

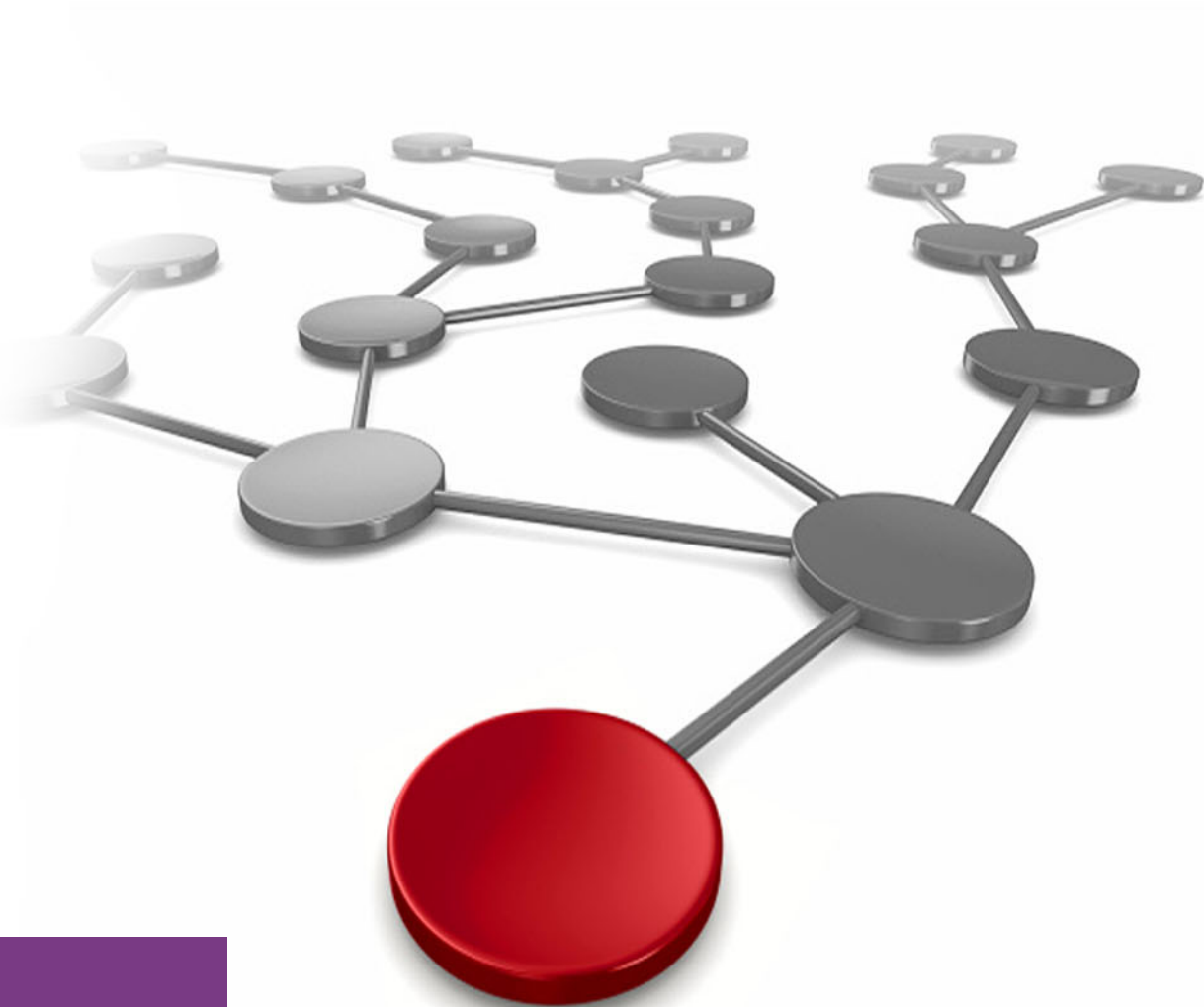
# DS8000 Cascading FlashCopy Design and Scenarios

Werner Bauer

Bert Dufrasne

Matthew Houzenga

Dennis Robertson



Storage





International Technical Support Organization

## **Cascading FlashCopy and Scenarios**

December 2017

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

**First Edition (December 2017)**

This edition applies to Version 8 Release 3 or later of the IBM DS8000 microcode.

**© Copyright International Business Machines Corporation 2017. All rights reserved.**

Note to U.S. Government Users Restricted Rights -- Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

# Contents

<b>Notices</b> .....	v
Trademarks .....	vi
<b>Preface</b> .....	vii
Authors .....	vii
Now you can become a published author, too .....	viii
Comments welcome .....	viii
Stay connected to IBM Redbooks .....	ix
<b>Chapter 1. Cascading FlashCopy concept</b> .....	1
1.1 Introduction .....	2
1.1.1 FlashCopy and FlashCopy options .....	2
1.1.2 FlashCopy prior to the Cascading capability .....	4
1.2 Cascading FlashCopy concept and design .....	4
1.2.1 Typical use cases .....	5
1.2.2 Terminology .....	5
1.3 Cascading FlashCopy and Fast Reverse Restore .....	7
1.4 Viewing Cascading FlashCopy relationships .....	7
1.4.1 Using DSCLI .....	7
1.4.2 Using the DS8880 Storage Management GUI .....	8
1.5 Thin provisioning considerations .....	8
1.5.1 Space Release .....	9
1.5.2 Withdrawing Cascading FlashCopy relationships .....	10
1.5.3 Out of Space Conditions .....	11
1.6 Design Limitations .....	11
<b>Chapter 2. Cascading FlashCopy use cases</b> .....	13
2.1 Maintain multiple FlashCopy relationships .....	14
2.1.1 Setup multiple FlashCopy relationships using DSCLI .....	15
2.1.2 Setup Multiple FlashCopy relationships using Copy Services Manager .....	16
2.2 Global Mirror configuration and multiple FC targets .....	22
2.3 Simple forward cascading FlashCopy .....	23
2.3.1 Update FlashCopy target volume .....	24
2.3.2 Reverse FlashCopy .....	27
2.4 Cascading data set level FlashCopy .....	30
2.5 Remote Pair FlashCopy .....	36
2.6 Multiple FC relationships in a more complex configuration .....	37
<b>Related publications</b> .....	41
IBM Redbooks .....	41
Other publications .....	41
Online resources .....	41
Help from IBM .....	41



# Notices

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

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

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

*IBM Director of Licensing, IBM Corporation, North Castle Drive, MD-NC119, Armonk, NY 10504-1785, US*

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

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

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

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

The performance data and client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

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

Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

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


## COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

# Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at “Copyright and trademark information” at <http://www.ibm.com/legal/copytrade.shtml>

The following terms are trademarks or registered trademarks of International Business Machines Corporation, and might also be trademarks or registered trademarks in other countries.

DB2®	Redbooks®	System Storage®
DS8000®	Redpaper™	z/OS®
FlashCopy®	Redbooks (logo)  ®	
IBM®	S/390®	

The following terms are trademarks of other companies:

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

# Preface

This IBM® Redpaper™ publication describes the IBM Cascading FlashCopy® support introduced with the IBM DS8000® License Machine Code (LMC) Release 8.3 (LMC 8.8.30.xx.xx, bundle version 88.30.xxx.xx) or later.

The Cascading FlashCopy support enables a FlashCopy source device to become a FlashCopy source, or a FlashCopy target device to become a FlashCopy source.

This capability provides improvements to a range of Copy Services scenarios, but essentially enables the reversal of one of several FlashCopy relationships without first withdrawing the other relationships.

This publication explores a range of use cases, explaining the improvements and benefits of the Cascading FlashCopy.

## Authors

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

**Werner Bauer** is a Certified Consulting IT Specialist in Germany. He has more than 30 years of experience in storage software and hardware, and with IBM S/390® and IBM z/OS®. His areas of expertise include DR solutions based on IBM enterprise disk storage systems. He has also written extensively in various IBM Redbooks® publications about the DS8000. He holds a degree in Economics from the University of Heidelberg, and in Mechanical Engineering from FH Heilbronn. He currently works with System Vertrieb Alexander (SVA), an IBM Premier Business Partner.

**Bert Dufrasne** is an IBM Certified IT Specialist and Project Leader for IBM System Storage® disk products at the ITSO, San Jose Center. He has worked at IBM in various IT areas. He has written many IBM publications, and has developed and taught technical workshops. Before Bert joined the ITSO, he worked for IBM Global Services as an Application Architect. He holds a Master's degree in Electrical Engineering.

**Matthew Houzenga** is an Executive IT Specialist and Client Technical Specialist for IBM Storage in Cleveland, OH. He joined IBM Storage in 2000 and worked for the support and development organizations in San Jose and Tucson before moving to the field in 2006. He holds a degree in Computer Science from Northern Illinois University.

**Dennis Robertson** is a DS8000 Subject Matter Expert with the Washington Systems Center. He has over 30 years of experience with roles as Program Support Representative, Large System Engineer, Storage Field Technical Support Specialist, and Systems Architect. Dennis is an IBM Storage Systems Certified specialist and has been a member of the IBM Certification Review Board. He graduated from the DeVry Institute of Technology.

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

Nick Clayton  
Lisa Gundy  
Paul Spagnolo  
Warren Stanley  
Alexander Warmuth  
Alex Westphal  
**IBM**

## Now you can become a published author, too

Here's an opportunity to spotlight your skills, grow your career, and become a published author—all at the same time. Join an ITSO residency project and help write a book in your area of expertise, while honing your experience using leading-edge technologies. Your efforts will help to increase product acceptance and customer satisfaction, as you expand your network of technical contacts and relationships. Residencies run 2 - 6 weeks in length, and you can participate either in person or as a remote resident working from your home base.

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

[ibm.com/redbooks/residencies.html](http://ibm.com/redbooks/residencies.html)

## Comments welcome

Your comments are important to us.

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

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

[ibm.com/redbooks](http://ibm.com/redbooks)

- Send your comments in an email to:

[redbooks@us.ibm.com](mailto:redbooks@us.ibm.com)

- Mail your comments to:

IBM Corporation, International Technical Support Organization  
Dept. HYTD Mail Station P099  
2455 South Road  
Poughkeepsie, NY 12601-5400

## Stay connected to IBM Redbooks

- ▶ Find us on Facebook:  
<http://www.facebook.com/IBMRedbooks>
- ▶ Follow us on Twitter:  
<http://twitter.com/ibmredbooks>
- ▶ Look for us on LinkedIn:  
<http://www.linkedin.com/groups?home=&gid=2130806>
- ▶ Explore new Redbooks publications, residencies, and workshops with the IBM Redbooks weekly newsletter:  
<https://www.redbooks.ibm.com/Redbooks.nsf/subscribe?OpenForm>
- ▶ Stay current on recent Redbooks publications with RSS Feeds:  
<http://www.redbooks.ibm.com/rss.html>





# Cascading FlashCopy concept

This chapter introduces the concepts and terminology related to the IBM DS8000 Cascading FlashCopy.

This chapter includes the following topics:

- ▶ Introduction
- ▶ FlashCopy function prior to the Cascading feature
- ▶ Cascading FlashCopy
- ▶ FlashCopy Options

## 1.1 Introduction

In today's modern, complex application environments, the need to have multiple copies of data for the purpose of backup, testing, development, data mining, and disaster recovery is essential. From that standpoint, The DS8000 series offers an extensive and comprehensive set of Copy Services that helps fulfill those business needs.

### 1.1.1 FlashCopy and FlashCopy options

Before diving into the specifics of the Cascading FlashCopy function, we review the basics of the FlashCopy function and its different options. Understanding those options is critical for a proper usage of the Cascading FlashCopy capability. As one of the basic DS8000 Copy Services, FlashCopy enables the creation of a point in time copy of a volume, or a data set (a subset of a volume in a z/OS environment).

When describing FlashCopy, the following terms are used:

- ▶ *Source* refers to the original data that is to be copied.
- ▶ *Target* refers to the destination where the data is to be copied.
- ▶ *FlashCopy relationship* refers to the relationship being created between the specified source volume or tracks and the specified target volume or tracks
- ▶ *Point-in-time copy* describes the result of a FlashCopy establish operation. The target of the establish contains a copy of the source as of the point in time that the establish was performed.
- ▶ *Time-zero (T0) copy* refers to the first point in time that a copy is taken. A subsequent version would be T1, then T2, and so on.
- ▶ *Background copy* refers to the physical copy operation that is performed by the DS8000 in the background, asynchronously after a FlashCopy has been established.

With FlashCopy, when the FlashCopy command completes, both the source and target copies are immediately available for read and write operations. FlashCopy is also known as a point-in-time copy, fast replication, or time-zero copy (T0 copy).

The following list provides a brief summary of each of the FlashCopy options:

**Full volume FlashCopy** A full volume FlashCopy can be accomplished by specifying only a source volume and a target volume, without any indication of a subset (range of tracks in z/OS) specified. This option tells the DS8000 to establish a relationship between the source volume and the target volume. As long as the target volume is the same size or larger than the source and the two volumes are compatible, the relationship will be established.

**Copy option** When the copy option is invoked and the establish process completes, a background process is started that copies all data from the source to the target. When the background copy is complete, if not explicitly established as a persistent relationship (or incremental), the FlashCopy relationship ends as soon as all of the data is copied.

**NoCopy option** With this option, the background copy task is not automatically started when the relationship is established. The only time data is transferred from the source to the target is when an update occurs to the T0 version of the data on the source.

That T0 version will be copied to the target before it is “hardened” on the source. The relationship lasts until it is explicitly withdrawn, or until every block or track of data on the source volume is modified.

**Persistent FlashCopy** With this option, the FlashCopy relationship continues until explicitly removed (until the user terminates the relationship by using one of the interface methods). If this option is not selected, the FlashCopy relationship exists only until all of the data is copied from the source volume to the target.

**Incremental FlashCopy** Incremental FlashCopy provides the capability to create a persistent FlashCopy relationship between two volumes with a change recording mechanism. The change recording capability allows you to *refresh* the target of the FlashCopy relationship by issuing another FlashCopy **Establish** command with the appropriate keywords. When a new increment is created, only the tracks or blocks that changed since the last increment are copied to the target (instead of all of the volume’s tracks or blocks), and those tracks or blocks on the target that changed are re-copied from the source to put the two volumes back in sync.

**Reversing FlashCopy** To reverse a FlashCopy relationship, you must wait for the background copy to complete, and then the relationship can be reversed. Tracks or blocks that were changed on what was the target (now the source after the reversal) must be copied, and tracks or blocks that have changed on what was the source (now the target after the reversal) must be restored by the new source’s tracks or blocks to put the two volumes back into sync.

**Fast reverse restore** Fast reverse restore performs the same operation as a FlashCopy reversal except it bypasses the check to ensure that the background copy completed before the reversal can happen. This means that the source volume (original target) cannot be relied upon for valid data. This option is used during Global Mirror recovery processes.

#### **Remote (in-band) FlashCopy**

Remote FlashCopy provides the ability to manage FlashCopy for Peer-to-Peer Remote Copy (PPRC) secondary devices at their remote site using their associated PPRC primary devices at the production site as an access device. The commands can be issued from the local site to a Metro Mirror or Global Copy primary and they are then transmitted over the Metro Mirror or Global Copy links to the secondary.

#### **FlashCopy for z/OS data sets**

Data set FlashCopy is only supported by z/OS and allows the copying of datasets from the same or different volumes, as long as the target is large enough to contain the source track range.

#### **FlashCopy (FlashCopy consistency group)**

FlashCopy consistency groups can be used to help create a consistent point-in-time copy across multiple volumes, and even across multiple DS8000 storage systems, maintaining the order of dependent writes.

### 1.1.2 FlashCopy prior to the Cascading capability

Before the Cascading FlashCopy capability, you were allowed to have one FlashCopy source and up to 12 FlashCopy target volume or dataset relationships. This allowed you to have multiple PiT(T0) relationships and therefore multiple copies of the source volume or dataset, created at different times against the same source., T1, T2...T11.

However, to restore one of the targets back to the source, you had to withdraw the other relationships that shared the same source, as shown in Figure 1-1.

Users could make multiple copies and then restore only one of these copies. However, they had to remove any existing relationships from the original source prior to reversing one relationship. If the wrong PiT is chosen, and the background copy has not completed, other PiT backups are no longer available.

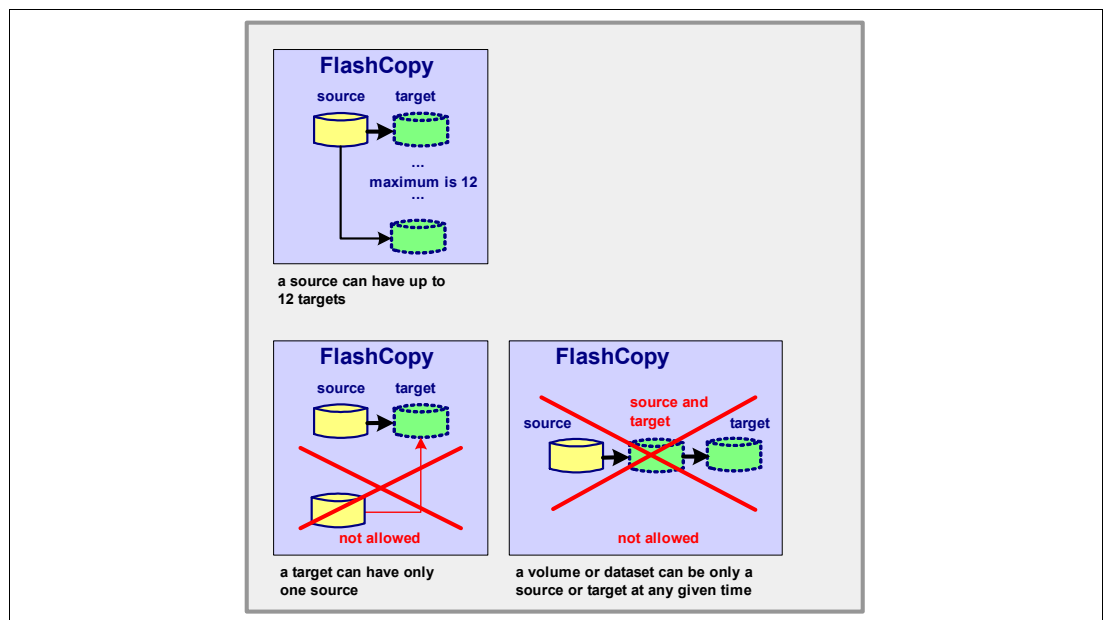


Figure 1-1 FlashCopy relationships prior to cascading.

Any volume can be the source of up to 12 FlashCopy relationships.

## 1.2 Cascading FlashCopy concept and design

Before DS8880 Release 8.3, a volume could not be both the source and a target in a FlashCopy relationship at the same time. The same restriction applied to data sets in the case of data set-level FlashCopy.

Release 8.3 lifts this restriction and allows a volume or track to be both a source in one FlashCopy relationship and target in a second FlashCopy relationship. This is referred to as a *Cascading FlashCopy* relationship, as illustrated in Figure 1-2 on page 5.

There is no architectural limit to the number of FlashCopy cascades other than the total number of volumes within a DS8880. However, the maximum number of FlashCopy relationships is still limited to 12.

A volume or dataset can perform *one* of the following functions:

- ▶ Source in up to 12 FlashCopy relationships OR
- ▶ Target in 1 FlashCopy relationship AND a source in up to 11 FlashCopy relationships

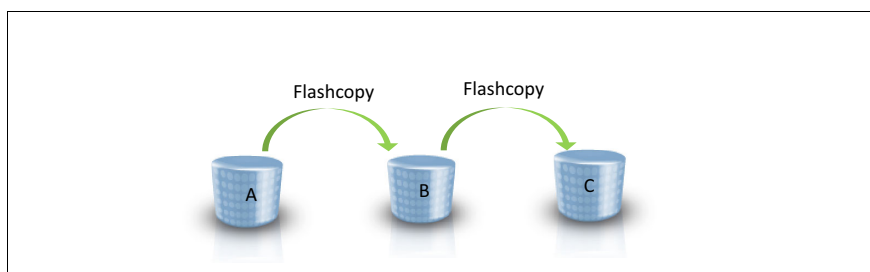


Figure 1-2 Cascading FlashCopy concept

## 1.2.1 Typical use cases

Typical use cases and applications for cascading FlashCopy include:

- ▶ Reversing one of several FlashCopy relationships from a source device to restore this copy without first removing the other relationships
- ▶ Recovering a Global Mirrored environment without needing to withdraw an existing FlashCopy used for Disaster Recovery testing
- ▶ Using a dataset FlashCopy between devices that are both the sources of *other* FlashCopy relationships including in Remote Pair FlashCopy environments
- ▶ Performing an object-level restore using FlashCopy from an IBM DB2® System Backup that still has an active FlashCopy relationship
- ▶ Increasing the flexibility of dataset FlashCopy where an existing source data set can become a target of a new FlashCopy and an existing track can become a source of a new FlashCopy.
- ▶ Recovering from Ransomware or other malicious event.

Also see Chapter 2, “Cascading FlashCopy use cases” on page 13 for a step-by-step walkthrough of practical use cases.

## 1.2.2 Terminology

Several new terms are being introduced to distinguish between different types of Cascading FlashCopy topologies.

### Forward cascade

A forward cascading relationship is created when the target volume in an existing FlashCopy relationship becomes the source volume in a new FlashCopy relationship.

In Figure 1-3, a FlashCopy relationship was established at Time 0 from Volume A to Volume B. A forward cascading FlashCopy relationship is then created at Time 1, when a FlashCopy is established from Volume B to Volume C.

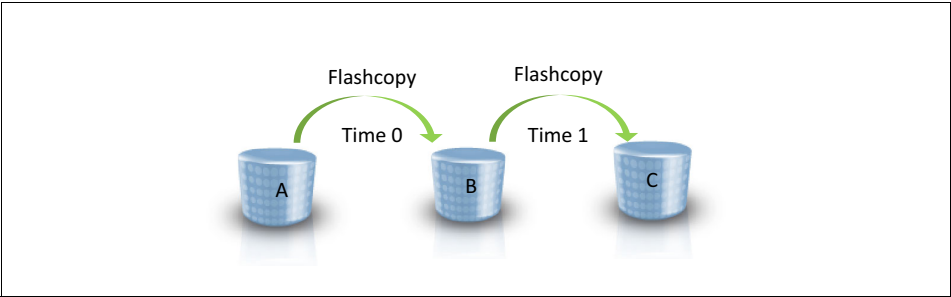


Figure 1-3 Forward Cascading FlashCopy Relationship

### Backward Cascade

A backward cascading relationship is created when the source volume in an existing FlashCopy relationship becomes the target volume in a new FlashCopy relationship.

In Figure 1-4, a FlashCopy relationship was established at Time 0 from Volume B to Volume C. A backward cascading FlashCopy relationship is then created at Time 1 when FlashCopy is established from Volume A to Volume B.

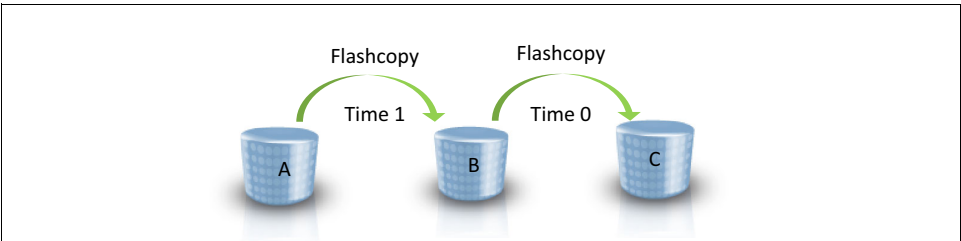


Figure 1-4 Backward Cascading FlashCopy Relationship

### Source Relation and Target Relation

In a Cascading FlashCopy relationship with FlashCopy established between volume A and volume B, and a second FlashCopy relationship between volume B and volume C (see Figure 1-5), the following terms are used to distinguish between the two FlashCopy relationships:

- ▶ Source Relation: the FlashCopy relationship between volume A and volume B
- ▶ Target Relation: the FlashCopy relationship between volume B and volume C

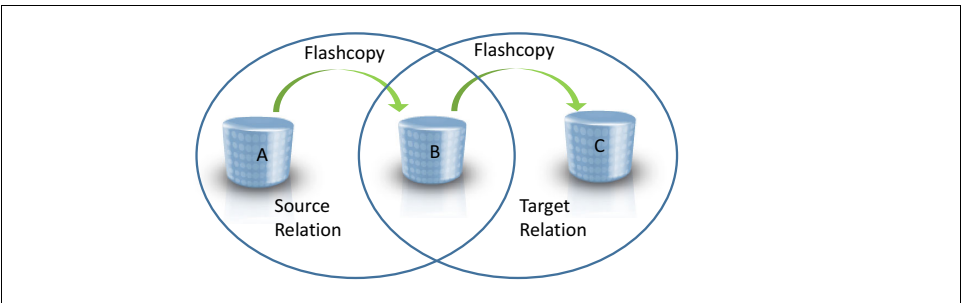


Figure 1-5 Source and Target FlashCopy relations

## 1.3 Cascading FlashCopy and Fast Reverse Restore

Because Cascading FlashCopy allows a volume to be simultaneously the source of one relation and the target of another relation, it is now possible to reverse one of the relationships without first withdrawing the other relationship. As illustrated in Figure 1-6 the **A → B** FlashCopy relationship can now be reversed while the **A → C** FlashCopy relationship is preserved.

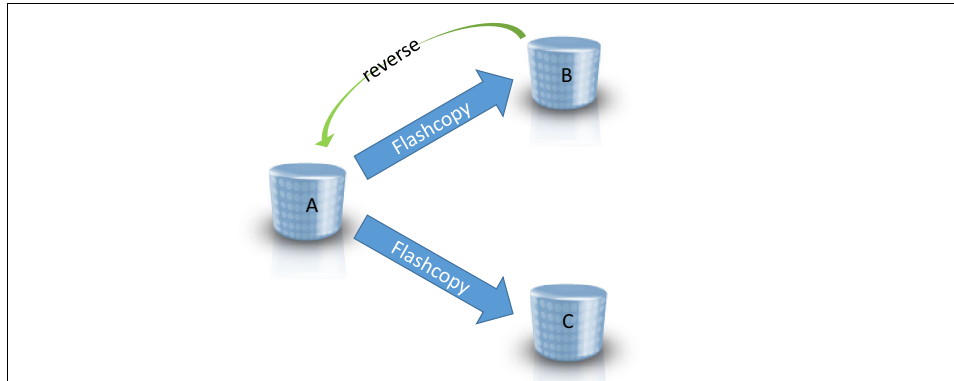


Figure 1-6 Cascading FlashCopy and Fast Reverse Restore

## 1.4 Viewing Cascading FlashCopy relationships

Although there are no special indicators added for Cascading FlashCopy relationships, the DSCLI and DS8000 Storage Management GUI can be used to determine whether a volume is in a cascading relationship. Additionally there are z/OS tools available to display Flashcopy relationships, but they are out of scope for this red paper.

### 1.4.1 Using DSCLI

The DSCLI **lsflash** command will display all the FlashCopy relationships that exist for a volume. If the volume is both a source volume and a target volume, thus in cascading relationships, both relationships will be displayed as illustrated in Example 1-1 for Volume\_ID 0021.

*Example 1-1 Using lsflash to check for cascading relationships*

```
dscli> lsflash 0021
```

```
Date/Time: August 29, 2017 5:09:24 PM CEST IBM DSCLI Version: 7.8.30.459 DS:
IBM.2107-75HGX91
```

```
ID          SrcLSS SequenceNum Timeout ActiveCopy Recording Persistent Revertible
SourceWriteEnabled TargetWriteEnabled BackgroundCopy
```

```
=====
```

```
0021:0011 00      0          60      Disabled Disabled Enabled Disabled
Enabled      Enabled
```

```
0221:0021 01      0          60      Disabled MultInc Enabled Disabled
Enabled      Enabled
```

## 1.4.2 Using the DS8880 Storage Management GUI

The DS8880 GUI has been updated in Release 8.3 to display FlashCopy relationships. To display all of the FlashCopy relationships on the DS8880, hover over the **Copy Services** icon and select **FlashCopy**, as shown in Figure 1-7.



Figure 1-7 DS8880 Storage Management GUI: Copy Services

When a Cascading FlashCopy relationship is present, a volume will be reported as both a source volume and a target volume, as shown in Figure 1-8 for the ITSO\_SLES12\_010C volume.

Actions Filter			
Date created	Source volume	Target volume	State
IBM.2107-75HGX91 (2)			
08/31/2017 07:33:50 AM	ITSO_SLES12_010C	ITSO_SLES12_010D	✓ Established
08/30/2017 05:37:59 AM	ITSO_SLES12_010B	ITSO_SLES12_010C	✓ Established

Figure 1-8 DS8880 Storage Management GUI: Copy Services - FlashCopy

## 1.5 Thin provisioning considerations

It is important to consider the implications of Cascading FlashCopy relationships when using extent space efficient (ESE) volumes on the DS8880. Withdrawing the source relation in a cascading relationship created with the NOCOPY option results in a background copy being started to the target volume in the source relation. This situation will cause the allocated physical capacity to increase for the target volume.

The behavior of withdrawing Flashcopy relationships established with COPY can also be affected by Cascading Flashcopy. When a Flashcopy target volume is in a cascading relationship, the Flashcopy is not withdrawn until the background copy completes.

Using the DSCLI, the allocated capacity for a volume can be displayed under the `realcap` output value of the `showckdvol` or `showfbvol` command. Refer to

Using the DS8880 Storage Management GUI, the allocated capacity can be viewed by hovering over the **Volumes** icon and clicking **Volumes**, as shown in Figure 1-9 on page 9.



Figure 1-9 DS8880 Storage Management GUI: Volumes

Make sure to specify the **Allocated Capacity** field by clicking the icon at the top-right corner and selecting **Allocated Capacity**, as shown in Figure 1-10.

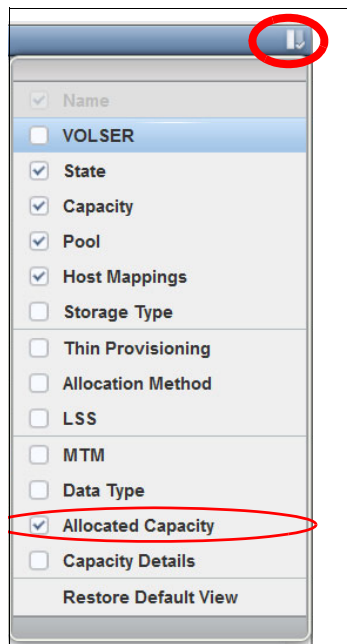


Figure 1-10 DS8880 Storage Management GUI: Volumes->Allocated Capacity

## 1.5.1 Space Release

For full volume FlashCopy relationships, space release will occur on an ESE target volume as long as the following conditions are met:

- ▶ The target volume is not cascading in a cascading FlashCopy relationship
- ▶ The target volume is not in any other copy services relationships.

Note that establishing data set level FlashCopy onto an ESE volume will not cause space to be released on the target volume.

## 1.5.2 Withdrawing Cascading FlashCopy relationships

In a non-cascading FlashCopy relationship, withdrawing FlashCopy does not affect the allocated capacity of a target volume.

In a cascading FlashCopy relationship **A → B → C**, withdrawing the **A → B** FlashCopy relationship while the **B → C** FlashCopy relationship still exists will result in all the allocated capacity on volume A being physically copied to volume B.

If volume A is a fully allocated volume, this will cause volume B to allocate its full capacity.

### Example

In Example 1-2, a cascading FlashCopy relationship is created from volume 0111 to volume 0110 to volume 0112. All three volumes are 1 GiB ESE volumes and volume 0111 has 176 MiB of real capacity allocated.

Before the Flashcopies are established, volume 0110 has 0 MiB of real capacity allocated.

When the FlashCopy relationship between volume 0111 and volume 0110 is removed, message CMUN81131I is issued to alert the user that a physical copy will be completed before the FlashCopy relationship is withdrawn.

*Example 1-2 Withdrawing a Cascading FlashCopy relationship*

---

```
dscli> showfbvol 0111
Name                ITS0_SLES12_
ID                  0111
cap (MiB)           1024
realcap (MiB)       176

dscli> showfbvol 0110
Name                ITS0_SLES12_
ID                  0110
cap (MiB)           1024
realcap (MiB)       0

dscli> mkflash -nocp -tgtse -persist 0111:0110
CMUC00137I mkflash: FlashCopy pair 0111:0110 successfully created.

dscli> mkflash -nocp -tgtse -persist 0110:0112
CMUC00137I mkflash: FlashCopy pair 0110:0112 successfully created.

dscli> rmflash 111:110
CMUC00144W rmflash: Are you sure you want to remove the FlashCopy pair 111:110:?
[y/n]:
y
CMUN81131I rmflash: 0111:0110: The Withdraw command was accepted and the FLC
relationship will be removed after the physical copy is completed.
```

---

A subsequent **showfbvol** command against volume 0110 shows that now the allocated capacity for volume 0110 matches that of volume 0111 after the FlashCopy is removed and the physical copy has completed. See Example 1-3.

*Example 1-3 Capacity allocated on Target*

---

```
dsccli> showfbvol 0110
Date/Time: August 28, 2017 9:56:06 AM CEST IBM DSCLI Version: 7.8.30.459 DS:
IBM.2107-75HGX91
```

Name	ITS0_SLES12_
ID	0110
cap (MiB)	1024
<b>realcap (MiB)</b>	<b>176</b>

---

### 1.5.3 Out of Space Conditions

When using Extent Space Efficient (ESE) FlashCopy target volumes, it is possible for the target extent pool to run out of space. This will cause the FlashCopy relationship to fail. The writes to the FlashCopy source volume will be allowed if there is no Cascading FlashCopy relationship. If there is a forward cascading FlashCopy relationship from the target volume that ran out of space, this FlashCopy relationship (target relation) will also fail. Note that Global Mirror FlashCopy targets that run out of space cause the source volume to inhibit writes.

## 1.6 Design Limitations

Note that the following restrictions apply to cascading FlashCopy relationships:

- ▶ A volume can only be a target volume in one FlashCopy relationship.
- ▶ A volume or dataset can be involved in a maximum of 12 relationships
  - 12 source relationships OR
  - 11 source relationships and 1 target relationship
- ▶ Cannot cascade from an incremental FlashCopy relationship until the background copy has completed
- ▶ Cannot create cyclic FlashCopy relationships between two volumes where there is a relationship from **Volume A** → **Volume B** and **Volume B** → **Volume A**.





## Cascading FlashCopy use cases

Although Cascaded FlashCopy allows for multiple FlashCopy configuration and permutations, this chapter focuses on several popular configurations.

The use cases in this chapter are grouped into the following categories:

- ▶ Focus on reversing a FlashCopy relationship without impacting other existing FlashCopy relationships off a common FlashCopy source volume:
  - Two or more FlashCopy relationships exist off a common source volume. With cascading FlashCopy support, it is now possible to reverse any one of these existing FlashCopy relationships without removing the other FlashCopy relationships.
  - Global Mirror with one, or even more, additional test volumes at the recovery site is another variation of the previous use case. This use case illustrates a Global Mirror failover process without impacting the existing FlashCopy relations between the Global Copy target and one or more FlashCopy target volumes.
- ▶ Focus on cascaded topologies:
  - Because cascading FlashCopy has no limit on the number of cascaded relationships, this example shows a cascaded FlashCopy configuration with just three cascaded volumes involved. This configuration is established in a forward cascaded manner.
  - Another use case applies to CKD volume environments and data set level FlashCopy. This use case dives a bit deeper into CKD volume considerations with FlashCopy.
  - The next use case builds on Remote Pair FlashCopy (preserve mirror), where FlashCopy commands are propagated from Metro Mirror primary volumes to corresponding Metro Mirror secondary volumes while keeping the existing Full Duplex Metro Mirror status of the Metro Mirror copy pairs. In this use case, we assume an existing FlashCopy relationship off the Metro Mirror secondary volumes, because one would have to create backups, for example. The resulting topology is also a cascading FlashCopy relationship.
  - The last use case explores configurations that are a bit more complex, expanding a Metro Mirror configuration and exploiting the cascading FlashCopy capability to allow new ways of managing backups or doing data analysis.

## 2.1 Maintain multiple FlashCopy relationships

A common approach to quickly create a set of volumes as a potential application restart point, is to utilize full volume FlashCopy.

A scenario might be to create such a set of volumes before starting a nightly batch activity. At another specified point in time in the batch window, another set of FlashCopy target volumes is established to ensure a potential restart point without the need to start from the beginning. When the batch processing completes, another FlashCopy creates yet another set of FlashCopy target volumes that contain the results of the entire batch window that might subsequently be used for backup to tape.

If there is a need to repeat part of the batch processing but not from the beginning, it is desirable to keep the copy from when the batch window was initially started in case the restart off the intermediate copy fails, as illustrated in Figure 2-1.

This is probably the most awaited FlashCopy function and a common use case in DB2 Backup System implementations in z/OS environments. It is now possible with cascading FlashCopy capability.

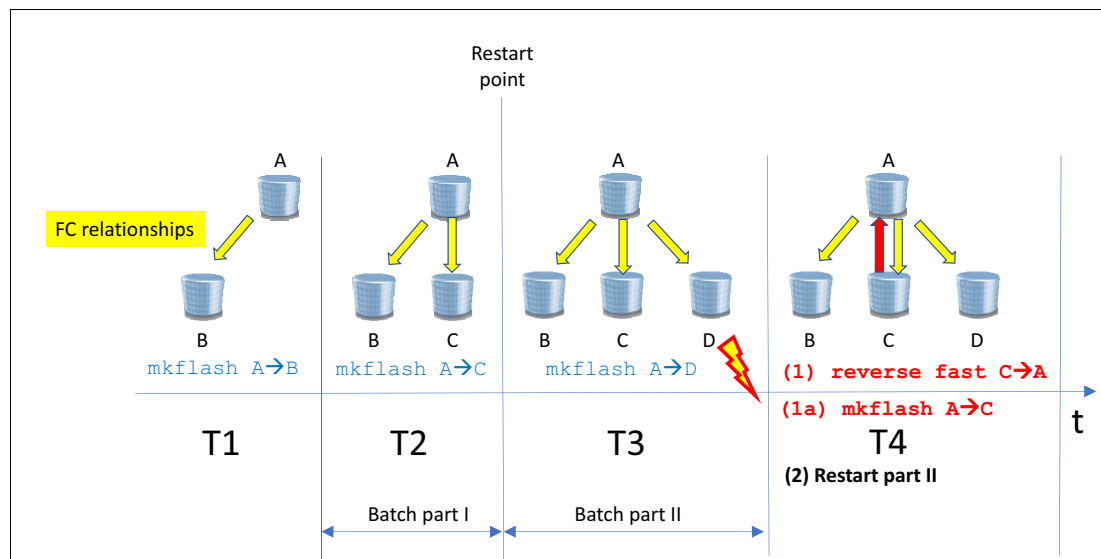


Figure 2-1 Maintain existing FlashCopy relationships

Consider the T4 time frame: After the Fast Reverse Restore (FRR) from C to A completes, the FlashCopy relationship A:C ends and volume C does not contain any useful data any longer. After the FRR completes, you can optionally re-establish the original FlashCopy relationship A:C (1a) to keep this data copy on volume C and preserve it for the case when another restart at this data point is needed. Alternatively, you can re-initialize the volume to make it usable again for any other purpose.

If the restart for batch part II in T4 fails, it will be necessary to repeat also batch part I from T2. In this case a fast reverse restore of the existing FlashCopy relationship between volume A and B in T2 provides the original copy of the data before restarting batch part I from T2 and still keep the data level from T1 and T3.

## 2.1.1 Setup multiple FlashCopy relationships using DSCLI

Example 2-1 shows a DSCLI command sequence that creates three multiple FlashCopy target volumes over a batch processing period from T1 to T4 in reference to Figure 2-1.

After listing through an **lsflash** command these new FlashCopy relationships off a common source volume with the address of **D700**, the second FlashCopy relationship **D700:D910** is reversed from **D700:D910** to **D910:D700**. After copying the changed tracks from **D910** back to **D700**, the FlashCopy relationship between these two devices is withdrawn. The other two existing FC relationships stay unchanged.

*Example 2-1 Create multiple FlashCopy target relations and reverse a relationship of a FlashCopy pair*

```
dscli> lsflash -l d700-d7ff
Date/Time: 30. August 2017 10:54:16 CEST IBM DSCLI Version: 7.8.30.435 DS:
IBM.2107-75xxx1
CMUC00234I lsflash: No Flash Copy found.

dscli> mkflash -nocp -persist D700:D900
Date/Time: 30. August 2017 10:54:29 CEST IBM DSCLI Version: 7.8.30.435 DS:
IBM.2107-75xxx1
CMUC00137I mkflash: FlashCopy pair D700:D900 successfully created.
dscli> mkflash -nocp -persist D700:D910
Date/Time: 30. August 2017 10:54:36 CEST IBM DSCLI Version: 7.8.30.435 DS:
IBM.2107-75xxx1
CMUC00137I mkflash: FlashCopy pair D700:D910 successfully created.
dscli> mkflash -nocp -persist D700:D930
Date/Time: 30. August 2017 10:54:42 CEST IBM DSCLI Version: 7.8.30.435 DS:
IBM.2107-75xxx1
CMUC00137I mkflash: FlashCopy pair D700:D930 successfully created.

dscli> lsflash d700-d7ff
Date/Time: 30. August 2017 10:54:59 CEST IBM DSCLI Version: 7.8.30.435 DS:
IBM.2107-75xxx1
ID          SrcLSS SequenceNum Timeout ActiveCopy Recording Persistent Revertible
SourceWriteEnabled TargetWriteEnabled BackgroundCopy
=====
=====
D700:D900 D7      0          120    Disabled Disabled Enabled Disabled
Enabled                      Enabled Disabled
D700:D910 D7      0          120    Disabled Disabled Enabled Disabled
Enabled                      Enabled Disabled
D700:D930 D7      0          120    Disabled Disabled Enabled Disabled
Enabled                      Enabled Disabled

dscli> reverseflash -fast D700:D910
Date/Time: 30. August 2017 14:21:00 CEST IBM DSCLI Version: 7.8.30.435 DS:
IBM.2107-75xxx1
CMUC00169I reverseflash: FlashCopy volume pair D700:D910 successfully reversed.

dscli> lsflash d700-d7ff
Date/Time: 30. August 2017 14:21:10 CEST IBM DSCLI Version: 7.8.30.435 DS:
IBM.2107-75xxx1
ID          SrcLSS SequenceNum Timeout ActiveCopy Recording Persistent Revertible
SourceWriteEnabled TargetWriteEnabled BackgroundCopy
```

```

=====
=====
D700:D900 D7      0      120    Disabled Disabled Enabled Disabled
Enabled      Enabled
D700:D930 D7      0      120    Disabled Disabled Enabled Disabled
Enabled      Enabled      Disabled
dscli>

```

When the last DSCLI **1sf1ash** command in the command sequence in Example 2-1 on page 15 lists all FlashCopy relationships, the fast reverse restore completed already from **D910** to **D700** and ended this FlashCopy relationship. Therefore, the corresponding FlashCopy relationship **D700:D910** does not appear in the FlashCopy listing anymore.

### 2.1.2 Setup Multiple FlashCopy relationships using Copy Services Manager

This section shows how to use the Copy Services Manager (CSM) GUI to establish three FlashCopy relationships off the same source volume. It also shows how to reverse the second FlashCopy relationship without affecting the existing first and third FlashCopy relationships. The CSM GUI guides the user through the process with an easy step-by-step fashion. Using CSM rather than DS CLI protects the user from potential procedural errors.

#### Creating FlashCopy sessions

Figure 2-2 shows the Copy Services Manager (CSM) Sessions view with the corresponding FC sessions. Each session addresses a source and a target FlashCopy volume, and in each of these three sessions the source volume is the same volume. Only the FlashCopy target volume is different in each relation.

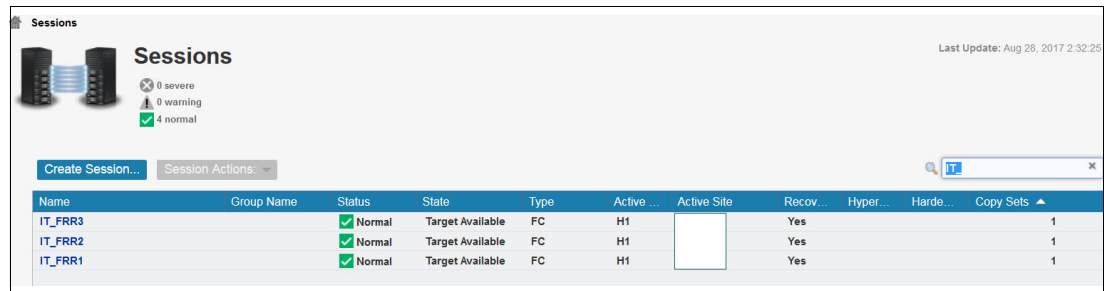


Figure 2-2 Target FlashCopy configuration

When creating the second and third CSM FlashCopy session, a warning message is displayed to point out that the source volume is already defined in another session. This is just a warning and can be ignored in this case.

Figure 2-3 on page 17 shows a CSM console fragment when creating the third FlashCopy session **IT\_FRR3**. The text explains that the same source volume is already defined in session **IT\_FRR1** and in session **IT\_FRR2**. In general, the CSM console log is a good source to understand what CSM did, and can be saved to a document and to trace corresponding activities.

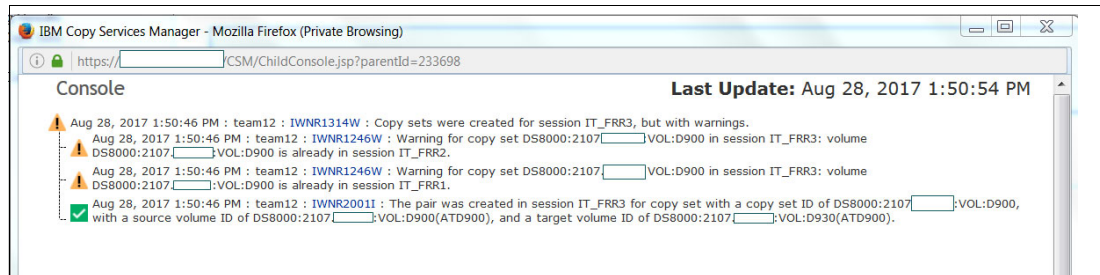


Figure 2-3 Copy Services Manager console log explains the warning message

In the course of creating a FlashCopy session, modify the properties of the CSM FlashCopy session to make the session a no background copy session and persistent:

1. Figure 2-4 shows the panel that leads to the Properties section. Note that this is not the **Commands** option but the **View/Modify** option that has to be selected.

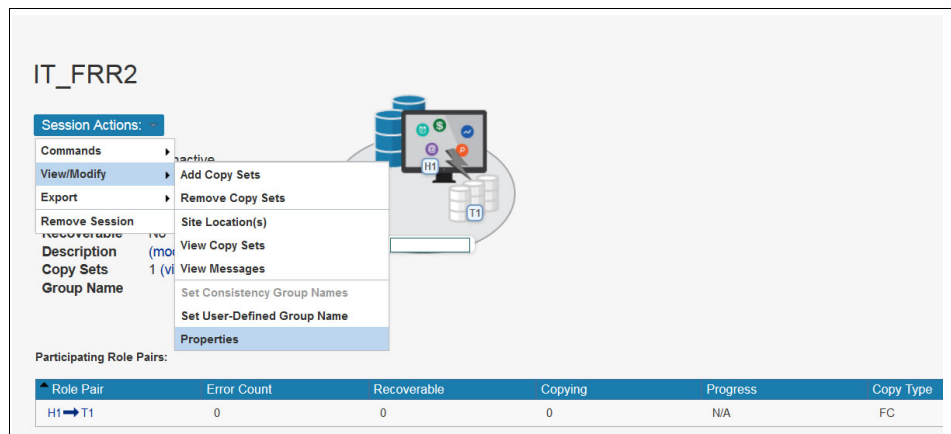


Figure 2-4 Modify the FlashCopy session through the Properties tag

2. Figure 2-5 shows the **Properties** window. Select the H1-T1 Options tab and select **Persistent** and **No Copy**.

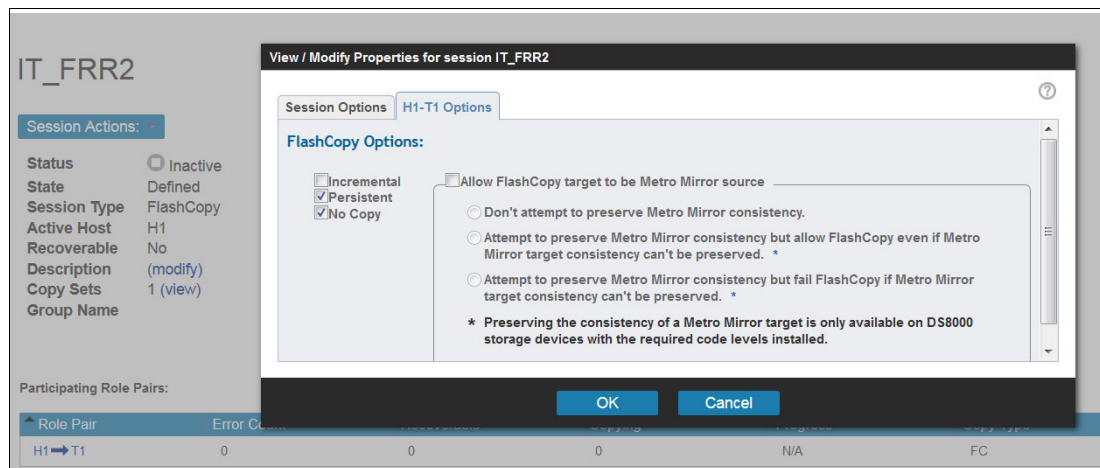


Figure 2-5 Make the session persistent and no background copy

3. Select **OK** to confirm, which makes the session is ready to run and to create an actual FlashCopy relationship. Figure 2-6 shows the actual **Command** → **Flash** action to start the FlashCopy operation.

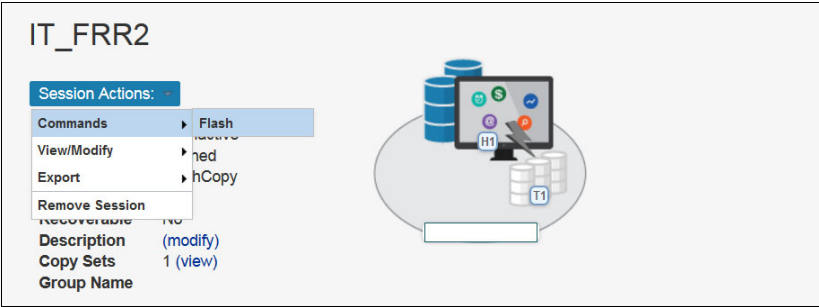


Figure 2-6 Submit FlashCopy - here through the CSM session IT\_FRR2

As shown in Figure 2-7, CSM asks for confirmation to create the FlashCopy relationship, which is normal CSM behavior for running any session

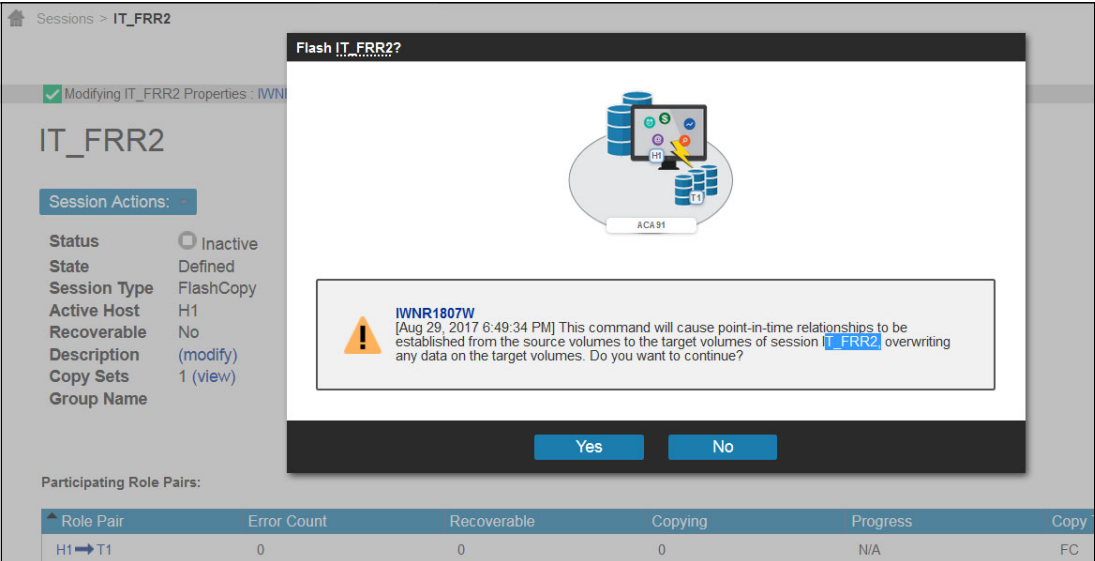


Figure 2-7 Acknowledge to execute a FlashCopy operation

Figure 2-8 shows the result after all three FlashCopy sessions became active. They are all in a healthy state *Normal*. For the sake of simplicity, each session here contains just one FlashCopy Copy Set.

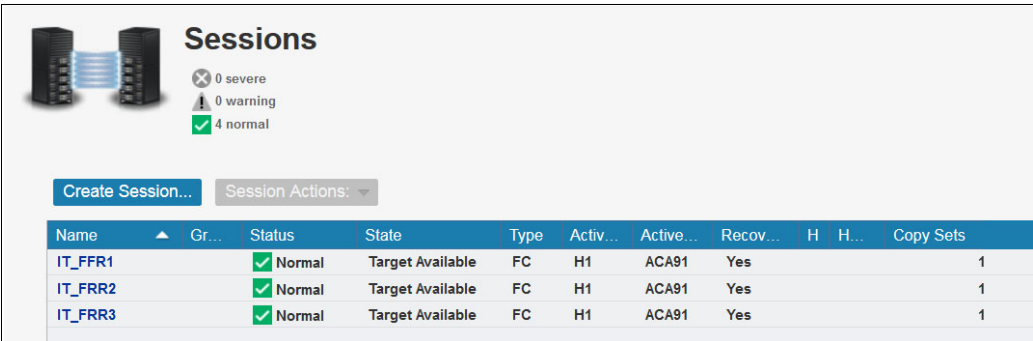


Figure 2-8 Three active FlashCopy sessions from the same source volume

- Click a dedicated session name under the *Name* heading to get details about the session itself. Figure 2-9 shows session details. Particularly under Role Progress, CSM displays the corresponding CKD volume labels (VOLSER) and the remaining tracks, which are not replicated to the target volume yet. Remember that this is a NOCOPY relationship, and only a copy-on-write will update the relevant target tracks.

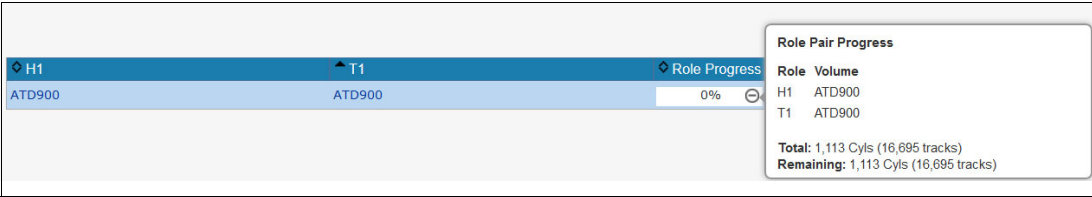


Figure 2-9 Display session details about session IT\_FRR2

### Restoring FlashCopy volumes

In Figure 2-10, the IT\_FRR2 session is selected and the **Commands** → **Restore** action is selected to restore the source volume from the second target volume. Figure 2-10 still shows the original FlashCopy relationships from H1 to T1.

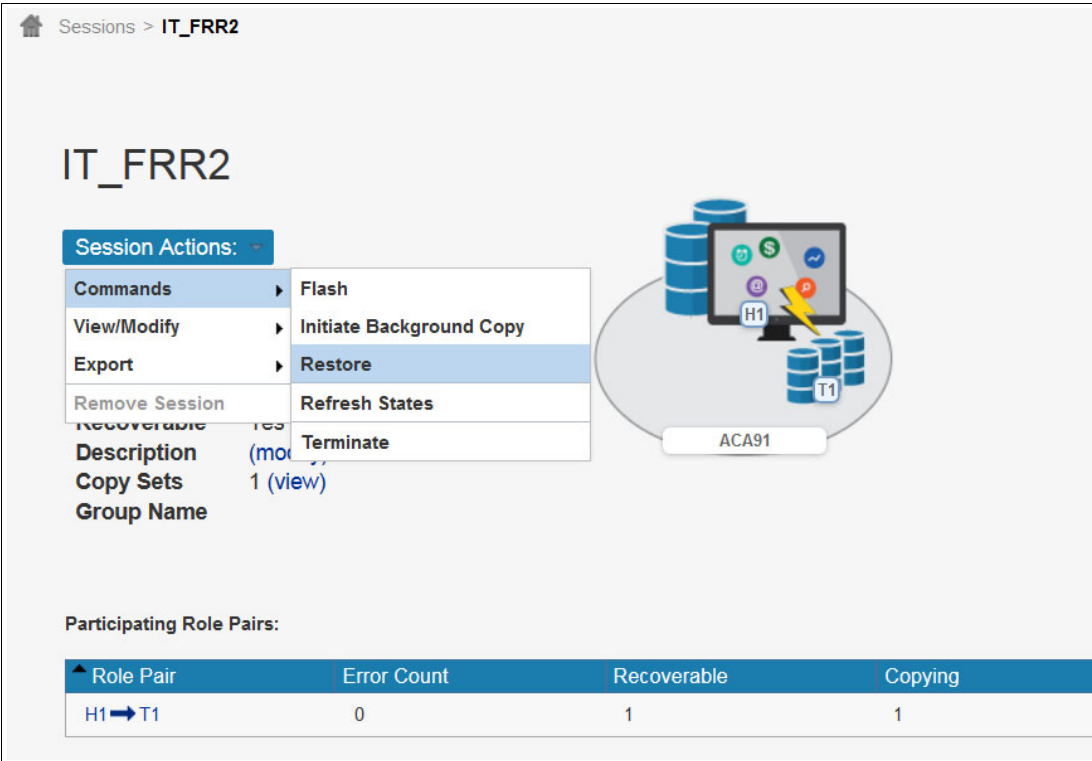


Figure 2-10 Restore FlashCopy session number 2, IT\_FRR2

As shown in Figure 2-11, CSM is as always asking for confirmation to perform the selected action.

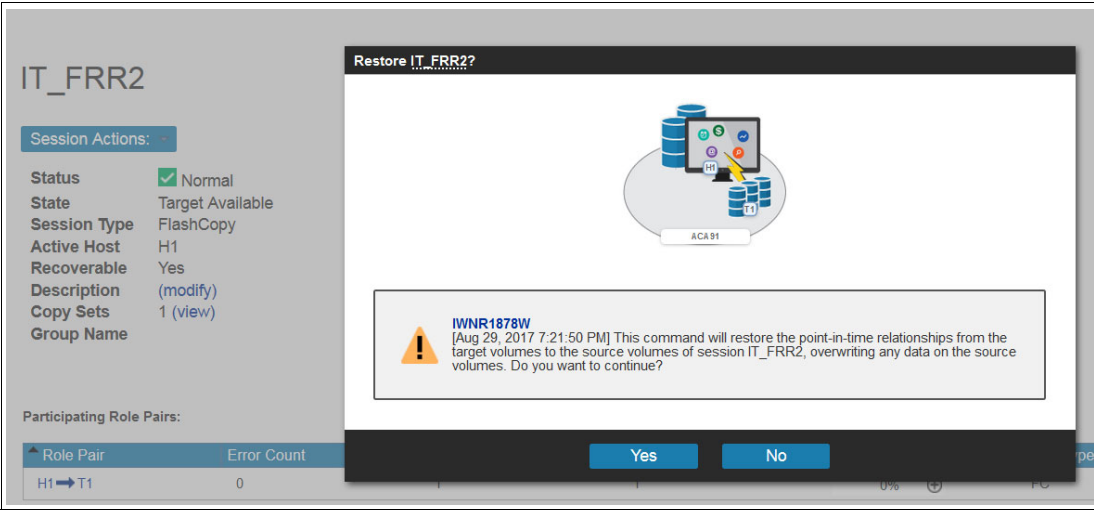


Figure 2-11 Acknowledge restore operations

Figure 2-12 shows the outcome of the CSM restore process. Note in Figure 2-12 that the arrow under the Role Pair heading now points from Target to Source volume (T1 to H1).

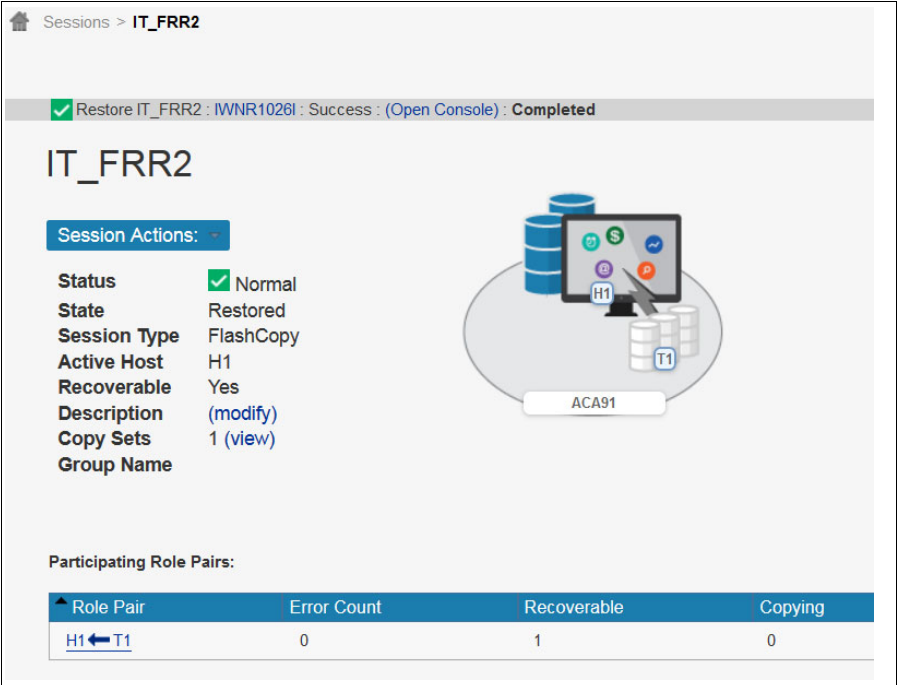


Figure 2-12 Restore completed successfully

Checking the CSM Console log verifies the steps that led to the CSM restore state. See Figure 2-13.

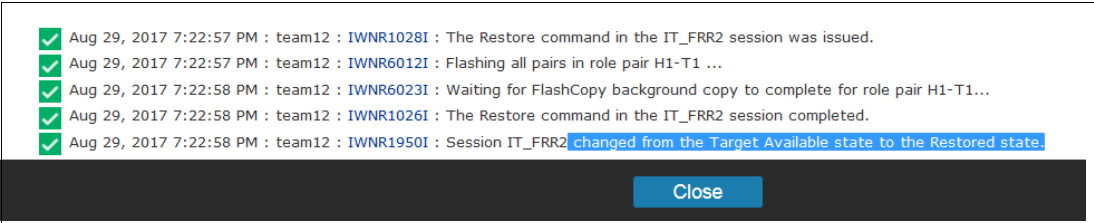


Figure 2-13 CSM Console log

Figure 2-14 displays all three FlashCopy sessions at this stage, and reflects the cascaded FlashCopy support. Session IT\_FRR2 shows as Restored, but sessions IT\_FRR1 and IT\_FRR3 stay unchanged, maintaining their FlashCopy relationship.

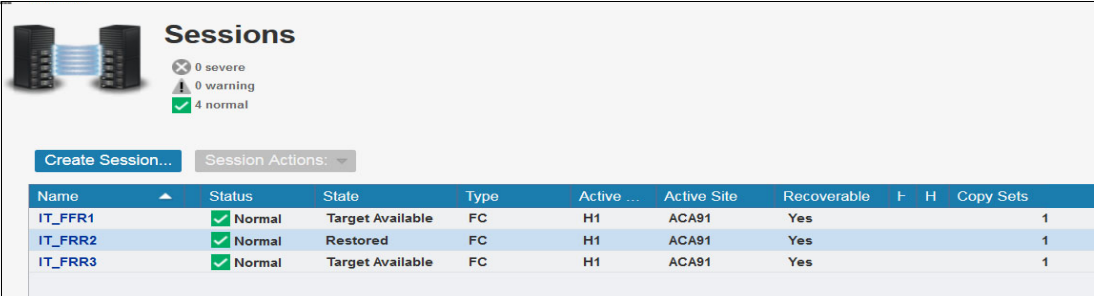


Figure 2-14 FlashCopy session IT\_FRR2 is restored.

## 2.2 Global Mirror configuration and multiple FC targets

This use case provides more data availability options through cascading FlashCopy support in a Global Mirror (GM) configuration with multiple FlashCopy sessions from the same FlashCopy source volume, as illustrated in Figure 2-15.

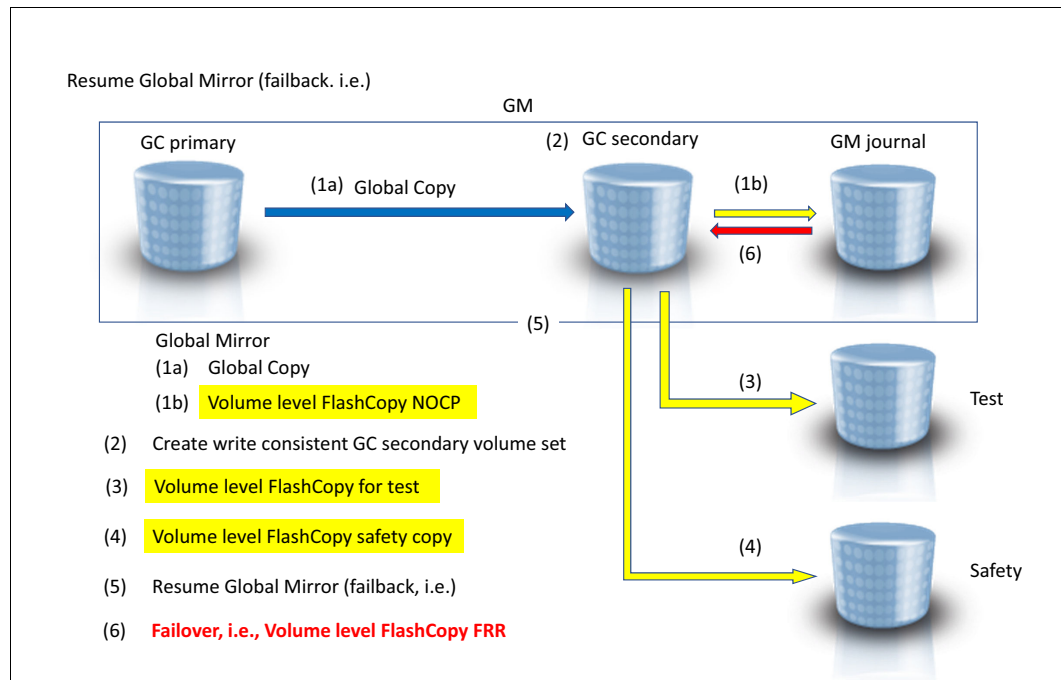


Figure 2-15 GM configuration and multiple FlashCopy target volumes

Figure 2-15 shows a GM configuration that includes a Global Copy with its associated FlashCopy relationship. Additionally, a Test FC volume is created for test and data verification purposes. Another FC volume is also created as a safety copy (no-UCB, for example) to provide a recovery point when the data in the GM configuration is corrupted or no longer usable for any reason. Before step (3) and step (4) in Figure 2-15, a GM failover process is required to create write-consistent data on the GC secondary volumes set. Note that there are other GM actions possible to create a consistent set of GC secondary volumes.

Step (1a) and (1b) create a GM session.

Step (2) creates a write consistent volume set for GC secondary volumes. Most likely, this is achieved through a GM failover process that includes an FRR. A GM failback process reestablishes the GM process (5).

So each time before step (3) and step (4) in Figure 2-15, a process is required to create write consistent data on the GC secondary volumes set (2), which serve as FlashCopy source volumes for all three FlashCopy target volumes.

Test is used as the source for the test efforts. The FlashCopy safety copy can be created to a no-UCB device (IBM Z) to provide a data copy, which cannot be corrupted through Ransomware attacks or destructive attempts to the production data. It can also be used to serve either as a source for potential application restarts, or as FRR source to put the data copy from the safety copy back to the GC secondary or FlashCopy the safety copy to any other application or production volume.

## 2.3 Simple forward cascading FlashCopy

This use case illustrates a pure forward cascading FlashCopy scenario.

Figure 2-16 shows a combination of cascading FlashCopy configuration with multiple target FlashCopy volumes from the same source volume.

In this use case, we created several FlashCopy target volumes from the same source volumes, as would be the case in a standard application production environment, simply to retain copies of application data at different times in the production process.

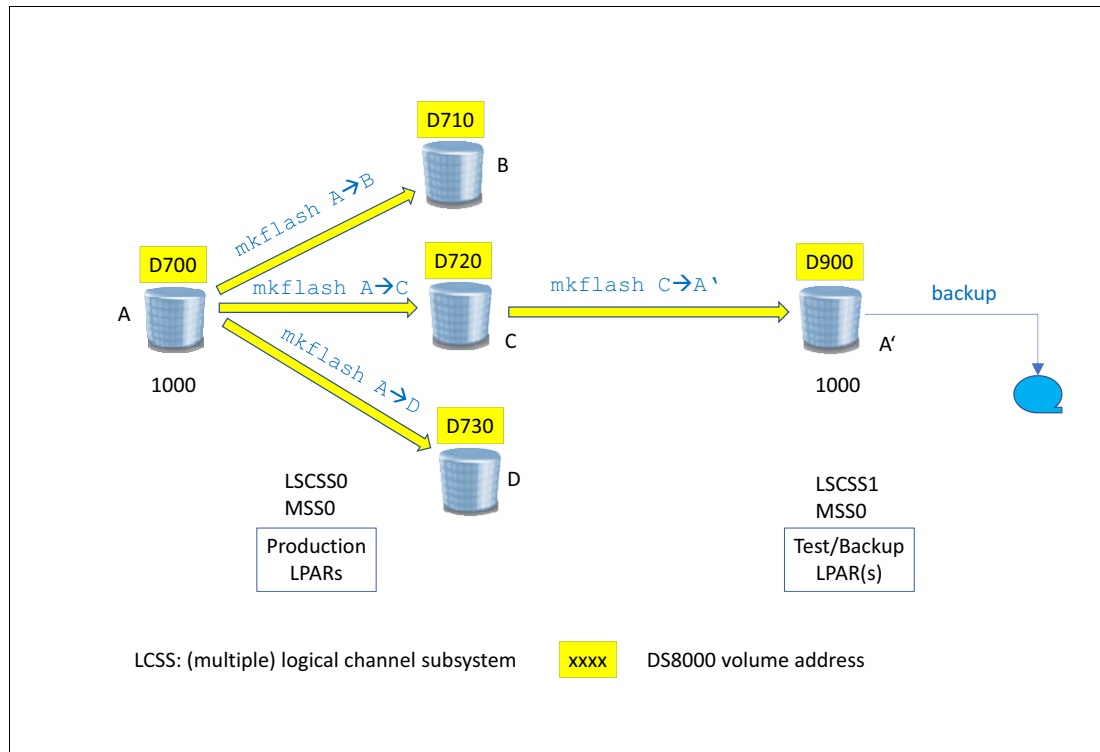


Figure 2-16 Use Cascaded FlashCopy

Test might pick a certain level of the production data for test efforts or verifying the actual production data and perhaps eventually create a backup onto tape from volume A'.

With Cascading FlashCopy support, this configuration allows for a restore of a certain copy of the data from any of the existing FlashCopy target volumes (B,C, or D) without impacting any other existing FlashCopy relationships.

**Tip:** Although not new, and not related to the cascading FlashCopy support, remember when you create more than one FlashCopy relationship to create the relationships with the consistency group attribute to guarantee a consistent write to all target volumes within the entire set of FlashCopy target volumes. Also, to keep the commands in the following examples simple, the FlashCopy consistency group parameter **-freeze** is not coded.

Example 2-2 illustrates the DSCLI commands required to create the configuration shown in Figure 2-16 on page 23.

*Example 2-2 Establish multiple target and cascaded FC relationships*

```

dscli> mkflash -nocp -persist D700:D710
Date/Time: 7. September 2017 14:44:07 CEST IBM DSCLI Version: 7.8.30.435 DS:
IBM.2107-75XXXX1
CMUC00137I mkflash: FlashCopy pair D700:D710 successfully created.

dscli> mkflash -nocp -persist D700:D720
Date/Time: 7. September 2017 14:44:33 CEST IBM DSCLI Version: 7.8.30.435 DS:
IBM.2107-75XXXX1
CMUC00137I mkflash: FlashCopy pair D700:D720 successfully created.

dscli> mkflash -nocp -persist D700:D730
Date/Time: 7. September 2017 14:44:51 CEST IBM DSCLI Version: 7.8.30.435 DS:
IBM.2107-75XXXX1
CMUC00137I mkflash: FlashCopy pair D700:D730 successfully created.

dscli> mkflash -nocp -persist D720:D900
Date/Time: 7. September 2017 14:45:17 CEST IBM DSCLI Version: 7.8.30.435 DS:
IBM.2107-75XXXX1
CMUC00137I mkflash: FlashCopy pair D720:D900 successfully created.

dscli> lsflash d700-d9ff
Date/Time: 7. September 2017 14:45:39 CEST IBM DSCLI Version: 7.8.30.435 DS:
IBM.2107-75XXXX1
ID          SrcLSS SequenceNum Timeout ActiveCopy Recording Persistent Revertible
SourceWriteEnabled TargetWriteEnabled BackgroundCopy
=====
=====
D700:D710 D7      0          120    Disabled Disabled Enabled Disabled
Enabled      Enabled      Disabled
D700:D720 D7      0          120    Disabled Disabled Enabled Disabled
Enabled      Enabled      Disabled
D700:D730 D7      0          120    Disabled Disabled Enabled Disabled
Enabled      Enabled      Disabled
D720:D900 D7      0          120    Disabled Disabled Enabled Disabled
Enabled      Enabled      Disabled
dscli>

```

### 2.3.1 Update FlashCopy target volume

To explore the scenario in Figure 2-16 on page 23, we update the FlashCopy target volume on device number **D730**.

First, we list the table of contents for device **D700**, as shown in Example 2-3.

*Example 2-3 List table of contents of device D700*

```

DSLST - Data Sets on volume ITD700                               Row 1 of 2
Command ==>                                                         Scroll ==> CSR

Command - Enter "/" to select action                                Message                                Volume
-----

```

```

SYS1.VTOCIX. ITD700
TEAM12.D700.TEXT
***** End of Data Set list *****

```

Device **D700** contains a user data set TEAM12.D700.TEXT (second data set name qualifier corresponds to the device number on which the data set was created) and its VOLSER is **ITD700**.

To gain access to this device **D730** device **D700** is varied offline first and then vary device **D730** online.

Example 2-4 show the z/OS system commands to gain access to device D730.

*Example 2-4 Gain access to FlashCopy target volume on device D730*

**V D700,OFFLINE**

```
IEF281I D700 NOW OFFLINE
```

**V D730,ONLINE**

```
IEE302I D730 ONLINE
```

**DS P,D730,1**

```

IEE459I 15.07.11 DEVSERV PATHS 198
UNIT DTYPE MD CNT VOLSER CHPID=PATH STATUS
RTYPE SSID CFW TC DFW PIN DC-STATE CCA DDC CYL CU-TYPE
OD730,33903 ,0 ,000, ITD700,9C=+ A6=+ AE=+ AF=+
2107 D701 Y YY. YY. N SIMPLEX 30 30 1113 2107
***** SYMBOL DEFINITIONS *****
0 = ONLINE + = PATH AVAILABLE

```

From now on, device **D730** is accessible under the same VOLSER **ITD700** as for device **D700**. **D700** became unavailable to the system.

Listing the table of contents of volume ITD700 in Example 2-5 now lists the volume on device number **D730**. Its content is identical with what **D700** listed before, because it results from a FlashCopy of **D700** (see Example 2-2 on page 24).

*Example 2-5 List table of content of FlashCopy target volume on device number D730*

```

DSLST - Data Sets on volume ITD700
Command ==>

Command - Enter "/" to select action
-----
          SYS1.VTOCIX. ITD700
          TEAM12.D700.TEXT
          ***** End of Data Set list *****

```

Now we create another data set to the FlashCopy target volume on device number **D730**, which is currently online to the system, as shown in Example 2-6.

*Example 2-6 Add data set to FlashCopy target volume on device number D730*

```

DSLST - Data Sets on volume ITD700
Command ==>

Command - Enter "/" to select action
-----
          SYS1.VTOCIX. ITD700
          TEAM12.D700.TEXT
          ***** End of Data Set list *****

```

```

-----
SYS1.VTOCIX.1TD700                                1TD700
TEAM12.D700.TEXT                                    1TD700
TEAM12.D730.TEXT                                    1TD700
***** End of Data Set list *****
-----

```

Because **D730** was modified, the number of OOS tracks changed for device number **D730** can be verified with a **lsflash** DSCSI command, as shown in Example 2-7.

*Example 2-7 OOS tracks on FlashCopy target volume D730*

```

dsccli> lsflash -l d700-d730
Date/Time: 7. September 2017 15:11:59 CEST IBM DSCSI Version: 7.8.30.435 DS:
IBM.2107-75xxxxx
ID          SrcLSS SequenceNum Timeout ActiveCopy Recording Persistent Revertible
SourceWriteEnabled TargetWriteEnabled BackgroundCo
py OutOfSyncTracks DateCreated          DateSynced
State isTgtSE Pmir
=====
=====
=====
D700:D710 D7      0          120      Disabled  Disabled  Enabled   Disabled
Enabled          Enabled          Disabled
16695          Thu Sep 07 14:44:05 CEST 2017 Thu Sep 07 14:44:05 CEST 2017
Valid No        No
D700:D720 D7      0          120      Disabled  Disabled  Enabled   Disabled
Enabled          Enabled          Disabled
16695          Thu Sep 07 14:44:32 CEST 2017 Thu Sep 07 14:44:32 CEST 2017
Valid No        No
D700:D730 D7      0          120      Disabled  Disabled  Enabled   Disabled
Enabled          Enabled          Disabled
16428          Thu Sep 07 14:44:49 CEST 2017 Thu Sep 07 14:44:49 CEST 2017
Valid No        No
=====

```

Example 2-7 shows that there is no change in the number of OOS tracks for FlashCopy relationships except for the FlashCopy relationship **D700:D730**. In our example, 16695-16428=267 tracks have changed.

## 2.3.2 Reverse FlashCopy

Now, as shown in Figure 2-17, we reverse the **D700:D730** relationship and apply all of the changes that had occurred to **D730** (267 tracks) back to **D700**. **D700** is now the target of a FlashCopy relationship. Again, thanks to the cascading FlashCopy feature, the other existing FlashCopy relationships (with **D700** as the source volume) are not affected and remain unchanged.

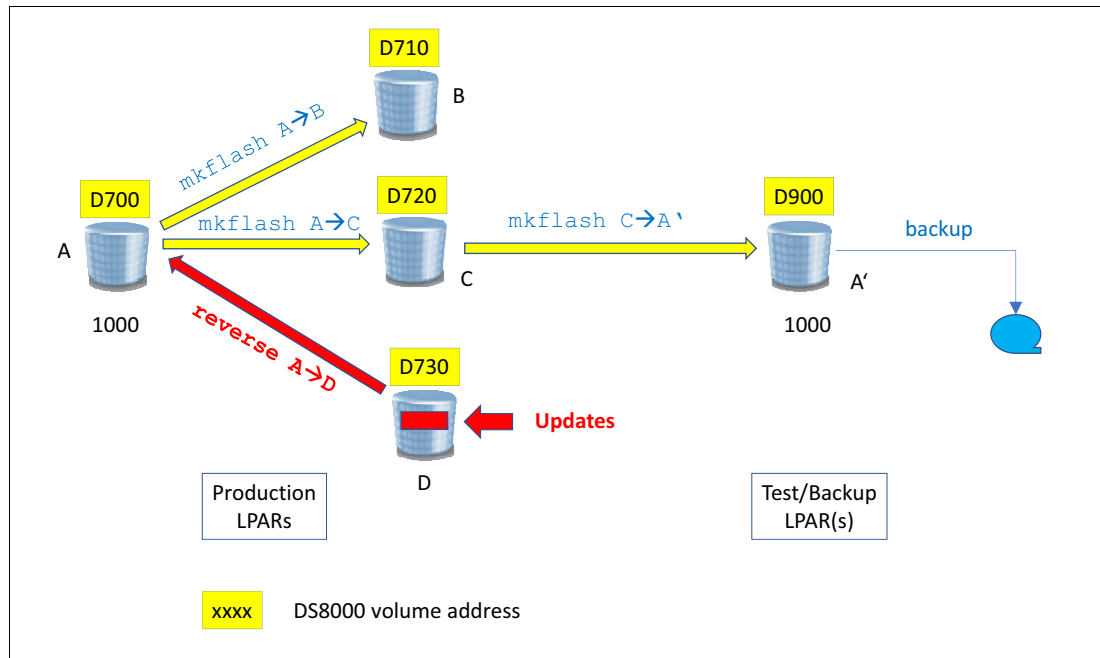


Figure 2-17 Reverse a FlashCopy relationship without affecting existing FlashCopy relationships

Example 2-8 shows the DSCLI commands used to reverse and list the FlashCopy relationship **D700:D730**.

The Fast Reverse Restore (FRR) applies all changed tracks in **D730** back to the original FlashCopy source volume **D700**. When FRR completes, the FlashCopy relationship between **D700:D730** ends.

*Example 2-8 Reverse FlashCopy relationship D700:D730 and list all FlashCopy relations*

```
dscli>
dscli> reverseflash -fast D700:D730
Date/Time: 7. September 2017 15:20:24 CEST IBM DSCLI Version: 7.8.30.435 DS:
IBM.2107-75xxxxx
CMUC00169I reverseflash: FlashCopy volume pair D700:D730 successfully reversed.
dscli>
```

```
dscli> lsflash -l d700-d9ff
Date/Time: 7. September 2017 15:20:36 CEST IBM DSCLI Version: 7.8.30.435 DS:
IBM.2107-75xxxxx
ID          SrcLSS SequenceNum Timeout ActiveCopy Recording Persistent Revertible
SourceWriteEnabled TargetWriteEnabled BackgroundCo
py OutOfSyncTracks DateCreated          DateSynced
State isTgtSE Pmir
```

```

=====
=====
D700:D710 D7      0      120      Disabled      Disabled      Enabled      Disabled
Enabled          Enabled          Disabled
16141          Thu Sep 07 14:44:05 CEST 2017 Thu Sep 07 14:44:05 CEST 2017
Valid No      No
D700:D720 D7      0      120      Disabled      Disabled      Enabled      Disabled
Enabled          Enabled          Disabled
16141          Thu Sep 07 14:44:32 CEST 2017 Thu Sep 07 14:44:32 CEST 2017
Valid No      No
D720:D900 D7      0      120      Disabled      Disabled      Enabled      Disabled
Enabled          Enabled          Disabled
16694          Thu Sep 07 14:45:15 CEST 2017 Thu Sep 07 14:45:15 CEST 2017
Valid No      No
dscli>

```

The **lsflash** command in Example 2-8 lists all FlashCopy relationships when FRR has completed. The data level on D730 is now useless. It might be feasible to reestablish the original FlashCopy relationship **D700:D730** (see Example 2-11 on page 29).

Because FRR changes tracks in **D700**, the number of OOS tracks between the **D700** FlashCopy source volumes and other targets changes also. These changes are immediately applied to all associated target volumes. The cascaded relationship **D720:D900** remains and stays unchanged.

If we bring **D700** back online, we can expect to see the changed data from **D730**.

Example 2-9 again shows the required z/OS **vary** commands to make **D700** accessible again.

*Example 2-9 Return to device number D700*

```

V D730,OFFLINE
IEF281I D730 NOW OFFLINE
V D700,ONLINE
IEE302I D700      ONLINE

DS P,D730,1
IEE459I 15.16.47 DEVSERV PATHS 238
UNIT DTYPE MD CNT VOLSER CHPID=PATH STATUS
RTYPE SSID CFW TC DFW PIN DC-STATE CCA DDC CYL CU-TYPE
OD730,33903 ,F ,000, ,9C=+ A6=+ AE=+ AF=+
2107 D701 Y YY. YY. N SIMPLEX 30 30 1113 2107
***** SYMBOL DEFINITIONS *****
F = OFFLINE + = PATH AVAILABLE

DS P,D700,1
IEE459I 15.16.53 DEVSERV PATHS 240
UNIT DTYPE MD CNT VOLSER CHPID=PATH STATUS
RTYPE SSID CFW TC DFW PIN DC-STATE CCA DDC CYL CU-TYPE
OD700,33903 ,0 ,000,ITD700,9C=+ A6=+ AE=+ AF=+
2107 D701 Y YY. YY. N SIMPLEX 00 00 1113 2107
***** SYMBOL DEFINITIONS *****
0 = ONLINE + = PATH AVAILABLE

```

Now volume ITD700 is online again on device address **D700**. Device address **D730** became offline to the concerned z/OS image(s).

For your reference, duplicate volume serial numbers and their implications in z/OS environments are discussed in *DS8000 Copy Services*, SG24-8367.

Finally, list the table of contents of volume ITD700 on device address **D700**, as shown in Example 2-10.

Example 2-10 List table of contents of volume on device number D700

DSLIS - Data Sets on volume ITD700		Row 1 of 3
Command ==>		Scroll ==> CSR
Command - Enter "/" to select action	Message	Volume
-----		
SYS1.VTOCIX.	ITD700	ITD700
TEAM12.D700.TEXT		ITD700
TEAM12.D730.TEXT		ITD700
***** End of Data Set list *****		

The data currency between **D700:D730** is identical, as shown in Example 2-11. The other FlashCopy targets from **D700**, **D710**, and **D720**, remain unchanged, as illustrated in Figure 2-17 on page 27.

Example 2-11 Reestablish original FlashCopy relationship

dscli> mkflash -nocp -persist D700:D730							
Date/Time: 7. September 2017 15:21:14 CEST IBM DSCLI Version: 7.8.30.435 DS: IBM.2107-75xxxxx							
CMUC00137I mkflash: FlashCopy pair D700:D730 successfully created.							
dscli>							
dscli> lsflash -l d700-d9ff							
Date/Time: 7. September 2017 15:21:30 CEST IBM DSCLI Version: 7.8.30.435 DS: IBM.2107-75xxxxx							
ID	SrcLSS	SequenceNum	Timeout	ActiveCopy	Recording	Persistent	Revertible
SourceWriteEnabled	TargetWriteEnabled	BackgroundCo					
py OutOfSyncTracks	DateCreated				DateSynced		
State	isTgtSE	Pmir					
=====							
=====							
=====							
<b>D700:D710</b>	D7	0	120	Disabled	Disabled	Enabled	Disabled
Enabled		Enabled		Disabled			
16141		Thu Sep 07 14:44:05	CEST 2017	Thu Sep 07 14:44:05	CEST 2017		
Valid No	No						
<b>D700:D720</b>	D7	0	120	Disabled	Disabled	Enabled	Disabled
Enabled		Enabled		Disabled			
16141		Thu Sep 07 14:44:32	CEST 2017	Thu Sep 07 14:44:32	CEST 2017		
Valid No	No						
<b>D700:D730</b>	D7	0	120	Disabled	Disabled	Enabled	Disabled
Enabled		Enabled		Disabled			
16695		Thu Sep 07 15:21:12	CEST 2017	Thu Sep 07 15:21:12	CEST 2017		
Valid No	No						
<b>D720:D900</b>	D7	0	120	Disabled	Disabled	Enabled	Disabled
Enabled		Enabled		Disabled			

Finally, reestablish the relationship between **D700:D730** and return to the configuration shown in Figure 2-16 on page 23.

However the data content of the involved volumes is now different. **D700** and **D730** are identical but different than the other three volumes, which have remained unchanged.

This also demonstrates that such configurations tend to become complex and hard to manually manage. In such circumstances, it is advisable to use a management tool like Copy Services Manager to manage Copy Services configurations.

## 2.4 Cascading data set level FlashCopy

Data set level FlashCopy is unique to z/OS. Therefore, the following paragraphs relate to FlashCopy CKD volumes and z/OS data sets.

The scenario discussed here would also not have been possible before support for cascading FlashCopy. An attempt to create a data set copy through DFSMSdss using FlashCopy (Fast Replication required) with the target data set on a FlashCopy source volume fails without cascaded FlashCopy support.

Figure 2-18 shows the use case scenario. FlashCopy relationships exist for all application volumes represented by source devices **D700** and **D701**. The corresponding target devices are **D900** and **D901**.

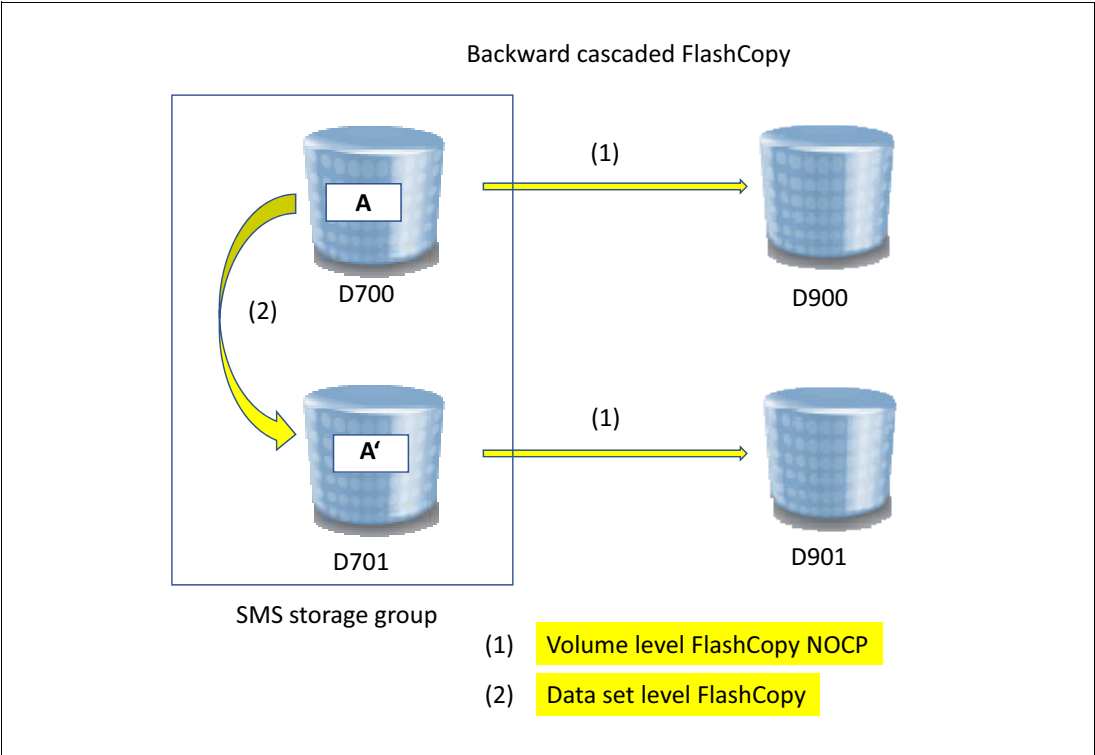


Figure 2-18 Data set level or volume level FlashCopy creating a backward-cascaded FlashCopy relationship

Now, a data set level FlashCopy is created through a DFSMSdss data set level copy.

First, query the status of the source and target volumes through the device service z/OS system command **DEVSERV** or short **DS**, as shown in Example 2-12. In this use case, we work with volumes that are not system-managed, making it possible to query the content of the FlashCopy target volume.

*Example 2-12 Query FlashCopy source volumes and FlashCopy target volumes*

---

```

DS P,D700,2
IEE459I 16.18.38 DEVSERV PATHS 511
UNIT DTYPE MD CNT VOLSER CHPID=PATH STATUS
      RTYPE SSID CFW TC DFW PIN DC-STATE CCA DDC      CYL CU-TYPE
OD700,33903 ,0 ,000,ITD700,9C=+ A6=+ AE=+ AF=+
      2107 D701 Y YY. YY. N SIMPLEX 00 00      1113 2107
OD701,33903 ,0 ,000,ITD701,9C=+ A6=+ AE=+ AF=+
      2107 D701 Y YY. YY. N SIMPLEX 01 01      1113 2107
***** SYMBOL DEFINITIONS *****
O = ONLINE                      + = PATH AVAILABLE

DS P,D900,2
IEE459I 16.19.13 DEVSERV PATHS 521
UNIT DTYPE MD CNT VOLSER CHPID=PATH STATUS
      RTYPE SSID CFW TC DFW PIN DC-STATE CCA DDC      CYL CU-TYPE
OD900,33903 ,0 ,000,ATD900,9C=+ A6=+ AE=+ AF=+
      2107 D901 Y YY. YY. N SIMPLEX 00 00      1113 2107
OD901,33903 ,0 ,000,ATD901,9C=+ A6=+ AE=+ AF=+
      2107 D901 Y YY. YY. N SIMPLEX 01 01      1113 2107
***** SYMBOL DEFINITIONS *****
O = ONLINE                      + = PATH AVAILABLE

```

---

Example 2-13 shows the DSCLI commands used to create the FlashCopy relationships shown in Figure 2-18 on page 30.

*Example 2-13 Create FlashCopy relationships and list all FlashCopy relationships*

---

```

dscli> mkflash -nocp -persist D700:D900
Date/Time: 7. September 2017 16:27:33 CEST IBM DSCLI Version: 7.8.30.435 DS:
IBM.2107-75xxxx
CMUC00137I mkflash: FlashCopy pair D700:D900 successfully created.

dscli> mkflash -nocp -persist D701:D901
Date/Time: 7. September 2017 16:27:41 CEST IBM DSCLI Version: 7.8.30.435 DS:
IBM.2107-75xxxx
CMUC00137I mkflash: FlashCopy pair D701:D901 successfully created.

dscli>
dscli> lsflash d700-d9ff
Date/Time: 7. September 2017 16:27:52 CEST IBM DSCLI Version: 7.8.30.435 DS:
IBM.2107-75xxxx
ID          SrcLSS SequenceNum Timeout ActiveCopy Recording Persistent Revertible
SourceWriteEnabled TargetWriteEnabled BackgroundCo
py
=====
=====
==

```

```

D700:D900 D7      0      120    Disabled  Disabled  Enabled   Disabled
Enabled          Enabled          Disabled
D701:D901 D7      0      120    Disabled  Disabled  Enabled   Disabled
Enabled          Enabled          Disabled
dscli>

```

Example 2-14 shows the content of a FlashCopy source and its associated target volume.

Because of the FlashCopy commands issued in Example 2-13 on page 31, both volumes appear empty except for a system data set on both volumes.

*Example 2-14 List volume content of a FlashCopy source volume and its target volume*

```

DSLIS - Data Sets on volume ITD700                                     Row 1 of 1
Command ==>                                                         Scroll ==> CSR

Command - Enter "/" to select action                                Message                                Volume
-----
          SYS1.VTOCIX.ITD700                                         ITD700
***** End of Data Set list *****

DSLIS - Data Sets on volume ATD900                                     Row 1 of 1
Command ==>                                                         Scroll ==> CSR

Command - Enter "/" to select action                                Message                                Volume
-----
          SYS1.VTOCIX.ITD700                                         ATD900
***** End of Data Set list *****

```

Example 2-14 shows the mismatch of Volume (Serial Number) of ATD900 on device D900, which contains the system data set SYS1.VTOCIX.ITD700 from volume ITD700 on device D700.

Next, we add a new data set (TEAM12.D700.TEXT) to the FlashCopy source volume D700. Example 2-15 shows all four involved volumes.

*Example 2-15 New data set on FlashCopy source volume on D700*

```

DSLIS - Data Sets on volume ITD700                                     Row 1 of 2
Command ==>                                                         Scroll ==> CSR

Command - Enter "/" to select action                                Message                                Volume
-----
          SYS1.VTOCIX.ITD700                                         ITD700
          TEAM12.D700.TEXT                                           ITD700
***** End of Data Set list *****

DSLIS - Data Sets on volume ITD701                                     Row 1 of 1
Command ==>                                                         Scroll ==> CSR

Command - Enter "/" to select action                                Message                                Volume
-----
          SYS1.VTOCIX.ITD701                                         ITD701
***** End of Data Set list *****

DSLIS - Data Sets on volume ATD900                                     Row 1 of 1

```

Command ==> Scroll ==> CSR

Command - Enter "/" to select action Message Volume

-----  
SYS1.VTOCIX. **ITD700** **ATD900**  
\*\*\*\*\* End of Data Set list \*\*\*\*\*

DSLIST - Data Sets on volume **ATD901** Row 1 of 1  
Command ==> Scroll ==> CSR

Command - Enter "/" to select action Message Volume

-----  
SYS1.VTOCIX. **ITD701** **ATD901**  
\*\*\*\*\* End of Data Set list \*\*\*\*\*

Now, we do a data set level FlashCopy to copy data set TEAM12.D700.TEXT on ITD700 to volume ITD701 under the new name of TEAM12.ITD701.TEXT.

We must use DFSMSDss to perform such a data set level copy. Example 2-16 shows a DFSMSDss job step to create a data set level FlashCopy on volume ITD701 on device **D700**. The parameter **FR(REQ)** dictates that FlashCopy must be used to create the new data set on volume ITD701.

*Example 2-16 Create new data set through data set level FlashCopy*

```
/* ----- ***
//COPY1 EXEC PGM=ADRDSSU
//SYSPRINT DD SYSOUT=*
//IO1 DD UNIT=3390,VOL=SER=ITD700,DISP=SHR
//001 DD UNIT=3390,VOL=SER=ITD701,DISP=SHR
//SYSIN DD *

COPY DS( INC(TEAM12.D700.TEXT)) -
      FR(REQ) -
      RENUNC((TEAM12.D700.TEXT -
                TEAM12.D701.TEXT )) -
      CAT -
      DEBUG(FRMSG(DTL)) -
      STORCLAS(NONSMS) -
      OUTDD(001)

/*
//* ----- JOB END ----- ***
//
```

Example 2-17 shows that FlashCopy was successfully invoked to create another data set copy with a new name on volume ITD701.

*Example 2-17 DFSMSDss job step output and catalog entries of both data sets*

PAGE 0001 5695-DF175 DFSMSDSS V2R02.0 DATA SET SERVICES 2017.250 16:46

```
COPY DS( INC(TEAM12.D700.TEXT)) -
      FR(REQ) -
      RENUNC((TEAM12.D700.TEXT -
                TEAM12.D701.TEXT )) -
```

```

CAT -
DEBUG(FRMSG(DTL)) -
STORCLAS(NONSMS) -
OUTDD(001)
ADR101I (R/I)-RI01 (01), TASKID 001 HAS BEEN ASSIGNED TO COMMAND 'COPY '

ADR109I (R/I)-RI01 (01), 2017.250 16:46:42 INITIAL SCAN OF USER CONTROL STATEMENTS
COMPLETED
ADR016I (001)-PRIME(01), RACF LOGGING OPTION IN EFFECT FOR THIS TASK
ADR006I (001)-STEND(01), 2017.250 16:46:42 EXECUTION BEGINS
ADR395I (001)-NEWDS(01), DATA SET TEAM12.D700.TEXT ALLOCATED WITH NEWNAME
TEAM12.D701.TEXT, ON VOLUME(S): ITD701
ADR806I (001)-TOMI (01), DATA SET TEAM12.D700.TEXT COPIED USING A FAST REPLICATION
FUNCTION
ADR465I (001)-CNVSM(01), DATA SET TEAM12.D701.TEXT HAS BEEN CATALOGED IN CATALOG
CATALOG.MVSICF1.VVSLWK1
ADR801I (001)-DDDS (01), 2017.250 16:46:42 DATA SET FILTERING IS COMPLETE. 1 OF 1
DATA SETS WERE SELECTED: 0 FAILED SERIALIZATION
AND 0 FAILED FOR OTHER REASONS
ADR454I (001)-DDDS (02), THE FOLLOWING DATA SETS WERE SUCCESSFULLY PROCESSED
TEAM12.D700.TEXT
ADR006I (001)-STEND(02), 2017.250 16:46:42 EXECUTION ENDS
ADR013I (001)-CLTSK(01), 2017.250 16:46:42 TASK COMPLETED WITH RETURN CODE 0000
ADR012I (SCH)-DSSU (01), 2017.250 16:46:42 DFSMSDSS PROCESSING COMPLETE. HIGHEST
RETURN CODE IS 0000
***** BOTTOM OF DATA *****

NONVSAM ----- TEAM12.D700.TEXT
IN-CAT --- CATALOG.MVSICF1.VVSLWK1
HISTORY
DATASET-OWNER----- (NULL) CREATION-----2017.250
RELEASE-----2 EXPIRATION-----0000.000
VOLUMES
VOLSER-----ITD700 DEVTYPE-----X'3010200F'
NONVSAM ----- TEAM12.D701.TEXT
IN-CAT --- CATALOG.MVSICF1.VVSLWK1
HISTORY
DATASET-OWNER----- (NULL) CREATION-----2017.250
RELEASE-----2 EXPIRATION-----0000.000
VOLUMES
VOLSER-----ITD701 DEVTYPE-----X'3010200F'

```

---

This operation implicitly created a backward-cascaded FlashCopy relationship for the time being when the data set copy was performed by FlashCopy in the DS8000 from source device **D700** to target device **D701**. **D701** is at the same time also the source device to the target device **D901**.

Example 2-17 on page 33 also shows the associated catalog entries for both data sets and proves that they exist on both FlashCopy source volumes

As before, without cascading FlashCopy support, to refresh an existing FlashCopy relationship it is required to delete the existing FlashCopy relationships first. This also applies to cascading FlashCopy relationships, no matter how long the cascading FlashCopy chain is.

Directly refreshing a FlashCopy relationship is only possible for incremental FlashCopy relationships as soon as the first full background replication finished. Here, no **rmflash** is required and another **mkflash** is sufficient to refresh an incremental FlashCopy relationship.

To do a new FlashCopy between D700:D900 and D701:D901 as shown in Example 2-18 assumes that the initial FlashCopy relationships had been previously removed, using the DSCLI command **rmflash**.

*Example 2-18 Refresh FlashCopy relationship*

---

```

dscli> mkflash -nocp -persist D700:D900
Date/Time: 7. September 2017 16:57:30 CEST IBM DSCLI Version: 7.8.30.435 DS:
IBM.2107-75ACA91
CMUC00137I mkflash: FlashCopy pair D700:D900 successfully created.

dscli> mkflash -nocp -persist D701:D901
Date/Time: 7. September 2017 16:57:47 CEST IBM DSCLI Version: 7.8.30.435 DS:
IBM.2107-75ACA91
CMUC00137I mkflash: FlashCopy pair D701:D901 successfully created.

dscli> lsflash d700-d9ff
Date/Time: 7. September 2017 16:58:14 CEST IBM DSCLI Version: 7.8.30.435 DS:
IBM.2107-75ACA91
ID          SrcLSS SequenceNum Timeout ActiveCopy Recording Persistent Revertible
SourceWriteEnabled TargetWriteEnabled BackgroundCo
py
=====
=====
==
D700:D900 D7      0          120    Disabled   Disabled   Enabled   Disabled
Enabled                                Disabled
D701:D901 D7      0          120    Disabled   Disabled   Enabled   Disabled
Enabled                                Disabled
dscli>

```

---

Example 2-19 shows all involved volumes and their contents, resulting from the new FlashCopy relationships created under Example 2-18.

*Example 2-19 Show updated volume content*

---

```

DSLST - Data Sets on volume ITD700
Command ==>
Row 1 of 2
Scroll ==> CSR
.
.
Command - Enter "/" to select action      Message      Volume
-----
SYS1.VTOCIX.ITD700                        ITD700
TEAM12.D700.TEXT                          ITD700
***** End of Data Set list *****

DSLST - Data Sets on volume ITD701
Command ==>
Row 1 of 2
Scroll ==> CSR
.
.
Command - Enter "/" to select action      Message      Volume
-----
SYS1.VTOCIX.ITD701                        ITD701
TEAM12.D701.TEXT                          ITD701

```

---

\*\*\*\*\* End of Data Set list \*\*\*\*\*

DSLIST - Data Sets on volume **ATD900** Row 1 of 2 .  
Command ==> Scroll ==> CSR .

Command - Enter "/" to select action	Message	Volume
-----		
SYS1.VTOCIX. <b>ITD700</b>		<b>ATD900</b>
TEAM12. <b>D700</b> .TEXT		<b>ATD900</b>

\*\*\*\*\* End of Data Set list \*\*\*\*\*

DSLIST - Data Sets on volume **ATD901** Row 1 of 2 .  
Command ==> Scroll ==> CSR .

Command - Enter "/" to select action	Message	Volume
-----		
SYS1.VTOCIX. <b>ITD701</b>		<b>ATD901</b>
TEAM12. <b>D701</b> .TEXT		<b>ATD901</b>

\*\*\*\*\* End of Data Set list \*\*\*\*\*

**Important:** As shown in Figure 2-18 on page 30, a full volume FlashCopy from device **D700** to **D701** also creates a backward cascade FlashCopy relationship from **D700** to **D701** when **D701** already has a relationship with **D901**.

## 2.5 Remote Pair FlashCopy

The use case described in this section is common in Metro Mirror configurations, such as shown in Figure 2-19 on page 37.

Without support for Cascading FlashCopy, an existing FlashCopy relationship at the secondary site prohibited a FlashCopy operation at the primary site. With Cascaded FlashCopy support, a data set level copy through DFSMSdss, the most common interface<sup>1</sup> in IBM Z environments, can now use FlashCopy. It runs very quickly, without the need to copy the data from source to target data set through the host channels. This does not require a change in existing DFSMSdss jobs if FlashCopy is already indicated in the DFSMSdss control statements.

<sup>1</sup> Other FlashCopy interfaces are TSO commands, ANTRQST Macro, ANTTREX program, or FlashCopy Manager, in addition to DSCLI, DSGUI, DFSMSdss, and Copy Services Manager.

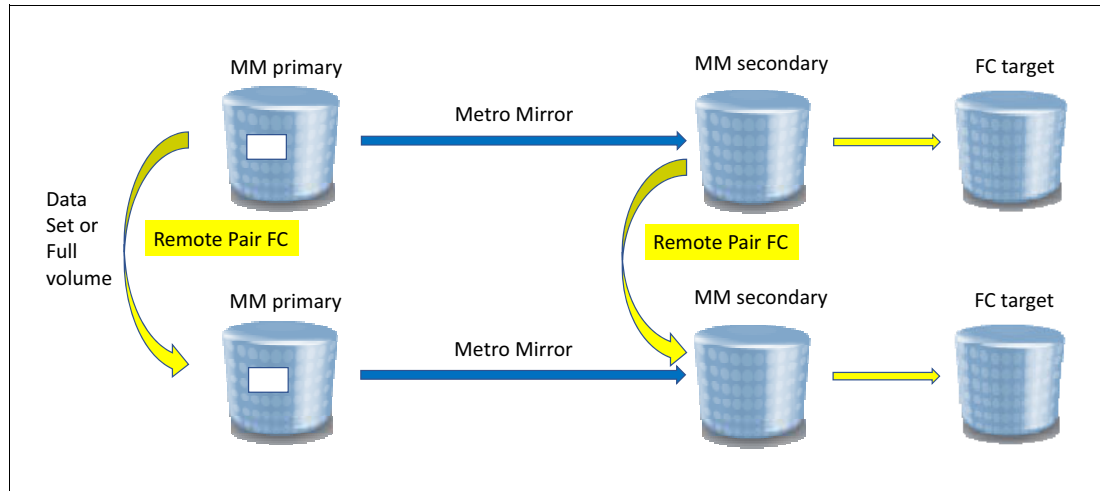


Figure 2-19 Remote Pair FlashCopy

Example 2-20 shows **COPY** command parameters to instruct DFSMSdss, which is the most common interface in IBM Z environments, to request FlashCopy processing for a target volume or data set in a Metro Mirror configuration.

Example 2-20 DFSMSdss COPY Command parameter

---

```
FR      (REQ) /* FlashCopy is required */ -
FCTOPPRCP (PMR) /* preserve Metro Mirror */ -
DEBUG    (FRMSG(DTL))
```

---

The commands have the following meanings:

**FR (REQ)** FlashCopy is required.  
**FCTOPPRCP (PMR)** FlashCopy target volume is a Metro Mirror primary volume.

Fast Replication Required, FR(REQ) can be set as the default through the DFSMSdss options installation exit routine, ADRUIXIT and the FCTOPPRCP(PMR) attribute. For more information, see *z/OS DFSMS Installation Exits Version 2 Release 3*, SC23-6850.

To receive more detailed information when DFSMSdss returns messages when a FlashCopy operation did not successfully complete, add the **DEBUG** parameter as shown in Example 2-20. This parameter requests detailed (**DTL**) Fast Replication messages from DFSMSdss.

## 2.6 Multiple FC relationships in a more complex configuration

Using a Metro Mirror configuration as a starting point, and then adding multiple target FlashCopy relationships and Cascading FlashCopy, offers new ways of managing backups and enables more comprehensive data-based investigations. Additional targets can also be used for creating isolated data copies for data safety in a world of ransomware attacks against application servers.

The use case discussed in this section relies again on maintaining existing FlashCopy relationships when reversing an existing FlashCopy relationship.

Figure 2-20 shows a more complex configuration than Figure 2-1 on page 14, and uses Fast Reverse Restore without impacting existing FlashCopy relationships. Of course, only a single FRR operation can happen at the same time to a FRR target volume (in Figure 2-20 to volume A).

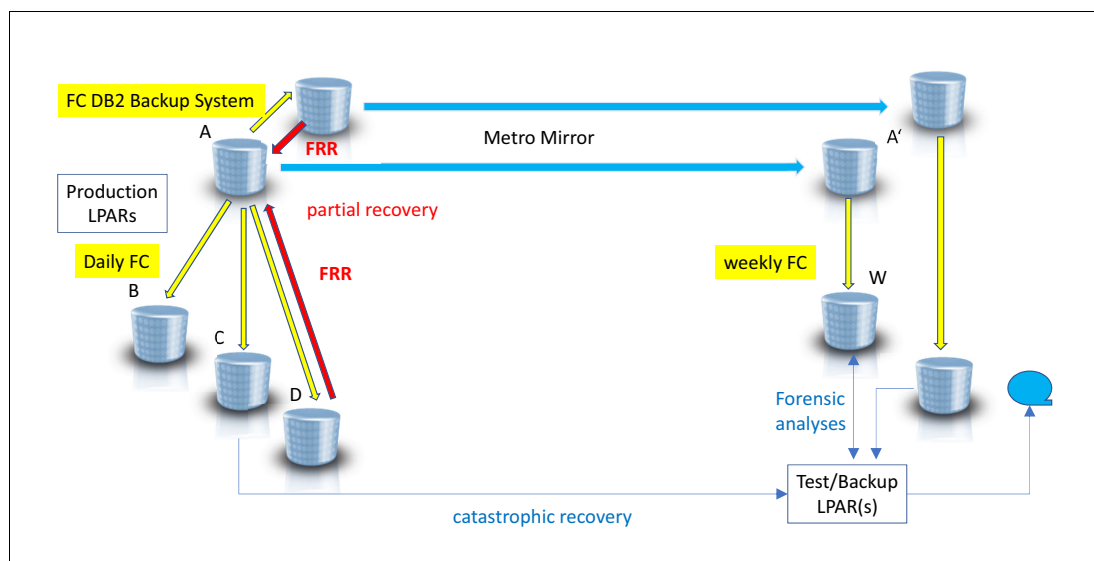


Figure 2-20 Maintaining existing FlashCopy relationships in a Metro Mirrored configuration

Here, FlashCopy provides data availability daily and weekly. To spread the FlashCopy load in a mirrored configuration, daily FlashCopies are run off the Metro Mirror primary volumes. Weekly FlashCopies are run off the Metro Mirror secondary volumes. To minimize excessive background I/O activity, the number of multiple FlashCopy target volumes is limited to three and rotated through these three FlashCopy target volumes during the week.

To minimize potential data loss in case of a catastrophic recovery, FlashCopy is used to create a set of volumes on every day of the week. These sets of FlashCopy volumes can be used to recover individual volumes or an entire FlashCopy volume set. This might happen to either recover data at the application site through Fast Reverse Restore (D to A), or to use a FlashCopy volume set to serve as the base to a catastrophic recovery using the backup and test LPARs.

The weekly FlashCopy volume set can be used to create backups onto tape via dump conditioned copies (A' via W to tape) as a last resort if all disk-based data is damaged or corrupted. They might also serve as a data repository (W) to analyze suspicious data, or to investigate the data for potential logical inconsistencies. This data processing is happening in the backup and test LPARs.

Figure 2-21 on page 39 shows a variation of the previous use case. Here it is assumed that data validation, or analysis is performed at the production site on D volumes on another system (test) and isolated from the production Sysplex.

D volumes might be no-UCB volumes to production, and A and B volumes might be no-UCB volume to test. Another construct could be with test as part of the production Sysplex but with the production volumes A and B either offline or even not defined to this test LPAR, and vice versa with D volumes offline or not defined in production LPARs.

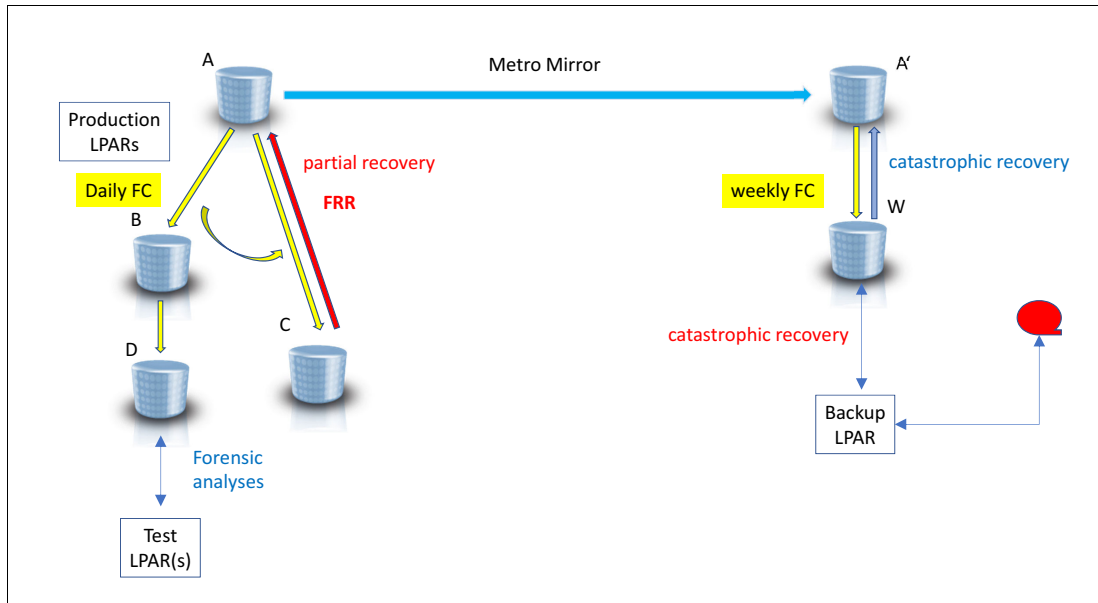


Figure 2-21 Multiple target and cascading FlashCopy to improve backup and data validation

In Figure 2-21, FlashCopy volumes, such as B and D, enable you to establish extra data copies for safety and security. For example, you can logically disconnect volume D and volume W from the production configuration (FlashCopy no-UCB) as a last data resort if production volumes got corrupted. Volume B/C might be used for data analysis and data validation, and for partial recovery in the production environment.

Those environments with D or W volumes, isolated from the production system, alleviate the risk of total data corruption.

Volume W might also be separated from the production environment (FlashCopy no-UCB), and is the source for backup copies to tape. Due to the capacity of the disk storage that needs to be backed up to tape, and a usually limited tape throughput, such a backup to tape might take more than 24 hours. This leads to a weekly based tape backup effort, which could also be dump conditioned with A' to W to tape in an IBM Z environment.

These use cases show that in more complex configurations the cascading FlashCopy capability opens the door to new approaches to achieve better data availability or explore new backup solutions.



# Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this paper.

## IBM Redbooks

The following IBM Redbooks publications provide additional information about the topic in this document. Note that some publications referenced in this list might be available in softcopy only.

- ▶ *IBM DS8880 Architecture and Implementation (Release 8.2.1)*, SG24-8323
- ▶ *DS8000 Copy Services*, SG24-8367

You can search for, view, download or order these documents and other Redbooks, Redpapers, Web Docs, draft and additional materials, at the following website:

[ibm.com/redbooks](http://ibm.com/redbooks)

## Other publications

These publications are also relevant as further information sources:

- ▶ *DS8000 Command-Line Interface User's Guide*, SC27-8526-00
- ▶ *DS8880 Introduction and Planning Guide*, GC27-8525

## Online resources

These websites are also relevant as further information sources:

- ▶ IBM DS8000 Knowledge Center:  
[https://www.ibm.com/support/knowledgecenter/ST56LJ\\_8.3.0/com.ibm.storage.ssic.help.doc/f2c\\_ichomepage\\_v8.30.html](https://www.ibm.com/support/knowledgecenter/ST56LJ_8.3.0/com.ibm.storage.ssic.help.doc/f2c_ichomepage_v8.30.html)
- ▶ IBM Techdocs Library - The IBM Technical Sales Library:  
<http://www.ibm.com/support/techdocs/atmastr.nsf/Web/Techdocs>

## Help from IBM

IBM Support and downloads

[ibm.com/support](http://ibm.com/support)

IBM Global Services

[ibm.com/services](http://ibm.com/services)







REDP-5463-00

ISBN 0738456438

Printed in U.S.A.

Get connected

