For example:

```
symmir -g ProdBgrp -reverse -i 30 -c 10 establish
```

This command checks every 30 seconds. If after five minutes (10 retries) the conditions are not met, TimeFinder rejects the establish operation. Without the use of the `-c` argument, TimeFinder would retry the establish operation indefinitely until the conditions were met.

You can also use `symmir verify -bcv_mirrors` to check if both BCV mirrors are synchronized:

```
symmir -g ProdBgrp verify -bcv_mirrors
```

This is useful in a script to test BCV mirror synchronization before starting an establish or restore operation. Use the `-i` and `-c` options here to specify a retry interval in seconds and a retry count (the number of times to repeat the retry).

The following command performs a reverse split on all BCV pairs in the group, but does not initiate copying from the BCV's fixed mirror to its moving mirror until the background split completes:

```
symmir -g ProdBgrp split -reverse
```

Figure 4 illustrates a practical use of a reverse split. At midnight a split results in a good point-in-time copy of a database. At 10 A.M. a corruption is discovered in the database, necessitating a database recovery. At 10:30 A.M. a restore operation is initiated from the BCV copy. Because the good data is immediately available to the BCV pair, the recovery begins shortly after initiating the restore process. At 11:00 A.M., during the recovery, one of the logs re-corrupts the database. Though the data on the BCV's moving mirror has changed during the recovery process, a reverse split can be initiated. At 11:20 A.M. the BCV's fixed mirror refreshes its moving mirror, providing access to the good midnight copy of the data.

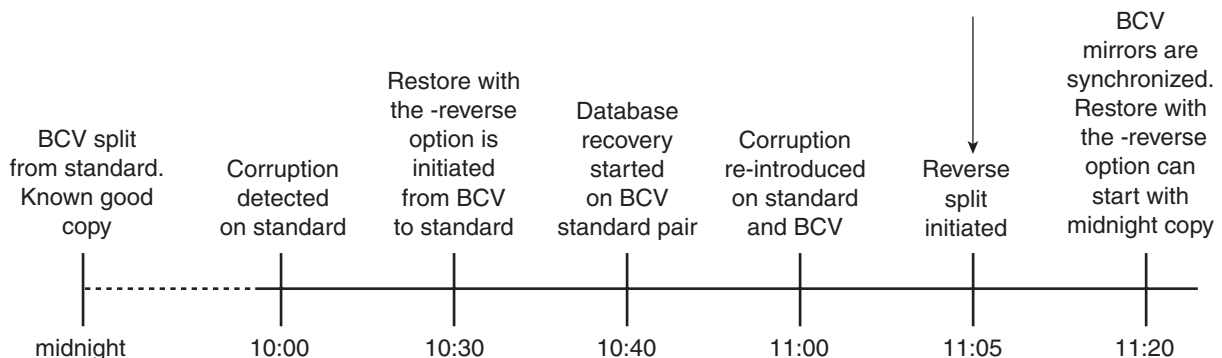


Figure 4. Practical Use of a Reverse Split

Protected BCV Establish When a BCV Has Two Mirrors

Protected BCV establish is a feature that allows you to simultaneously move both mirrors of a BCV to the standard device when establishing that BCV as part of a BCV pair. Under a normal establish, a BCV's fixed mirror does not acquire the latest data until after the BCV is split from the standard device. Then the moving mirror copies its changed data to the fixed mirror, thus insuring immediate protection of the data on the BCV in case of a disk failure.

Protected BCV establish causes both BCV mirrors to be synchronized with the standard device when the BCV pair is established, thus eliminating the extra time after the split operation for the fixed mirror to become synchronized with the moving mirror. With this protection option, both mirrors of the split BCV always have the same data at the same time.

The `symmir establish` command with the `-protbcvest` option moves both BCV mirrors to the BCV's paired standard device. For example:

```
symmir -g ProdBgrp establish -protbcvest
```

The following `symmir split` command performs a consistent split on all BCV pairs in the device group. No special option is required to split BCV pairs that were synchronized using protected BCV establish.

```
symmir -g ProdBgrp split -consistent
```

A *consistent split* is a special implementation of instant split that allows you to avoid inconsistencies and restart problems that can occur if you split a database-related BCV without first quiescing the database or putting it in hot backup mode. The difference between a normal instant split and a consistent instant split is the use of the `-consistent` option to suspend the database device writes for a very short time while the foreground split occurs. For more information about consistent split, refer to the white paper *Using SYMCLI to Perform Consistent Splits with TimeFinder Family Products* (P/N 300-000-283).

Restoring a BCV Pair

A restore operation synchronizes a BCV pair but copies the contents of the BCV device *to* the standard device, thus reversing the copy direction of an establish operation. A full restore operation (which requires the `-full` option) copies the entire contents of the BCV to the standard device. An incremental restore operation (omitting the `-full` option) saves time by copying to the standard device only the new data that was written to the BCV by its host while the BCV pair was split. Also, any tracks on the standard device that changed during the split are overwritten by corresponding tracks on the BCV.

One of TimeFinder's advantages with respect to a restore operation is its ability to make the restore appear instantaneous. Once you initiate a restore from the BCV to the standard device, data from the BCV is immediately available to a host accessing the BCV pair. During the time the pair are joined, writes are sent to the standard and the BCV, while reads are satisfied by the data on the BCV if the data on the standard has not yet been completely updated from the BCV. However, this process can alter the BCV data during the restore operation. If you want to retain the original BCV data, use protected restore described below.

To initiate an incremental restore operation on all BCV pairs in device group `ProdEgrp`, use this command:

```
symmir -g ProdEgrp restore
```

In the options file, you can set a `SYMAPI_SYNC_DIRECTION` variable that allows SYMCLI to accept only establish operations and reject any attempt to perform a restore operation, or vice versa. The default is `BOTH`. To restrict SYMCLI to only `symmir restore` commands, set the variable to `RESTORE`:

```
SYMAPI_SYNC_DIRECTION = RESTORE
```

Setting the variable to `RESTORE` may be useful if, for example, you have a host on which you normally perform only restore operations and you want to block any attempt on that host to perform an establish operation. On the other hand, if you want to restrict SYMCLI to accepting only `symmir establish` commands, set this variable to `ESTABLISH`.

Protected BCV Restore

As mentioned above, a regular restore can alter the BCV data during the restore operation. Beginning with Solutions Enabler version 5.4, you can use protected restore if you want to retain the original BCV data during the restore process, while the host is reading and writing to the standard device. With the protected BCV in the Restored state, its associated standard device cannot copy data to it despite their current pairing relationship. Also, you cannot pair the standard device with another BCV until you split the protected BCV.

Protected restore offers the same benefit as reverse split without requiring the use of a mirrored BCV. However, mirrored BCVs may still be desirable to protect the BCV device against physical drive failure.

To initiate a protected restore operation, use the `-protect` option:

```
symmir -g ProdEgrp -protect restore
```

You also need the `-protect` option to split a protected BCV:

```
symmir -g ProdEgrp -protect split
```

The `-protect` option is also necessary to check on the status of protected BCVs during query and verify operations:

```
symmir -g ProdEgrp -protect -restored verify
```

```
symmir -g ProdEgrp -protect query
```

Using the `-not_ready` Option to Prevent Host Access

At the end of a split operation, the BCV becomes accessible to the host. To make the BCV inaccessible to the host after the split, use the `split -not_ready` option. Similarly, using the `-not_ready` option with a restore action makes the *standard* inaccessible upon initiating the restore operation. There are several situations where using this option may be useful:

- When a BCV is established with a standard, all information (including volume manager specific information such as a Windows 2000 device signatures) is copied to the BCV. While the standard and BCV are synchronized, this is not a problem because the BCV is not ready (not visible to the host). However, when the BCV is split, if the host accessing the standard can also access the now ready (host-visible) BCV, the Windows Disk Administrator incurs a problem. When it detects a duplicate signature, Windows 2000 arbitrarily places one of the drives offline. To avoid this problem, you can make the BCV device temporarily inaccessible while in the split condition by using a `symmir split -not_ready` command. Subsequently, after relabelling the BCV with a new signature, you can use the `symld ready` command to make the BCV accessible again.
- A similar situation arises when a restore operation overwrites the contents of the standard volume (including the signature) with the contents of the BCV. Using the `restore -not_ready` action allows you to relabel the standard before making it accessible to the host. The `-not_ready` option is also useful when you want to wait until the restore operation completes before using the standard.
- When TimeFinder control operations are performed on a Symmetrix by more than one process, regardless of whether they are executing on one or more hosts, there is a danger that one application will accidentally usurp a BCV that contains data belonging to a different application. Using the `split -not_ready` command can make it easier to prevent accidental misuse.

The `symlabel` command works on Symmetrix FBA (Fixed Block Architecture) devices on Windows 2000 to avoid the label conflict problem described above. To examine the emulation mode of a Symmetrix device, use the `symdev show` command. The following sequence outlines how to establish and split a BCV pair (DEV001/BCV001) and suspend read/write operations to the BCV until you have relabeled the BCV device:

1. Define device signatures for the standard and BCV devices. The signature that you specify must be an 8-character hexadecimal value (for example, 12ABCDEF). The `-type` parameter specifies that the operation is on a Windows 2000 system. If you omit the `label` option as in the BCV001 signature definition, SYMCLI reads the signature off the disk and stores it in the SYMAPI database.


```
symlabel -g ProdBgrp define -type WNT DEV001 label 12ABCDEF
symlabel -g ProdBgrp define -type WNT BCV001 -bcv
```
5. Establishing the BCV pair causes the BCV signature to become identical to the standard signature:


```
symmir -g ProdBgrp -full establish DEV001 bcv ld BCV001
```
6. After synchronization, split the pair so that the BCV device is in the Not Ready state after the split:


```
symmir -g ProdBgrp split -not_ready
```
7. Relabelling the BCV device causes the BCV signature to revert to its original signature that was stored in the SYMAPI database:

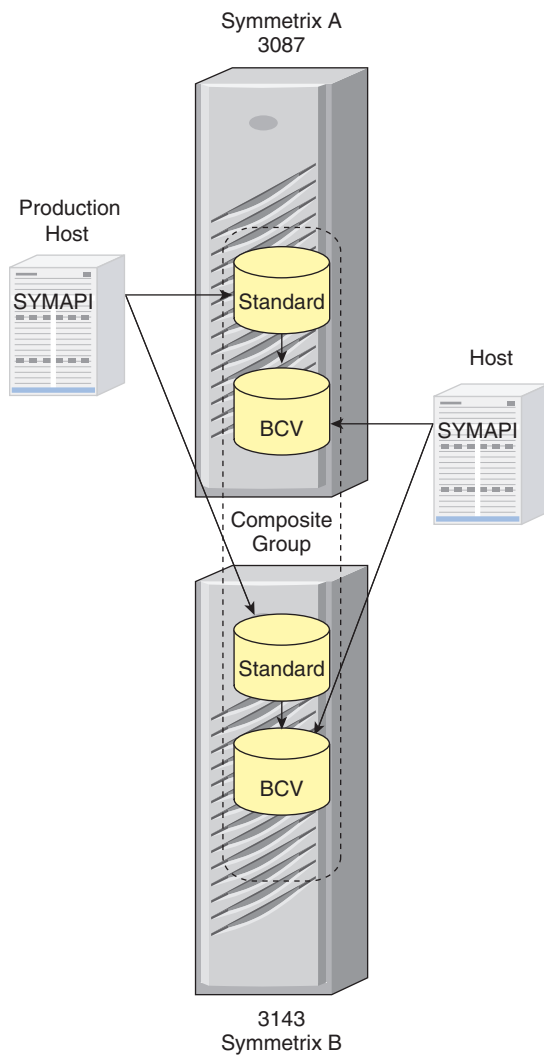

```
symld -g ProdBgrp relabel -bcv
```
8. Change the status of the BCV device to Ready:


```
symld -g ProdBgrp ready -bcv
```

Using a Composite Group to Control a Set of BCV Pairs That Spans Multiple Symmetrix Arrays

A *composite group* is a user-defined group of devices that can span multiple Symmetrix arrays. This feature provides greater flexibility than a device group, which can define devices only on a single Symmetrix. Beginning with Solutions Enabler version 6.0, you can control specific BCV pairs within the composite group instead of having to operate on the entire group as in previous versions.

Figure 5 illustrates a production host that is locally connected to two Symmetrix arrays (A and B). A composite group is defined on the production host and includes BCV pairs from each Symmetrix array. Another locally connected host allows you to access the BCVs once the BCV pairs are split.



CLI-000081

Figure 5. Using a Composite Group When a Set of Devices Spans Two Symmetrix Arrays

Although TimeFinder control operations on BCV pairs might normally be performed from the production host (as shown in Figure 5) because the composite group is defined there in its SYMAPI database, there are methods that would allow you to initiate copy sessions from another locally connected host. One way is to copy the composite group definition to another host. A more efficient method is to enable Group Name Services (GNS), which automatically propagates the composite group definition to the Symmetrix arrays

and other locally attached hosts that are running the GNS daemon. For more information, refer to the white paper *Using SYMCLI and GNS to Propagate Group Definitions to Multiple Hosts* (P/N 300-301-384).

Prior to Solutions Enabler version 6.0, composite group support did not allow you to copy from a single source standard device to more than one BCV (that is, not to concurrent BCVs). Consequently, you had to add an equal number of standard devices and BCV devices to the composite group. Solutions Enabler version 6.0 lifts this restriction by allowing you to create each BCV pair explicitly, including concurrent BCVs.

If you do not create each BCV pair explicitly, certain options such as `-opt`, `opt_rag`, and `-exact` allow you to control how multiple devices in a composite group are paired. Otherwise, a device-pairing algorithm checks if there were any previous pair assignments among the devices and, if not, pairs standards and BCVs of equal sizes.

Used only with a full establish operation, the optimize option that you choose depends on whether you are establishing local BCV pairs or remote BCV pairs. The `-opt` option is for local. It optimizes pairings across the local Symmetrix array without regard for whether the devices belong to different RDF (RA) groups. The `-opt_rag` option is for remote and requires the `-rdf` option. It uses optimization rules to create remote BCV pairs from devices within the same RDF (RA) group on a Symmetrix array.

The following steps outline the setup required for controlling a set of BCV pairs that spans two Symmetrix arrays as shown in Figure 5:

1. From the production host, create a Regular type composite group (for example, MyGrp).

```
symcgs create MyGrp -type regular
```
2. Add to the composite group those standard devices on Symmetrix A (3087) and Symmetrix B (3143) that are the source devices.

```
symcgs -cg MyGrp -sid 3087 add dev 0076
symcgs -cg MyGrp -sid 3143 add dev 0091
```
3. Add a BCV device from each Symmetrix array to the composite group.

```
symbcv -cg MyGrp -sid 3087 associate dev 0051
symbcv -cg MyGrp -sid 3143 associate dev 004F
```
4. Create the BCV pairs and initiate full copying from the standards to the BCVs.

```
symmir -cg MyGrp establish -full
```
5. When the BCV pairs are fully synchronized, you can split all BCV pairs in the composite group in order to access the BCVs.

```
symmmir -cg MyGrp split
```

Beginning with Solutions Enabler version 6.0, you can control specific BCV pairs within the composite group instead of having to control the group as a whole. For example, to establish only the DEV001/BCV001 pair from all devices in the composite group MyGrp:

```
symmir -cg MyGrp establish DEV001 bcv 1d BCV001
```

Instant Control Response for Establish and Restore Operations

Beginning with Solutions Enabler version 6.0 and Enginuity version 5x71, `symmir establish` and `symmir restore` become instant operations. That is, Enginuity version 5x71 performs the initiation much faster than in previous versions and returns control faster to the SYMCLI. This instant return of control allows you to perform other activities while the BCV pair is initiating establish or restore operations in the background.

To check the progress of the background restore/establish operation and display the percent initiated, execute the `symmir query -percent -bg` command.

Clone Emulation

Beginning with Solutions Enabler version 6.0, TimeFinder automatically maps TimeFinder/Mirror commands to existing TimeFinder/Clone commands when the BCV is a RAID-5 BCV, provided that no non-RAID-5 BCVs exist in the devices being controlled. In a mixed environment, you must explicitly set clone emulation mode. Under clone emulation mode, TimeFinder can initiate the concurrent pre-copying of data to all members of the RAID-5 BCV.

When you establish a BCV pair under clone emulation, the `symmir establish -full` command maps to the `symclone create -precop` command. This action causes copying to begin while checking still for new writes. The `symmir split` command maps to the `symclone activate` command. This action causes the data to become available to the host as an instant point-in-time copy.

For a complete list of mappings of `symmir` operations to `symclone` operations, refer to Appendix A.

To operate in a mixed BCV set of RAID-5 BCVs and non-RAID-5 BCVs, you must set the clone emulation environment variable to `ENABLED`.

```
SYMCLI_CLONE_EMULATION=ENABLED
```

In a *mixed* BCV set, if clone emulation is *disabled* (the default), any control or list operation produces an error when a RAID-5 BCV is encountered.

When in clone emulation mode, a standard device can be paired with as many as eight concurrent BCVs. Therefore, you can issue the `symmir establish -concurrent` command for the same standard device up to eight times, adding one additional BCV each time. For information on establishing concurrent BCVs, refer to the white paper *Using SYMCLI to Set Up TimeFinder/Mirror Multiple BCVs and Concurrent BCVs* (P/N 300-000-073).

In clone emulation mode, the following limitations exist:

- Reverse split operations are not allowed.
- Protected BCV establish operations are implicit.
- Restore operations will be protected restore.
- An incremental restore operation can only be accepted if all tracks were copied from the source device prior to initiating the restore operation.
- The option file setting of the `SYMAPI_DEFAULT_BCV_SPLIT_TYPE` variable is ignored.
- Any snap or clone operation to an emulated device will not work.
- The maximum number of BCVs that can be incrementally established with an emulated standard device is eight (as opposed to the normal TimeFinder maximum of sixteen).
- The BCV pair states of Split-Before-Restore and Split-Before-Sync are not valid states of emulation devices. In either case, a `symmir split` command with the `-symforce` option will complete a split operation that returns a message relevant to these states.

For information about clone operations, refer to the white paper *Using SYMCLI to Perform TimeFinder/Clone Operations* (P/N 300-001-382).

Example 1: Split Operations

The examples below 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 such as backups.

Creating a device group and adding devices to it are prerequisites for establishing BCV pairs. The `symdbg` command creates a device group (ProdBgrp). The `symld` commands add standard devices to the group, using a device's Symmetrix device (dev) name. The `symbcv` command associates one or all BCVs with the device group; the `-range` option can be used with the `associateall` action to limit the selection to the BCVs that are within the specified range.

```
# symdbg create ProdBgrp
# symld -g ProdBgrp -sid 3264 add dev 84
# symld -g ProdBgrp -sid 3264 add dev 85
# symld -g ProdBgrp -sid 3264 add dev 86
# symld -g ProdBgrp -sid 3264 add dev 87
# symbcv -g ProdBgrp -sid 3264 -RANGE 88:8B associateall dev
# symdbg show ProdBgrp
```

Group Name: ProdBgrp

```
Group Type                : REGULAR
Device Group in GNS       : Yes
Valid                     : Yes
Symmetrix ID              : 000000003264
Group Creation Time       : Tue Jan  6 11:17:08 2004
Vendor ID                 : EMC Corp
Application ID            : SYMCLI
```

```
Number of STD Devices in Group : 4
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 (4):

LdevName	PdevName	Sym Dev	Att.	Sts	Cap (MB)
DEV001	/dev/rdisk/c4t0d0s2	0084		RW	516
DEV002	/dev/rdisk/c4t0d1s2	0085		RW	516
DEV003	/dev/rdisk/c4t0d2s2	0086		RW	516
DEV004	/dev/rdisk/c4t0d3s2	0087		RW	516

BCV Devices Locally-associated (4):

LdevName	PdevName	Sym Dev	Att.	Sts	Cap (MB)

BCV001	/dev/rdisk/c4t1d0s2	0088	RW	516
BCV002	/dev/rdisk/c4t1d1s2	0089	RW	516
BCV003	/dev/rdisk/c4t1d2s2	008A	RW	516
BCV004	/dev/rdisk/c4t1d3s2	008B	RW	516

}

The `symmir establish` command matches BCV pairs using the `-exact` option.

```
# symmir -g ProdBgrp -full -exact -v establish -noprompt
```

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

SELECTING Standard devices in the group:

```
Device: 0084 [SELECTED]
Device: 0085 [SELECTED]
Device: 0086 [SELECTED]
Device: 0087 [SELECTED]
```

SELECTING Target devices in the group:

```
Device: 0088 [SELECTED]
Device: 0089 [SELECTED]
Device: 008A [SELECTED]
Device: 008B [SELECTED]
```

PAIRING of Standard and BCV devices:

```
Devices: 0084(S) - 0088(B) [PAIRED]
Devices: 0085(S) - 0089(B) [PAIRED]
Devices: 0086(S) - 008A(B) [PAIRED]
Devices: 0086(S) - 008B(B) [PAIRED]
```

STARTING a BCV 'ESTABLISH' operation.

The BCV 'ESTABLISH' operation SUCCEEDED.

'Full Establish' operation successfully initiated for device group 'ProdBgrp'.

The `symmir verify` command verifies when the BCV pairs reach the Synchronized state. A new message is displayed every 30 seconds until the pair is established. Then the verify loop ends automatically. The time to reach the Synchronized or Restored state varies with the number of devices being established and the amount of data being copied to each BCV.

```
# symmir -g ProdBgrp -i 30 verify
```

None of the devices in group 'ProdBgrp' are in the 'Synchronized or Restored' state.

None of the devices in group 'ProdBgrp' are in the 'Synchronized or Restored' state.

Not all devices in group 'ProdBgrp' are in the 'Synchronized or Restored' state.

All devices in group 'ProdBgrp' are in the 'Synchronized or Restored' state.

The following command performs an instant split on BCV pairs in a device group ProdBgrp while the group is being subjected to a high write load (70 percent write, 30 percent read). In UNIX, the `timex` command measures the time it takes to execute a command. The "real" time is the elapsed time. The foreground split completed in 0.90 seconds.

```
# timex symmir -g ProdBgrp split -noprompt -instant
```

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

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

```
real      0.90
user      0.22
sys       0.05
```

The `symmir verify` command with the `-split` option confirms that the foreground split is complete.

```
# symmir -g ProdBgrp verify -split
```

All devices in group 'ProdBgrp' are in the 'Split' state.

The following `symmir query` command uses the `-bg` option to display the status of the background split. The background split is still continuing, indicated by a state of Split (bg).

```
# symmir -g ProdBgrp -bg 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
DEV001	0084	0	BCV001	0088 *	0	Split	(bg)
DEV002	0085	0	BCV002	0089 *	0	Split	(bg)
DEV003	0086	0	BCV003	008A *	0	Split	(bg)
DEV004	0087	0	BCV004	008B *	0	Split	(bg)
Total		-----			-----		
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 the status of the background split every 10 seconds until it completes.

```
# symmir -g ProdBgrp verify -split -bg -i 10
```

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

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

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

The `symmir query -bg` command confirms that the background split is complete (state is Split).

```
# symmir -g ProdBgrp query -bg
```

```
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
DEV001	0084	0	BCV001	0088 *	0	Split
DEV002	0085	0	BCV002	0089 *	0	Split
DEV003	0086	0	BCV003	008A *	0	Split
DEV004	0087	0	BCV004	008B *	0	Split
Total	-----				-----	
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.

To demonstrate the effect of a reverse split on a BCV that is configured to have two mirrors, the following commands temporarily create a new group called ProdAgrp and move one standard and one BCV to this device group. This action takes both devices out of ProdBgrp and places them in ProdAgrp.

```
# symdbg create ProdAgrp -type regular
# symld -g ProdBgrp move DEV001 ProdAgrp
# symbcv -g ProdBgrp move ld BCV001 ProdAgrp
# symdbg show ProdAgrp
```

Group Name: ProdAgrp

```
Group Type                : REGULAR
Device Group in GNS       : Yes
Valid                     : Yes
Symmetrix ID              : 000000003264
Group Creation Time       : Tue Jan  6 18:19:05 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       :    1
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              Sts      (MB)
-----
DEV001            /dev/rdisk/c4t0d0s2    0084          RW      516
}

```

BCV Devices Locally-associated (1):

```

{
-----
LdevName          PdevName          Sym      Att.  Sts      Cap
Dev              Dev              Dev              Sts      (MB)
-----
BCV001            /dev/rdisk/c4t1d0s2    0088          RW      516
}

```

Because Symmetrix devices 84 and 88 have a previous pairing relationship, it is possible to incrementally establish these devices with the `symmir establish` command.

```
# symmir -g ProdAgrp -noprompt establish
```

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

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

The `newfs` command creates a file system on the standard volume `/dev/rdisk/c4t0d0s2`.

```
# newfs /dev/rdisk/c4t0d0s2
```

newfs: /dev/rdisk/c4t0d0s2 last mounted as /mnt

newfs: construct a new file system /dev/rdisk/c4t0d0s2: (y/n)? **y**

/dev/rdisk/c4t0d0s2: 1054080 sectors in 1098 cylinders of 15 tracks, 64 sectors
514.7MB in 69 cyl groups (16 c/g, 7.50MB/g, 3584 i/g)

super-block backups (for fsck -F ufs -o b=#) at:

```

32, 15456, 30880, 46304, 61728, 77152, 92576, 108000, 123424, 138848, 154272,
169696, 185120, 200544, 215968, 231392, 245792, 261216, 276640, 292064,
307488, 322912, 338336, 353760, 369184, 384608, 400032, 415456, 430880,
446304, 461728, 477152, 491552, 506976, 522400, 537824, 553248, 568672,
584096, 599520, 614944, 630368, 645792, 661216, 676640, 692064, 707488,
722912, 737312, 752736, 768160, 783584, 799008, 814432, 829856, 845280,
860704, 876128, 891552, 906976, 922400, 937824, 953248, 968672, 983072,
998496, 1013920, 1029344, 1044768,

```

```
# mkdir /stdmnt
```

The `mount` command mounts the standard device to an existing mount point.

```
# mount /dev/dsk/c4t0d0s2 /stdmnt
```

The `cp` command copies a large file named `bigfile` to the BCV pair.

```
# cp /mnt/bigfile /stdmnt
```

```
# ls -l /stdmnt
```

```
total 989984
-rw-r--r--   1 root      other    506601472 Mar 13 18:21 bigfile
drwx-----   2 root      root      8192 Mar 13 17:41 lost+found
```

```
# umount /stdmnt
```

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. If you include the `-instant` option, SYMCLI returns immediately after the foreground split, allowing you to perform other TimeFinder operations while the BCV pair is splitting in the background.

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

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

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

The `symdev show` command on the BCV displays the number of tracks that have to be copied from the BCV's moving mirror (Mirror 1) to its fixed mirror (Mirror 2). Because the tracks on Mirror 2 have to be updated, the invalid tracks show up in the second mirror position displayed at the line "Mirror Set Inv. Tracks." The following output is an abbreviated display.

```
# symdev show 88
```

```
Symmetrix ID: 000000003264
```

```
Device Physical Name      : /dev/rdisk/c4t1d0s2
Device Symmetrix Name     : 0088
Device Serial ID          : 64088320
Symmetrix ID              : 000000003264

Device Group Name         : ProdAgrp
Device Logical Name       : BCV001
```

```
.....
Mirror Set Type           : [Data,Data,N/A,N/A]
Mirror Set DA Status      : [RW,RW,N/A,N/A]
Mirror Set Inv. Tracks    : [0,16470,0,0]
```

The `symmir verify -bcv_mirrors` command verifies the progress of the synchronization of the BCV's moving mirror with its fixed mirror. The ellipsis (.....) represents several lines of repetitive output that were omitted.

```
# symmir -g ProdAgrp -bcv_mirrors verify -i 30
```

None of the devices in group 'ProdAgrp' have a paired BCV device that has its mirrors in the 'Synchronized' state.

.....
All devices in group 'ProdAgrp' have a paired BCV device that has its mirrors in the 'Synchronized' state.

```
# mkdir /bcvmnt
```

The `mount` command mounts the BCV. The directory listing confirms that the file named `bigfile` is on the BCV.

```
# mount /dev/dsk/c4t1d0s2 /bcvmnt
```

```
# ls -l /bcvmnt
```

```
total 989984
-rw-r--r--  1 root      other    506601472 Mar 13 18:21 bigfile
drwx-----  2 root      root      8192 Mar 13 17:41 lost+found
```

```
# umount /bcvmnt
```

The following `symmir establish` command re-establishes the BCV with the standard. The `-reverse` option ensures that the BCV's two mirrors are synchronized, qualifying them for a successful reverse split operation later.

```
# symmir -g ProdAgrp -noprompt -reverse establish
```

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

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

The following commands replace file `bigfile` on the BCV pair with a new file named `bigfile2`.

```
# mount /dev/dsk/c4t0d0s2 /stdmnt
```

```
# ls /stdmnt
```

```
bigfile      lost+found
```

```
# rm /stdmnt/bigfile
```

```
# cp /mnt/bigfile2 /stdmnt
```

```
# ls -l /stdmnt
```

```
total 884368
-rw-r--r--  1 root      other    452551168 Mar 13 19:09 bigfile2
drwx-----  2 root      root      8192 Mar 13 17:41 lost+found
```

```
# umount /stdmnt
```

The following command initiates a reverse split operation in which original data (bigfile) is copied from the BCV's fixed mirror to its moving mirror, replacing the more recent data (bigfile2).

```
# symmir -g ProdAgrp -noprompt -reverse split
```

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

'Reverse Split' operation successfully executed for device group 'ProdAgrp'.

While the BCV mirrors are synchronizing, a `symdev show` command on the BCV displays that the moving mirror (Mirror 1) has invalid tracks because its tracks are being updated from the fixed mirror (Mirror 2).

```
# symdev show 88
```

Symmetrix ID: 000000003264

```
Device Physical Name      : /dev/rdisk/c4t1d0s2
Device Symmetrix Name     : 0088
Device Serial ID         : 64088320
Symmetrix ID             : 000000003264

Device Group Name        : ProdAgrp
Device Logical Name      : BCV001
```

```
.....
Mirror Set Type          : [Data,Data,N/A,N/A]
Mirror Set DA Status     : [RW,RW,N/A,N/A]
Mirror Set Inv. Tracks   : [13860,0,0,0]
```

While the moving mirror is being refreshed from the fixed mirror, any attempt to restore data from the BCV to the standard will fail (as the following `symmir restore` command demonstrates).

```
# symmir -g ProdAgrp restore -noprompt
```

'Incremental Restore' operation execution is in progress for device group 'ProdAgrp'. Please wait...

BCV Library error is: The operation cannot be executed because the moving mirror of the BCV is invalid

However, if `symmir restore` is issued with the `-reverse` option and the `-i 30` option, TimeFinder delays the restore operation until both BCV mirrors are synchronized. An alternative to this approach is to use the `verify -bcv_mirrors` option demonstrated earlier. The ellipsis (.....) represents several lines of repetitive output that were omitted.

```
# symmir -g ProdAgrp restore -noprompt -reverse -i 30
```

'Incremental Restore' operation execution is in progress for device group 'ProdAgrp'. Please wait...

Performing this operation at this time will not allow you to perform the next BCV split as a reverse split

'Incremental Restore' operation execution is in progress for device group 'ProdAgrp'. Please wait...

'Incremental Restore' operation successfully initiated for device group 'ProdAgrp'.

As soon as the restore operation is initiated, data on the BCV becomes accessible to the standard. The following commands mount the standard device and examine the directory. Bigfile2 is gone now, and bigfile is back. The reverse split operation copied bigfile from the BCV's fixed mirror to its moving mirror. The restore operation copied bigfile from the BCV's moving mirror to the standard.

```
# mount /dev/dsk/c4t0d0s2 /stdmnt
```

```
# ls -l /stdmnt
```

```
total 989984
```

```
-rw-r--r--  1 root      other    506601472 Mar 13 18:21 bigfile
drwx-----  2 root      root        8192 Mar 13 17:41 lost+found
```

Example 2: Using the split -not_ready Command

The following demonstration of the `split -not_ready` command uses a Windows 2000 server.

The `sympd list` command displays physical devices available on the server devices. The example uses devices 59 (\PHYSICALDRIVE6) and 6A (\PHYSICALDRIVE11). The ellipsis (.....) represents output that was truncated.

```
C:\> sympd list
```

```
Symmetrix ID: 000000003263
```

Device Name		Directors			Device		
Physical	Sym	SA	:P	DA :IT	Config	Attribute	Sts Cap (MB)
\\.\PHYSICALDRIVE1	0006	16B:1	02B:C0	Unprotected	N/Grp'd	RW	516
\\.\PHYSICALDRIVE2	0007	16B:1	01A:D3	Unprotected	N/Grp'd	RW	516
\\.\PHYSICALDRIVE3	0012	16B:1	02A:C1	BCV	N/Asst'd	RW	516
\\.\PHYSICALDRIVE4	002A	16B:1	02A:D2	RDF2	N/Grp'd	NR	516
\\.\PHYSICALDRIVE5	002B	16B:1	01B:C1	RDF2	N/Grp'd	NR	516
\\.\PHYSICALDRIVE6	0059	16B:1	02B:C1	Unprotected	N/Grp'd	RW	960
\\.\PHYSICALDRIVE7	005A	16B:1	01A:D2	Unprotected	N/Grp'd	RW	960
\\.\PHYSICALDRIVE8	005B	16B:1	01A:D1	Unprotected	Grp'd	RW	960
\\.\PHYSICALDRIVE9	0036	16B:1	02B:C3	RDF1-BCV	N/Asst'd	RW	516
\\.\PHYSICALDRIVE10	0037	16B:1	01A:D0	RDF1-BCV	N/Asst'd	RW	516
\\.\PHYSICALDRIVE11	006A	16B:1	01A:C0	2-Way BCV Mir	N/Asst'd	RW	960
\\.\PHYSICALDRIVE12	006B	16B:1	02A:C0	2-Way BCV Mir	N/Asst'd	RW	960
\\.\PHYSICALDRIVE13	006C	16B:1	01B:C0	2-Way BCV Mir	Asst'd	RW	960

The following commands create a device group and add one standard and one BCV to it.

```
C:\> symdg create ProdGrp
C:\> symld -g ProdGrp add dev 59
C:\> symbcv -g ProdGrp associate dev 6A
```

The first `symlabel list` command displays the signature of the standard; the second `symlabel list` command displays the BCV. Notice that the column for the defined device label reads N/A, while the actual label column contains a value. The Disk Administrator places that value on the physical disk when it first detects the device.

```
C:\>symlabel -g ProdGrp -type wnt list
```

```
Device Group (DG) Name: ProdGrp
DG's Type           : REGULAR
DG's Symmetrix ID    : 000000003263
```

Device Name			Device Label			Device		
Logical	Physical	Sym	Type	Defined	Actual	Sts	BCV	<=> STD
DEV001	\\.\PHYSICALDRIVE6	0059	N/A	N/A	3F7AB9E8	RW	N/A	

C:\>**symlabel -g ProdCgrp -type wnt -bcv list**

Device Group (DG) Name: ProdCgrp
 DG's Type : REGULAR
 DG's Symmetrix ID : 000000003263

Device Name			Device Label			Device	
Logical	Physical	Sym	Type	Defined	Actual	Sts	BCV <=> STD
BCV001	DRIVE11	006A	N/A	N/A	87654321	RW	NeverEstab

The `symlabel define` command defines a label for the BCV but does not specify a label explicitly. Therefore, the existing “actual” label is read off the physical device and stored as the “defined” label in the SYMAPI database. The `symlist -offline` command allows the defined label to be read from the SYMAPI database without accessing the physical device.

C:\>**symlabel -g ProdCgrp -bcv -type wnt define BCV001**

C:\>**symlabel -g ProdCgrp -bcv -type wnt -offline list**

Device Group (DG) Name: ProdCgrp
 DG's Type : REGULAR
 DG's Symmetrix ID : 000000003263

Device Name			Device Label			Device	
Logical	Physical	Sym	Type	Defined	Actual	Sts	BCV <=> STD
BCV001	DRIVE11	006A	WNT	87654321	N/A	RW	NeverEstab

The following commands initiate an establish operation followed by a `split -not_ready` operation. Wait for an establish operation to complete before attempting a split operation.

C:\>**symmir -g ProdCgrp -full -exact -noprompt establish**

'Full Establish' operation execution is in progress for device group 'ProdCgrp'.
 Please wait...

'Full Establish' operation successfully initiated for device group 'ProdCgrp'.

C:\>**symmir -g ProdCgrp -noprompt split -not_ready**

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

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

An examination of the BCV label reveals that its defined label is the original BCV label that was saved in the SYMAPI database. But its actual label matches that of the standard device (\PHYSICALDRIVE6). Although the BCV is split, the BCV status is NR (not ready).

```
C:\>symlabel -g ProdCgrp -bcv -type wnt list
```

```
Device Group (DG) Name: ProdCgrp
DG's Type           : REGULAR
DG's Symmetrix ID    : 000000003263
```

Device Name			Device Label			Device	
Logical	Physical	Sym	Type	Defined	Actual	Sts	BCV <=> STD
BCV001	DRIVE11	006A	WNT	87654321	3F7AB9E8	NR	Split

The `symlld relabel` command restores the original label to the BCV. The `-v` option provides more detail.

```
C:\>symlld -g ProdCgrp relabel -bcv -v -noprompt
```

```
Scanning device(s) for a 'RELABEL' operation (LOCAL).
```

```
Device: 006A (BCV ) [SELECTED]
```

```
STARTING a 'RELABEL' control operation.
```

```
Relabel device(s) at the local Symmetrix.....Done.
```

```
The 'RELABEL' control operation SUCCEEDED.
```

```
'Relabel' Device operation successfully completed for the device group.
```

The following listing of the BCV label confirms that relabeling was successful.

```
C:\>symlabel -g ProdCgrp -bcv -type wnt list
```

```
Device Group (DG) Name: ProdCgrp
DG's Type           : REGULAR
DG's Symmetrix ID    : 000000003263
```

Device Name			Device Label			Device	
Logical	Physical	Sym	Type	Defined	Actual	Sts	BCV <=> STD
BCV001	DRIVE11	006A	WNT	87654321	87654321	NR	Split

The BCV becomes accessible to the host by issuing the `symld ready` command.

```
C:\>symld -g ProdCgrp -bcv -v -noprompt ready
```

Scanning device(s) for a 'READY' operation (LOCAL).

Device: 006A (BCV) [SELECTED]

STARTING a 'READY' control operation.

Set device(s) Ready at local Symmetrix.....Done.

The 'READY' control operation SUCCEEDED.

'Ready' Device operation successfully completed for the device group.

An examination of the BCV reveals an RW (read/write) status, indicating that the ready action succeeded.

```
C:\>symlabel -g ProdCgrp -bcv -type wnt list
```

```
Device Group (DG) Name: ProdCgrp
DG's Type           : REGULAR
DG's Symmetrix ID    : 000000003263
```

Device Name			Device Label			Device	
Logical	Physical	Sym	Type	Defined	Actual	Sts	BCV <=> STD
BCV001	DRIVE11	006A	WNT	87654321	87654321	RW	Split

Example 3: Using a Composite Group

This example uses Solutions Enabler version 5.4. The hardware setup includes a host connected to two source Symmetrix arrays (sid 35 and 60). A composite group defined on the host contains a set of BCV pairs that spans the two Symmetrix arrays. (Beginning with Solutions Enabler version 6.0, you can control specific BCV pairs within the composite group as well as performing control operations on the entire group as shown here. For example, you can split one BCV pair without splitting the group's other pairs.)

The `symcg create` command creates a Regular type composite group named TimeFinder.

```
# symcg create TimeFinder -type regular
```

The following `symcg` commands add to the composite group a range of standard devices from each of the two source Symmetrix arrays.

```
# symcg -cg TimeFinder addall dev -range 56:6E -sid 35
# symcg -cg TimeFinder addall dev -range 14:27 -sid 60
```

The following `symbcv` commands add to the composite group a range of BCV devices from each of the two source Symmetrix arrays.

```
# symbcv -cg TimeFinder associateall dev -range 182:19A -sid 35
# symbcv -cg TimeFinder associateall dev -range 3B6:3C9 -sid 60
```

The `symmir establish` command creates optimized standard/BCV pairings among devices within each Symmetrix array and performs a full establish operation on them.

```
# symmir -cg TimeFinder establish -full -optimize -noprompt
```

```
'Full Establish' operation execution is in progress for composite group
'TimeFinder'. Please wait...
```

```
'Full Establish' operation successfully initiated for composite group
'TimeFinder'.
```

The `symmir query` command shows synchronization is in progress (SyncInProg) for the BCV pairs. Note that devices 56 and 59 are meta devices (as are their paired BCV devices), and only the meta head device is displayed.

```
# symmir -cg TimeFinder query
```

```
Composite Group Name      : TimeFinder
Composite Group Type      : REGULAR
Number of Symmetrix Units : 2
Number of RDF (RA) Groups : 2
```

```
Symmetrix ID              : 000187900035      (Microcode Version: 5670)
```

Standard Device		BCV Device		State
Sym	Inv. Tracks	Sym	Att.	Inv. Tracks
0056	0	0182	*	344150
0059	0	0185	*	228145
005B	0	0187	*	120030
005C	0	0188	*	120181

STD <=> BCV

005D	0 0189	*	120878	SyncInProgress
005E	0 018A	*	110078	SyncInProgress
005F	0 018B	*	41523	SyncInProgress
0060	0 018C	*	33900	SyncInProgress
0061	0 018D	*	123432	SyncInProgress
0062	0 018E	*	17008	SyncInProgress
0063	0 018F	*	53389	SyncInProgress
0064	0 0190	*	42266	SyncInProgress
0065	0 0191	*	7995	SyncInProgress
0066	0 0192	*	37293	SyncInProgress
0067	0 0193	*	45200	SyncInProgress
0068	0 0194	*	44386	SyncInProgress
0069	0 0195	*	45943	SyncInProgress
006A	0 0196	*	55380	SyncInProgress
006B	0 0197	*	55525	SyncInProgress
006C	0 0198	*	43544	SyncInProgress
006D	0 0199	*	48643	SyncInProgress
006E	0 019A	*	13424	SyncInProgress

Symmetrix ID : 000184500160 (Microcode Version: 5568)

Standard Device		BCV Device		State
Sym	Inv. Tracks	Sym	Att.	Inv. Tracks
0014	0 03B6	*	131367	SyncInProgress
0015	0 03B7	*	120967	SyncInProgress
0016	0 03B8	*	114601	SyncInProgress
0017	0 03B9	*	101797	SyncInProgress
0018	0 03BA	*	122833	SyncInProgress
0019	0 03BB	*	117237	SyncInProgress
001A	0 03BC	*	115669	SyncInProgress
001B	0 03BD	*	121046	SyncInProgress
001C	0 03BF	*	132493	SyncInProgress
001D	0 03BE	*	125720	SyncInProgress
001E	0 03C0	*	113095	SyncInProgress
001F	0 03C1	*	103544	SyncInProgress
0020	0 03C2	*	124720	SyncInProgress
0021	0 03C3	*	109486	SyncInProgress
0022	0 03C4	*	109383	SyncInProgress
0023	0 03C5	*	122115	SyncInProgress
0024	0 03C6	*	133037	SyncInProgress
0025	0 03C7	*	110196	SyncInProgress
0026	0 03C8	*	118622	SyncInProgress
0027	0 03C9	*	118089	SyncInProgress
Total				
Track(s)	0		4118330	
MB(s)	0		128698	

Legend:

(*): The paired BCV device is associated with this composite group.
 (p): The paired BCV device was restored using the protect option.
 (a): All mirrors of this BCV were moved to the STD.

The `symmir split` command performs an instant split on all BCV pairs in the composite group.

```
# symmir -cg TimeFinder split -instant -noprompt
```

'Split' operation execution is in progress for composite group 'TimeFinder'.
Please wait...

'Split' operation successfully executed for composite group 'TimeFinder'.

The `symmir query` command with the `-bg` and `-percent` options shows the percentage completion of the split that is currently occurring in the background.

```
# symmir -cg TimeFinder query -bg -percent
```

```
Composite Group Name      : TimeFinder
Composite Group Type      : REGULAR
Number of Symmetrix Units : 2
Number of RDF (RA) Groups : 2
```

```
Symmetrix ID              : 000187900035      (Microcode Version: 5670)
```

Standard Device		BCV Device		State		Split
Sym	Inv. Tracks	Sym	Att.	Inv. Tracks	STD <=> BCV	(%)
0056	0	0182	*	0	Split (bg)	14
0059	0	0185	*	0	Split (bg)	14
005B	0	0187	*	0	Split (bg)	15
005C	0	0188	*	0	Split (bg)	15
005D	0	0189	*	0	Split (bg)	15
005E	0	018A	*	0	Split (bg)	13
005F	0	018B	*	0	Split (bg)	15
0060	0	018C	*	0	Split (bg)	15
0061	0	018D	*	0	Split (bg)	15
0062	0	018E	*	0	Split (bg)	13
0063	0	018F	*	0	Split (bg)	15
0064	0	0190	*	0	Split (bg)	15
0065	0	0191	*	0	Split (bg)	15
0066	0	0192	*	0	Split (bg)	13
0067	0	0193	*	0	Split (bg)	15
0068	0	0194	*	0	Split (bg)	15
0069	0	0195	*	0	Split (bg)	15
006A	0	0196	*	0	Split (bg)	13
006B	0	0197	*	0	Split (bg)	15
006C	0	0198	*	0	Split (bg)	15
006D	0	0199	*	0	Split (bg)	15
006E	0	019A	*	0	Split (bg)	13

Symmetrix ID : 000184500160 (Microcode Version: 5568)

Standard Device		BCV Device		State		Split
Sym	Inv. Tracks	Sym	Att.	Inv. Tracks	STD <=> BCV	(%)
0014	0	03B6	*	0	Split (bg)	5
0015	0	03B7	*	0	Split (bg)	5
0016	0	03B8	*	0	Split (bg)	5
0017	0	03B9	*	0	Split (bg)	2
0018	0	03BA	*	0	Split (bg)	2
0019	0	03BB	*	0	Split (bg)	2
001A	0	03BC	*	0	Split (bg)	2
001B	0	03BD	*	0	Split (bg)	2
001C	0	03BF	*	0	Split (bg)	5
001D	0	03BE	*	0	Split (bg)	2
001E	0	03C0	*	0	Split (bg)	9
001F	0	03C1	*	0	Split (bg)	8
0020	0	03C2	*	0	Split (bg)	6
0021	0	03C3	*	0	Split (bg)	6
0022	0	03C4	*	0	Split (bg)	18
0023	0	03C5	*	0	Split (bg)	17
0024	0	03C6	*	0	Split (bg)	9
0025	0	03C7	*	0	Split (bg)	8
0026	0	03C8	*	0	Split (bg)	6
0027	0	03C9	*	0	Split (bg)	7
Total -----						
Track(s)	0			0		
MB(s)	0.0			0.0		

Legend:

(*): The paired BCV device is associated with this composite group.
 (p): The paired BCV device was restored using the protect option.
 (a): All mirrors of this BCV were moved to the STD.
 (bg): The paired BCV device is splitting in the background.

The `symmir verify` command checks the background split at 60-second intervals until the split operation on the composite group has completed.

```
# symmir -cg TimeFinder verify -bg -split -i 60
```

Not all devices in Composite Group 'TimeFinder' have finished splitting in the background.

Not all devices in Composite Group 'TimeFinder' have finished splitting in the background.

All devices in Composite Group 'TimeFinder' have finished splitting in the background.

Appendix A: Clone Emulation Mapping

Table 1 shows the mapping of normal TimeFinder operations to various clone operations.

Table 1. Mapping `symmir` Operations to `symclone` Operations

TimeFinder <code>symmir</code> Operation	TimeFinder <code>symclone</code> Operation
Full establish	Create with pre-copy
Split (after an establish)	Activate
Incremental establish	Recreate
Full restore	Full restore
Incremental restore	Incremental restore
Verify	Same
Attach	Same
Detach	Same
Cancel	Terminate
Protected restore	The default feature
Protected BCV establish	The default feature
Query	Same
Background split	Copy in Progress
Reverse split	Not supported

For information about clone operations, refer to the white paper *Using SYMCLI to Perform TimeFinder/Clone Operations* (P/N 300-001-382).