Chapter Table of Contents

Chapter 4.2

Restore and Restart Techniques

Aim

To equip the students with restore and restart techniques

Instructional Objectives

After completing this chapter, you should be able to:

- Outline the considerations to restore and restart of replica
- Explain local replication methodologies
- Describe the process of creating multiple replicas
- Elaborate various methods of managing an interface

Learning Outcomes

At the end of this chapter, you are expected to:

- Discuss the key considerations to restart and restore replica
- Explain the factors affecting local replication technologies
- Illustrate creating multiple replicas
- Compare CLI and GUI

4.2.1 Introduction

An efficient data protection strategy considers all the aspects required in protecting the data and apply them to different classes of data, as necessary. Replication of data is one way of supporting the data integration requirements. It provides trusted data synchronisation as well as provides capabilities to capture the change in data. It ensures the availability of data that enables the businesses to manage the data growth efficiently, while increasing the revenue by utilising up-to-the-minute information.

In the previous chapter, we understood the concept of local replication and discussed the technologies associated with it. We are now aware of the types of replication techniques, host-based and storage array-based replication as discussed in the previous chapter.

This chapter deals with the restore and restart techniques of the replica and focuses primarily on when to consider restoring and restarting a replica. We will also discuss the benefits of multiple replicas and will learn to create multiple replicas on multiple servers in the coming topics. Apart from that, we will also elaborate on the different methods of managing an interface and understand the differences between them.

Let us begin with understanding when to consider performing the restore from the replica and restart the applications.

4.2.2 Restore and Restart Considerations

In organisations, on a regular basis, there are numerous replication requests. Although these requests might not be constant. These requests emerge out as a result of:

- Need for data relocation for the purpose of hardware upgrade.
- Need for data replication for the purpose of availability and disaster recovery.
- Need for data duplication for the purpose of backup.
- Need for data relocation for performance reasons.
- Need for data duplication for the purpose of testing.

Local replicas can be used to restore the data back to the production devices. Similarly, it can also be used to restart the applications using the consistent point-in-time copy of the data available on the replicas.

In a situation of logical corruption of production devices, which means the production devices are fine but the data contained by them got corrupted and is now invalid and unavailable, a local replica can be used to restore the data. In situations, such as accidental deletion of data like table or entries in a database, incorrect data entry and incorrect update to existing information, the local replicas can be considered for restoring the data.

Performing restore operations from a local replica is an **incremental process** and, sometimes, require a very small RTO. In an incremental restore process, data is restored regularly, usually in a small amount. Usually, only the data that has changed since the last backup is restored. Such kind of restoration technique provides certain benefits, such as:

- It minimises the time required for daily backups.
- It reduces the network bandwidth usage by restoring small amount of data and not full data at a time.

In some cases, the applications may be resumed by the production devices to be used before the restoration of data has actually completed. However, it is a good practice to disable the access to production devices and replica devices before the restore operations begin to avoid any discrepancy in the actual data and its copy.

There are times, when the production devices can also become unavailable due to physical failures. These physical failures can be a production server or a physical device failure. In such cases, the applications can be restarted with the help of the data present on the latest replica. After the production server failure issue has been resolved, the latest information on the production devices can be restored back to the production devices.

It is interesting to know that while the production devices are failed and are in the mode of being recovered, the applications can continue to run on the replica devices. Even if the production devices fail to recover, the replica devices can be used to run the applications. A new point-in-time copy of the replica devices can be created, or you can fetch the latest information from the replica devices and restore it on a new set of production devices.

It is a good practice to stop all access to the replica devices, prior to restarting the applications using the replica devices. It saves us from any sort of data discrepancy. Also, as a precaution against further failures, a Gold Copy of the replica devices must be created.

A Gold Copy is an another copy of the replica device created in order to preserve a copy of data in the unfortunate event of failure or corruption of the replica devices itself. It's more like a

Plan B for the replica devices. But you should create only one copy of the Gold Copy and not more. Because with a Gold Copy in place, you have already lowered the risk of data loss to three levels. First, the data can be accessed from the production devices, in case it fails, then the data can be restored from the replica devices. And lastly, even if the replica devices fail, the data can still be restored from the Gold Copy. A Gold Copy consumes considerable amount of space. So, if the volume of data is very high, the storage size can increase proportionally and your storage costs can shoot.

Let us look at the following points that very well sums up the question, "When to consider restore and restart?":

- In case of a logical corruption of a production device, restore from local replicas can be made.
- In case of a physical failure of a production device, when the physical device can be recovered, restore from the local replicas can be made onto the same physical device.
- In case of a complete physical failure, when the production device cannot be recovered, restore from the local replicas can be made to a new physical device.
- In case of a logical corruption of the production devices, applications can be restarted for being used even before the data restore has completed.
- In case of a physical corruption of the production devices, applications can be restarted using the replica devices.

When performing a restore, in case of a physical failure of the production devices, full-volume replicas (both full-volume mirrors and pointer-based in Full Copy mode) can be easily restored to the original production devices or onto a new set of production devices. When performing the restore to the original production devices, the restore is incremental. However, when the restore is to be made to a new set of production devices, it is a full-volume copy operation.

In pointer-based virtual mode and pointer-based full volume in Copy on First Access (CoFA) mode, the access to data available on the replica depends on the health and accessibility of the original production devices. In case, if the original production devices are corrupted and cannot be reused again, these replicas can also be not used for any restore or restart purpose.

You have already read about all these modes of replication in Module 4, Chapter 1. Let us quickly recall the basics of each before we go ahead.

- **Full-volume mirroring:** In this technique, the target gets attached to the source and establishes itself as a mirror of the source. Existing data as well as new updates on the source gets copied to the target. After the copying of data is complete, the source and the target becomes identical; the target is then considered as the mirror of the source data. During this process, the target remains unavailable to be accessed by any other host; however, both the target and source can be accessed by the production host. After the synchronisation process gets complete, the target gets detached from the source and is made available for the business continuity operations.

- **Pointer-based full volume in Full Copy mode:** In pointer-based full volume in Full copy mode, the copying of data from the source to target takes place in the background and the data gets copied regardless of access. In case, when access to a block that has not yet been copied is required, that block is preferentially copied to the target. In case, the replication session gets terminated in between, the target contains all the original data from the source at the point-in-time of activation. This makes the target a viable copy for restoration, recovery, or any other business continuity operations.

- **Pointer-based full volume in CoFA mode:** In the CoFA mode, when the replication operation is initiated, the source data gets copied to the target at the trigger of the following:

    - **When a write operation gets issued to a specific address on the source for the first time:** In this case, the original data at that address gets copied to the target and then the source is updated with new data. This is done to ensure that the original data at the point-in-time of activation has been preserved on the target.

    - **When a read or write operation gets issued to a specific address on the target for the first time:** In this case, the original data gets copied from the source to target. However, in case of the read operation, after the copying of data it is made available to the host. In case of the write operation, after the copying of data, new data is updated on the target.

- **Pointer-based virtual mode**: In this technique, when the backup session is activated, the target contains the pointers to the location of data on the source. At any time, the target does not contain the actual data but only contains the pointer to the data on the source. Hence, the target is also known as a virtual replica. Similar to a pointer-based full volume replication, a protection bitmap is created for all the data available on the

source device and the target is immediately accessible, unlike full volume mirroring. Similar to a file system snapshot, a pointer-based virtual replication also uses the CoFA technology.

Let us refer to the Table 4.2.1 that presents a comparative analysis of these storage array-based local replication technologies.

Responsible Factors	Full-Volume Mirroring	Pointer-Based Full Volume Replication	Pointer-Based Virtual Replication
Performance impact on the production source	No impact	CoFA Mode: Little impact Full Copy Mode: No impact	High impact
Target size	Same or bigger than source	Same or bigger than source	Small fraction of the source
Accessibility of source while restoration	Not required	CoFA Mode: Required Full Copy Mode: Not required	Highly required
Accessibility to target	Only after synchronisation and detachment from the source	Immediately accessible	Immediately accessible

Table 4.2.1: Comparing Local Replication Techniques

(i) Tracking Changes to Source and Target

Updates are something that one has to accept and just deal with; it cannot be stopped. After the point-in-time local replicas are created, updates continue to take place on the source device. The point-in-time replicas can either be saved from making any change or can be changed depending on the requirement. If the main purpose of creating a local replica is to have a point-in-time copy of the data that can be used for data recovery and restore operations, then the target devices should be refrained from being modified. If the replicas are not used for any business continuity operations, then changes may be made to the target devices.

Changes to both the source and target devices after the creation of point-in-time copy can be tracked to enable the incremental resynchronisation or restore operations. The tracking is

usually done with the help of **bitmaps**. A bitmap is a kind of memory organisation or an image file format that is used to store digital images. The data stored as bitmap can be generated with the help of a computer and can be stored in strings of 0s and 1s. Data in this form is known as digital data.

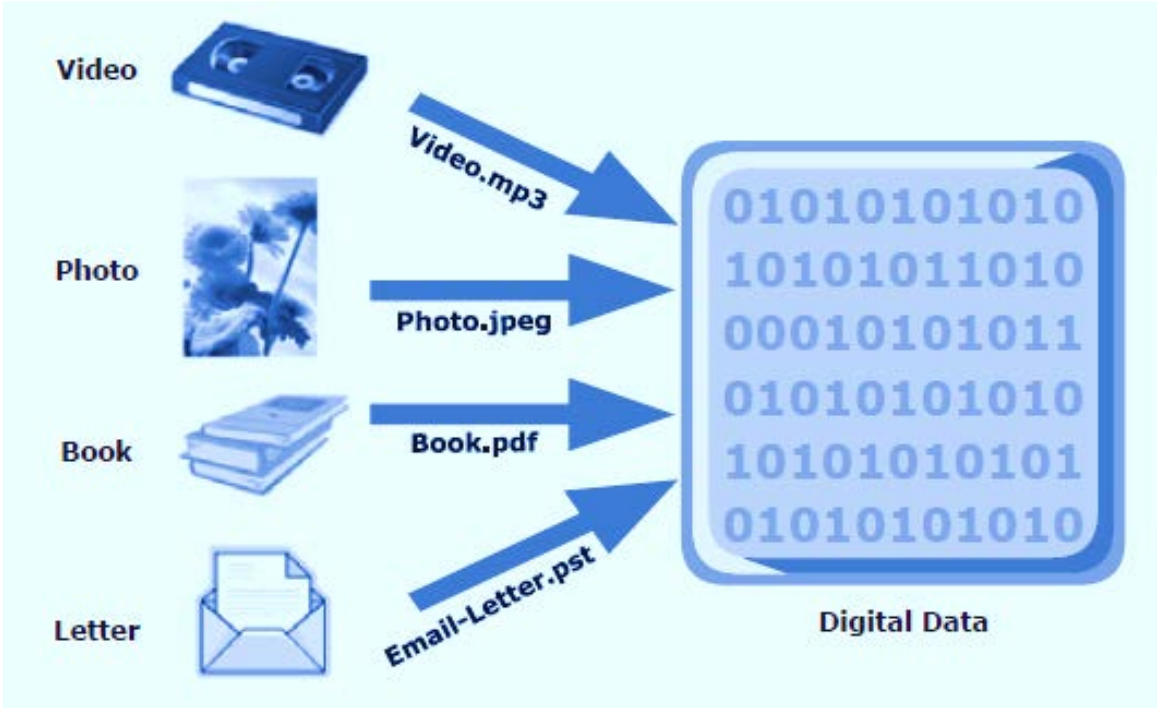

Figure 4.2.1: Digital Data Stored as Bitmap

A bitmap defines a display space and consists of a digital image made up of matrix of dots known as pixels. The term bitmap was derived from the computer programming technology, which means just a map of bits.

The tracking of changes in target and source data is done with the help of bitmaps, with one bit per block of data. The block size can range from 512 bytes to 64 Kilobits (KB) or greater. ***For example***, if the block size is 32 KB, then a 1 Gigabits (GB) device would need 32,768 bits. Not to forget that 1 GB = 1048576 KB. The size of the bitmap would be 4 KB.

The bitmap is used to keep a track of the blocks that got changed in the source device after the creation of a point-in-time copy. Considering the previous example where the block size was 32 KB, if there occurs a change in any or all of the 32 KB block, the corresponding bit in the bitmap is flagged. If the block size is reduced for tracking purposes, then the size of the bitmap increases correspondingly.

When a point-in-time copy is created, the bits in the source and target bitmaps are all set to 0 (zero). When there happens a change in either the source device or in the target device, then the changes are flagged and the appropriate bits are set to 1 in the bitmap. Hence, the bitmap flags the changes in the affected bits by setting the changed bits from 0 to 1.

When a resynchronisation or a restore is required, a **logical OR** operation is performed between the source bitmap and the target bitmap. A logical OR operation returns the Boolean value True if either one or both the operands are true, else it returns a False value.

The bitmaps that result from the logical OR operation references all the blocks that were changed or modified in either the source device or the target device. This results in eliminating the need of copying all the blocks between the source device and the target device and enables an optimised resynchronisation or a restore operation.

Let us refer to the Figure 4.2.2 that depicts bitmaps before and after the change and the impact of the logical OR operation in enabling an optimised resynchronisation or restore process. In this figure, you can see the bitmap created at the time of creating the point-in-time copy. You can see that all the blocks are marked as 0, which depicts that the data in the source and target devices are same and unchanged. After there has been some changes made into both the source and target devices, the changed blocks are marked as 1. In this case, blocks 1, 4 and 6 (read from left) are marked 1 in the source device and blocks 3, 4 and 8 (read from left) are marked 1 in the target device. When the logical OR operation is performed at the time of a resynchronisation or restore operation, you can see that all the changed blocks from target and the source device are highlighted and marked as 1. In this case, blocks 1, 3, 4, 6 and 8 (read from left) are highlighted by being marked as 1. This highlighting eliminated the need for copying all the blocks all over again and focuses only on the changed blocks.

Did You Know?

The numbering of blocks starts from 0 and the block with address 0 is read as block number 1.

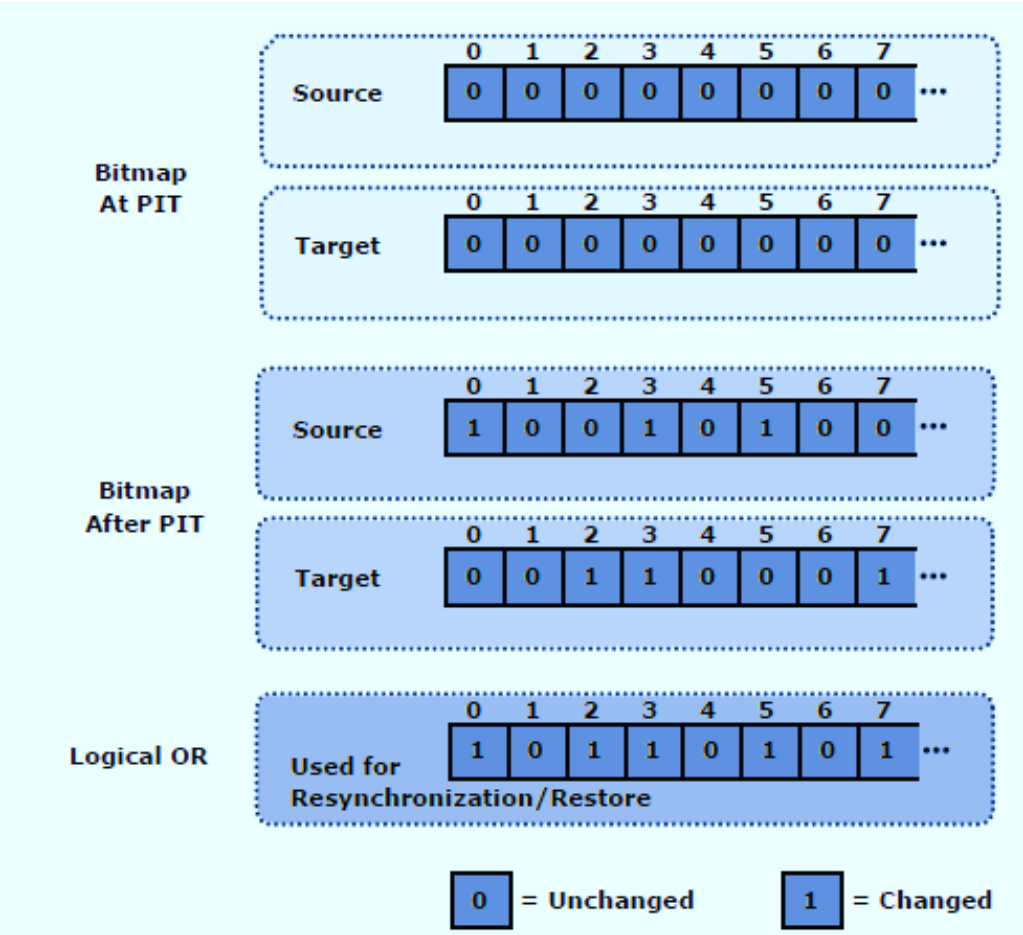

Figure 4.2.2: Tracking Changes

After the changed blocks are identified, the data in the source and target devices needs to be resynced to eliminate the differences. The data needs to be moved from one device to the other to resolve the difference. The direction of the movement of data completely depends on the kind of operation that is being performed, such as restore or resynchronisation. In a restore operation, the movement of data would take place from the target device to the source device. And in the case of a resynchronisation operation, the movement of data would occur from the source device to the target device.

In the resynchronisation operation, the changes on the target device gets overwritten with the corresponding blocks from the source device. Refer to Figure 4.2.2, in this example, blocks 3, 4 and 8 (read from left) in the target device will get overwritten by the corresponding blocks in the source device when a resynchronisation operation will occur.

In the restore operation, the changes on the source device are overwritten with the corresponding blocks from the target device. Refer to Figure 4.2.2, in this example, blocks 1, 4

and 6 (read from left) in the source device will get overwritten by the corresponding blocks on the target device when a restore operation will take place.

In either case, changes to both the source device and the target device cannot be simultaneously preserved.

Self-assessment Questions

1) Which of the following situations can be taken into account as considerations for performing restore and restart operations?
    a) In case of a logical corruption of a production device, restore from local replicas can be made.
    b) In case of a physical failure of a production device, when the physical device can be recovered, restore from the local replicas can be made on to the same physical device.
    c) In case of a complete physical failure, when the physical device cannot be recovered, restore from the local replicas can be made to a new physical device.
    d) In case of a logical corruption of the production devices, applications can be restarted for being used even before the data restore has completed.

2) In case of a restore made to the original production device after a physical failure, what kind of restore is performed?
    a) Full-volume copy
    b) Incremental
    c) Gold Copy
    d) Bitmap

3) When the original production devices are corrupted and cannot be reused again, which of the following replicas also lose their meaning and cannot be used for any restore or restart purpose?
    a) Pointer-based full volume in CoFA mode
    b) Full-volume mirroring
    c) Pointer-based virtual node
    d) Gold Copy

4) Which of the following helps to track changes in the source and target devices?
    a) Gold Copy
    b) Full-volume mirroring
    c) Pointer-based virtual node
    d) Bitmap

4.2.3 Creating Multiple Replicas

Apart from serving as the restore and resynchronisation tools, replicas are also created in order to make the data available to users in different locations, different networks, or in different time zones.

File system snapshots is a technique that enables the creation of multiple logical point-in-time copies of a production volume, eliminating the creation of physical copies of the production data. You have already learnt about file system snapshot in Module 4, Chapter 1. Let us quickly recall the definition. A file system snapshot is a pointer-based replica that only requires a fraction of the space consumed by the original file system. When you create a snapshot, then you actually create a new logical volume that acts as a clone of the original logical volume. The snapshot initially does not consume any space; however, with time, as the changes are made to the original logical volume, the changed blocks, before getting changed, gets copied into the snapshot for being preserved. This means, that the more changes are made to the original volume, the more space will be required by the snapshot.

Most storage array-based replication technologies, such as full-volume mirroring, pointer-based full volume replication and pointer-based virtual replication, enable the source devices to maintain a replication relationship with more than one target devices. Each point-in-time copy can be utilised for different business continuity activities or can be used at the time of restore and resynchronisation operations.

The more the number of replicas, the more are the chances of changes that can slowly creep in between the replica and the source devices. However, changes made to the source device and each of the related target devices can be tracked using bitmaps, where bitmaps use 0s and 1s to highlight the changed and unchanged blocks. This enables incremental resynchronisation of all the target devices.

Let us refer to Figure 4.2.3 where multiple replicas are created from the same source after every six hours.

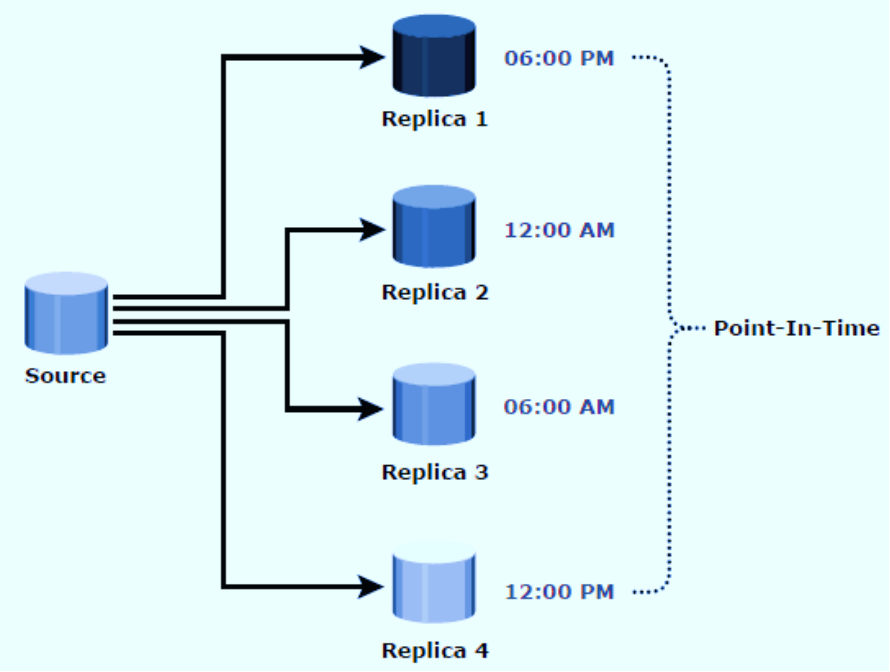

Figure 4.2.3: Creating Multiple Replicas

In this case, if the source data is corrupted, then to perform the restore operation, the data from the latest point-in-time copy should be used. The maximum Recovery Point Objective (RPO) in this case would be 6 hours.

You have already read about RPO in Module 3, Chapter 1. Let us quickly recall the definition. RPO along with Recovery Time Objective (RTO) are one of the most important parameters of disaster recovery. RPO defines the point in time until when the business process's recovery can tolerably proceed, keeping in mind the volume of data lost in that interval. RTO is the amount of time and a service level taken by a business, within which the business should be restored to avoid unacceptable consequences associated with a break in continuity.

In order to reduce the RPO and RTO, more frequent replicas should be created while creating multiple replicas.

Storage array-based local replication technologies also enable the creation of multiple concurrent point-in-time replicas. In this case, the copying of data from the same source to multiple targets will take place at the same time and hence, all the replicas will contain identical

data. Some of the replicas created can be used for restore or recovery operations and some of them can be utilised for performing decision-support activities.

You have already read about decision-support activities in Module 4, Chapter 1. Let us quickly recall the definition. Local replication also provides decision-support activities, such as reporting. You can use the local replicas to create a report that shows the input/output (I/O) workload on the production volume. These reports can be utilised for further analysis and decision-making to reduce the burden on the production volume. Hence, the local replicas can be utilised for minimising the burden on the production volumes.

Need for Creating Multiple Replicas

You might wonder what is the need for creating multiple replicas from the same source when a single replica can also serve the purpose of restore. Let us understand it this way, when a source device creates multiple replicas, it is done so in order to fulfil some business continuity operations as well other than using the replicas for restore operations. Some of the examples of such business continuity operations would be:

- Making a copy of data available for users in the remote location.
- Providing a replica to a specific workgroup that contains only a subset of information relevant to that workgroup.
- Making a copy of data available for users in different time zones.
- Creating a copy of data to perform data health analysis.
- Creating a copy of data to perform other decision-support activities.

Performing all these activities is just not possible with having only one copy of the source data that is dedicated for restore operations and hence, at times, organisations tend to create multiple copies of the same data.

Creating Replicas on Multiple Servers

Let us follow the steps that allows us to create multiple replicas of the same source device on multiple servers.

Step 1: Go to the server location on your computer where the databases are saved.

Step 2: Select the **Files** tab.

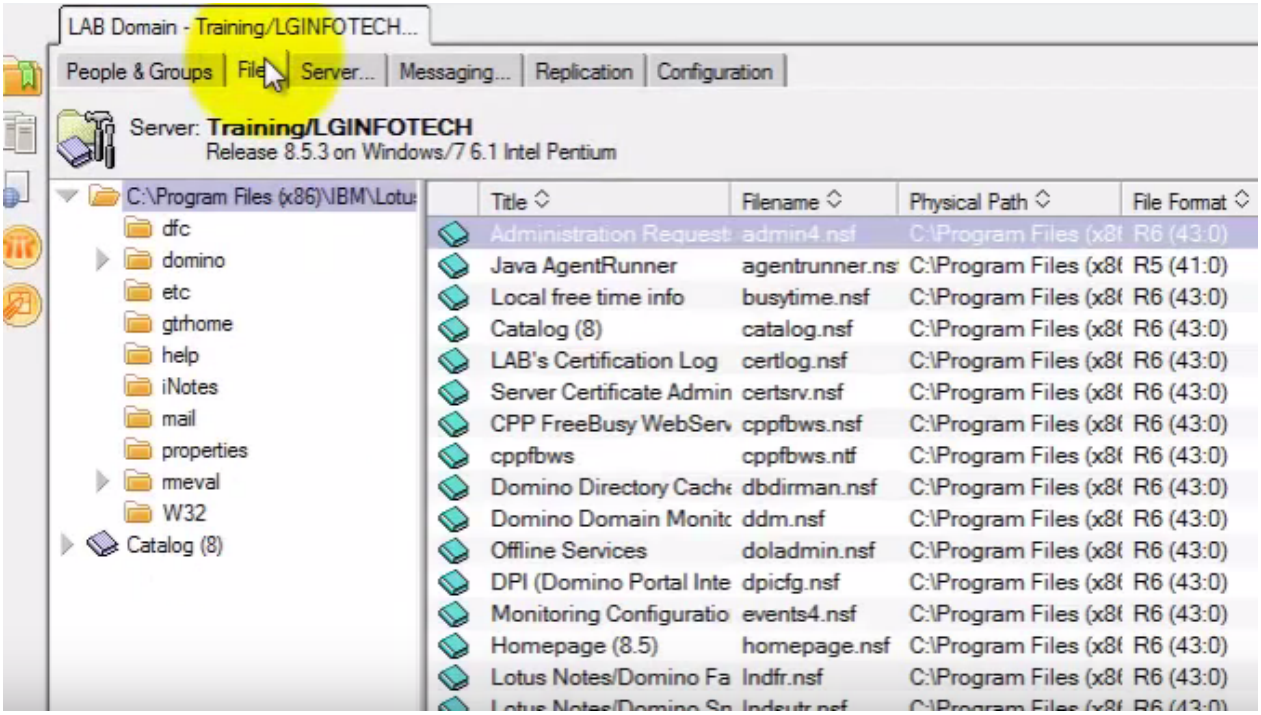

Figure 4.2.4: Selecting the Files Tab

Step 3: Choose the database that needs to be replicated.

Step 4: Scroll the window to the right to see the list of available options.

Step 5: Select **Create Replica(s)** from the available options to open the Create Replica window.

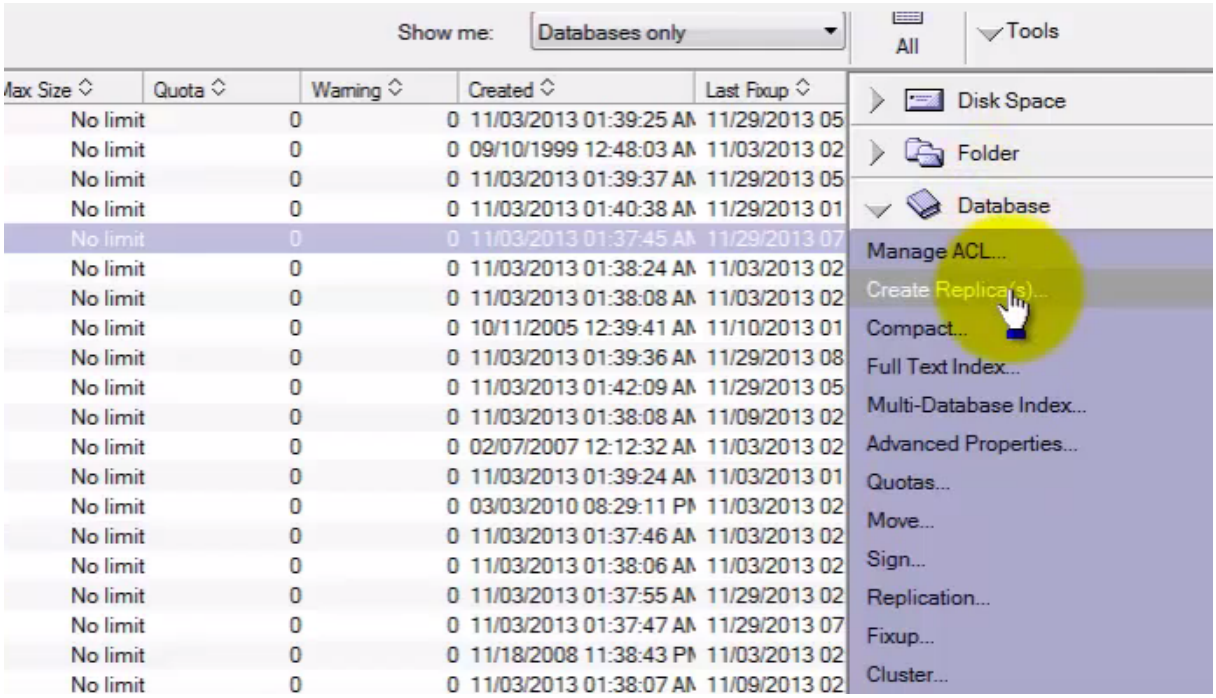

Figure 4.2.5: Creating Replicas

Step 6: On the Create Replica window, under Create replicas on these servers box, select the names of the server on which the replica needs to be created.

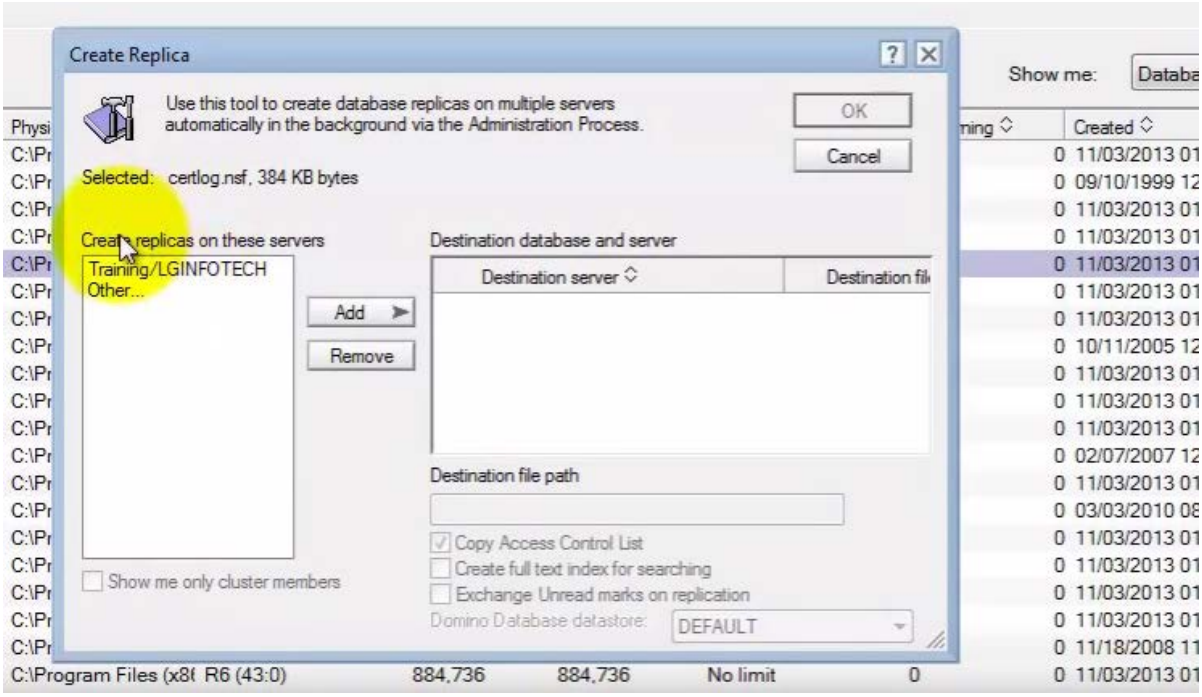

Figure 4.2.6: Create Replicas on These Serves Box

Step 7: Select the names of the server and click **Add** to add the name of the server in the Destination database and server box.

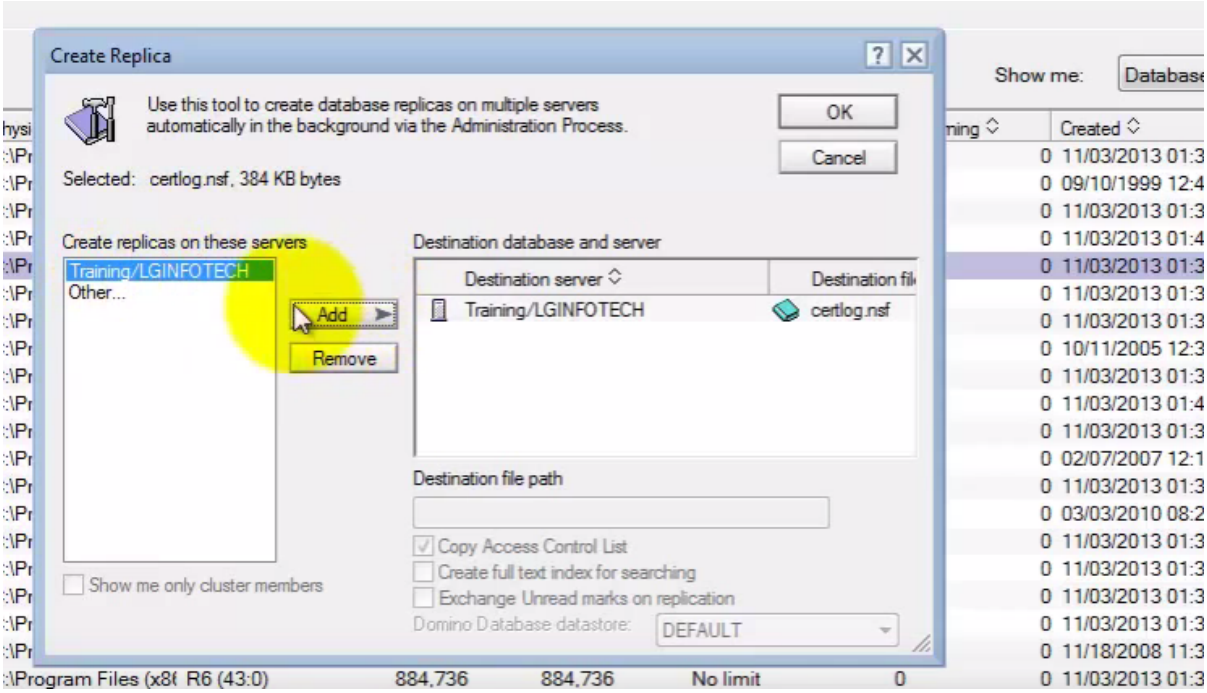

Figure 4.2.7: Adding Servers to Destination Servers List

Step 8: One by one, you can add the list of servers from the Create replicas on these servers box to the Destination database and server box.

Step 9: To add a desired database that is not available in the Create replicas on these servers box, select **Other** from the Create replicas on these servers box and click **Add**. The Add a Server window opens.

Did You Know?

If you have accidently added a server name to the destination server list, you can always delete it by selecting the server name and clicking **Remove**.

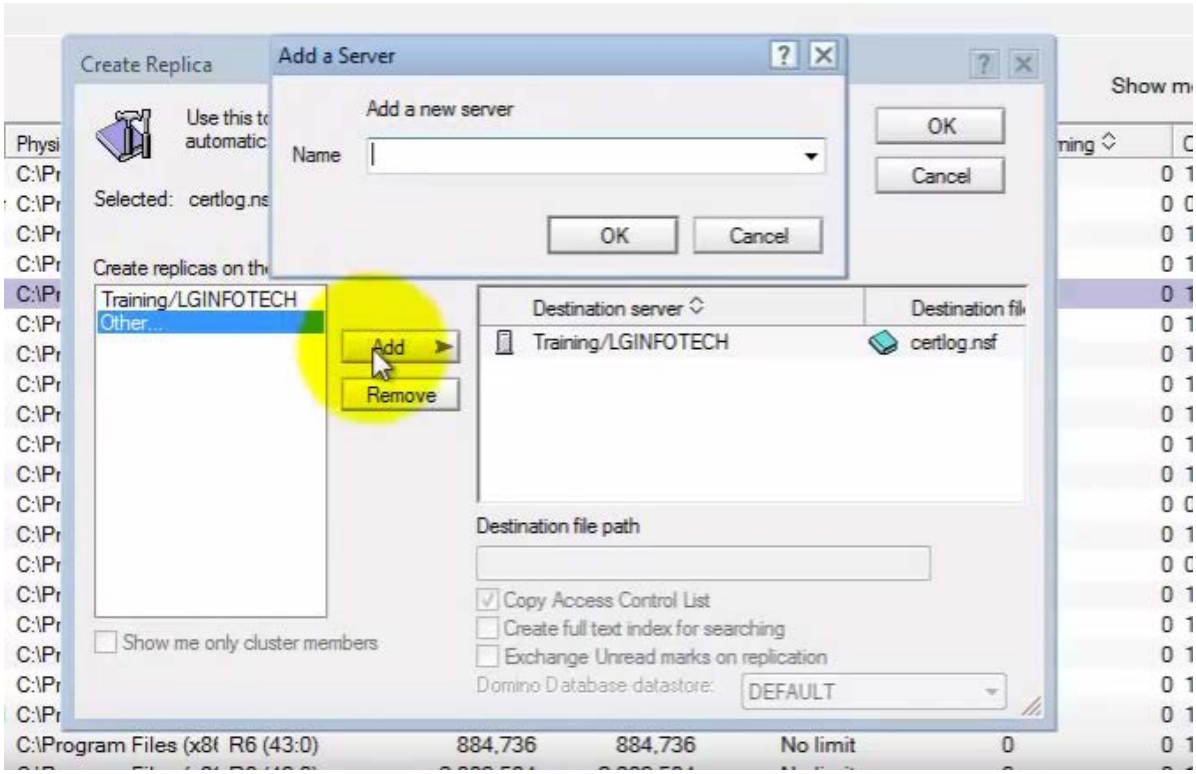

Figure 4.2.8: Adding a New Server

Step 10: Type the name of the desired server in the Name textbox and click **OK**. The server name gets added to the Destination database and server box.

Step 11: After the destination servers are selected, click **OK**. Multiple replicas of the selected database will be created on the selected destination servers.

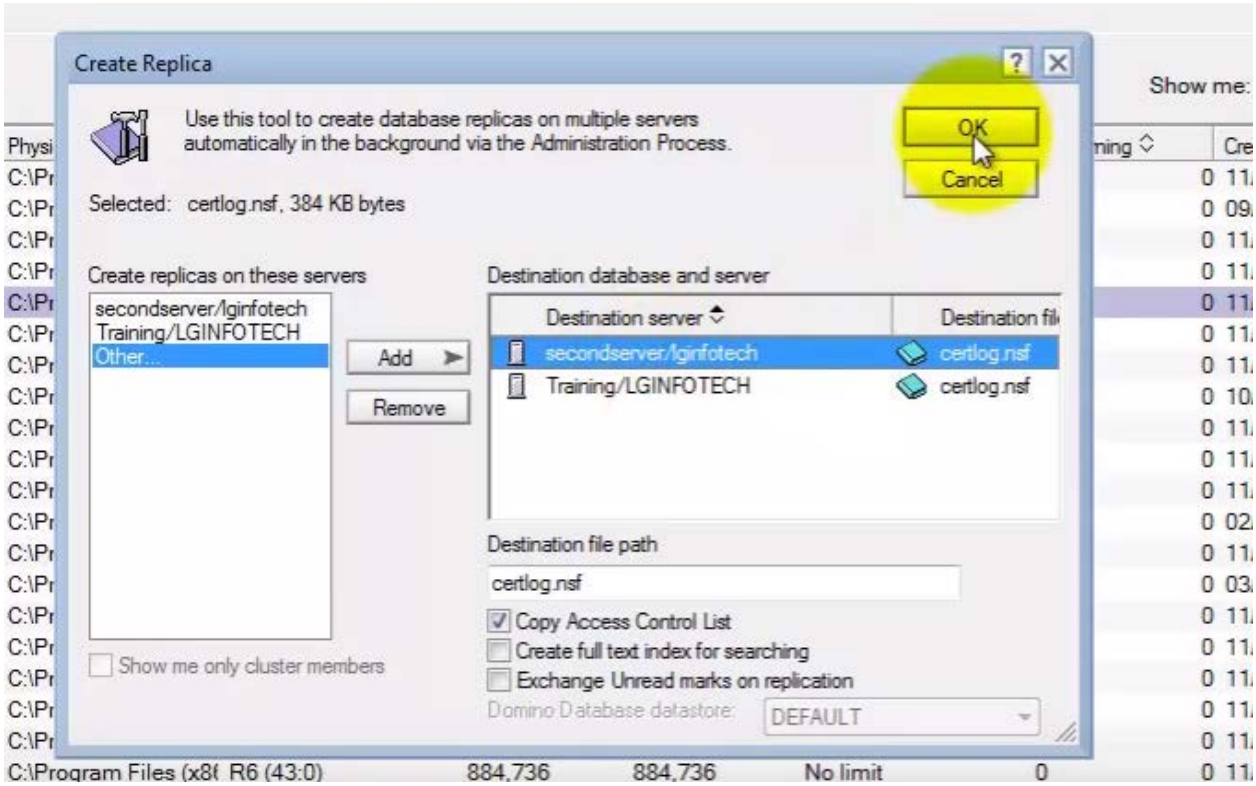

Figure 4.2.9: Selecting OK to Create Multiple Replicas

Self-assessment Questions

5) Which of the following techniques enables the creation of multiple logical point-in-time copies of source, eliminating the need of creating physical copies?

a) Full volume mirroring
b) Gold Copy
c) File system snapshot
d) Bitmap

6) Multiple replicas are being created from the same source after every 12 hours. In order to reduce the RPO and RTO, what should be the best approach?

a) Increase the replication time from a difference of 12 hours to 15 hours.
b) Decrease the replication time from a difference of 12 hours to 4 hours.
c) Increase the replication time from a difference of 12 hours to 24 hours.
d) The time duration is fine and the RPO and RTO cannot be reduced.

7) Which of the following are true, when concurrent point-in-time replicas are created from the same server?

a) RPO and RTO is reduced with the time difference in creating multiple replicas
b) All the replicas contain identical data
c) Replication takes place at same time
d) Replication takes place at different time intervals

8) How do you add a new server to the destination server that does not appear in the Create replicas on these servers list?

a) Select Create Replicas from the Files tab
b) Select Other and click Add to add the server name
c) Go to the Files tab and select the server
d) Select the server and right-click to add

4.2.4 Management Interface

The replication management software present on the storage array provides an interface that promotes a smooth and error-free replication. This interface provides various options such as options for synchronisation, resynchronisation, splitting, starting and stopping a replication session and monitoring the performance of replication. In general, the replication management software provides two types of interface:

- Command line interface
- Graphical user interface

(i) Command Line Interface

A command line interface (CLI) is a text-based user interface to a computer's operating system (OS) or an application. It is used to operate software and OS by allowing the user to respond to visual prompts by typing commands into the interface. The system also responds back in the similar way in the form of commands.

In a CLI, an administrator can directly enter commands against a prompt. A user can also perform some command operations based on the assigned privileges. The CLI scripts are developed in order to perform certain business continuity operations in a database or in an application environment.

CLI is an older method of interacting with the OS and the applications and is used to perform specific tasks required by the users. Its working mechanism is easy; however, it is not very user friendly. In a CLI, user enters the specific command, presses Enter and then awaits the response from the system. After receiving the command from the user, CLI processes it accordingly and displays the result on the same screen. A command line interpreter is required for this purpose.

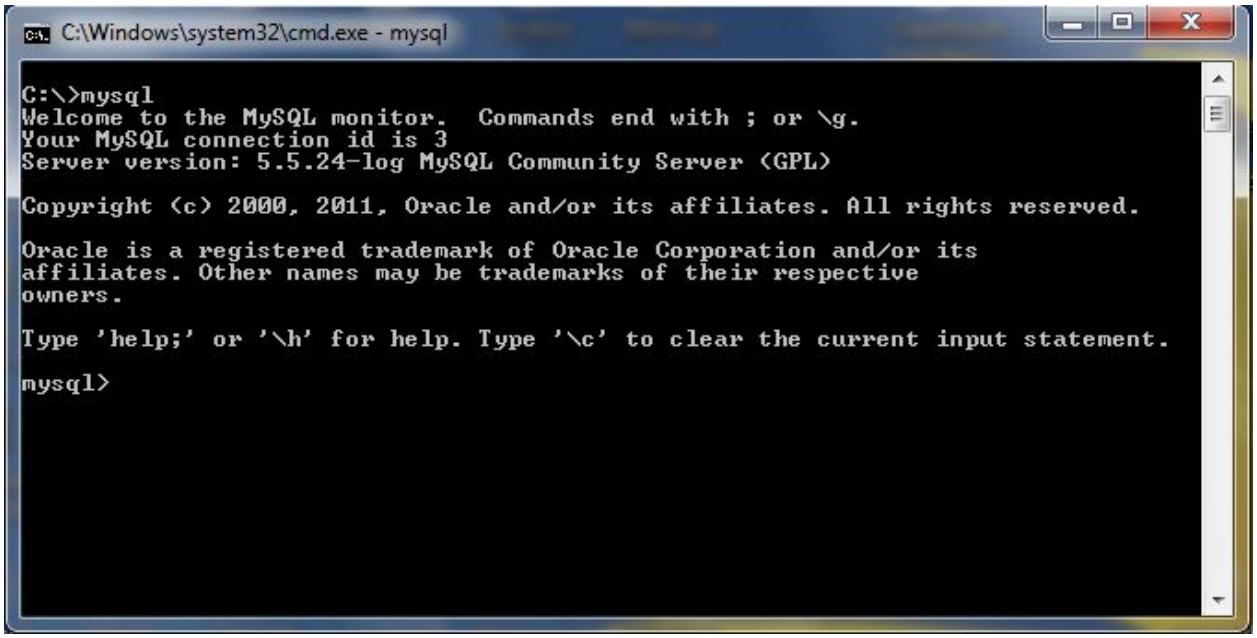

Figure 4.2.10: CLI

(ii) Graphical User Interface

A graphical user interface (GUI), pronounced gooey, is an interface that takes advantage of the computer graphic capabilities to make the application user friendly. The GUI interface consists of toolbar and menu bar options and includes other graphical elements, such as windows, buttons and icons. A well designed GUI saves the user from learning complex command languages.

This type of interface allows the user to select an operation name and monitor its performance in real time. Most of the tools today have a browser-based GUI that enables the transferring of commands and are often integrated with several storage management suite products.

Figure 4.2.11: GUI

Let us refer to Table 4.2.2 that provides a comparison between CLI and GUI

CLI	GUI
Test-based user interface	Graphic-based user interface
Requires a command prompt or terminal or console window to type commands	Commands can be given by as easy as through a mouse click
Constitutes of command line	Constitutes of icons, buttons and windows
Less user friendly as a lot of complex commands are to be memorised in order to use CLI	More user friendly as learning command languages are replaced by interacting with graphics
Difficult to learn	Can be learnt easily

CLI commands are very less frequently changed and most of the basic commands are being used since ages	GUI software tend to upgrade to provide a neater version to the user
CLI commands are powerful and robust and they fail barely.	GUI commands requires a lot of RAM and supporting background resources, hence, there are more chances for their failure.
Some complex commands can only be executed by the CLI, such as checking CPU temperature and sending it as an email message.	Such commands cannot be performed by GUI.
In a replication environment, CLI scripts are generated to perform certain business continuity operations in a database or application environment.	In a replication environment, GUI are used for selecting an operation and monitoring its real time performance.

Table 4.2.2: Comparing CLI and GUI

Both CLI and GUI are popular amongst the users depending on their needs. Almost all the activities on a computer can be performed by both CLI and GUI, such as:

- Downloading from a direct link
- Extracting files easily
- Installing and upgrading software
- Starting and stopping services
- Killing processes
- Opening a new browser, refreshing the browser
- You can perform an activity to open the calculator application present in your computer using both GUI and CLI and share your experience with others.

Self-assessment Questions

9) Which of the two interfaces are provided by the replication management software?

a) CLI
b) GUI
c) RPO
d) RTO

10) Which interface allows the user to select an operation name and monitor its performance in real time?

a) RPO
b) CLI
c) GUI
d) RTO

11) Which interface scripts are developed to perform certain business continuity operations in a database or in an application environment?

a) GUI
b) RPO
c) RTO
d) CLI

Summary

- Local replicas can be used to restore the data back to the production devices.
- Performing restore operations from a local replica is an incremental process and, sometimes, require a very small RTO.
- In an incremental restore process, data is restored regularly in a small amount and only the data that has changed since the last backup is restored.
- A Gold Copy is an another copy of the replica device created in order to preserve a copy of data in the unfortunate event of failure or corruption of the replica devices itself.
- When the original production devices get corrupted to be never used again, the pointer-based virtual mode and pointer-based full volume in CoFA mode replicas also become useless to be used for any restore or restart operation.
- Tracking of changes to source and target device is done with the help of bitmaps, with one bit per block of data.
- The bitmap uses logical OR operation and flags the changes in the affected bits by setting the changed bits from 0 to 1.
- In the resynchronisation operation, the changes on the target device gets overwritten with the corresponding blocks from the source device.
- In the restore operation, the changes on the source device are overwritten with the corresponding blocks from the target device.
- Most storage array-based replication technologies enable the source devices to maintain a replication relationship with more than one target devices.
- Each point-in-time copy can be utilised for different business continuity activities or can be used at the time of restore and resynchronisation operations.
- Storage array-based local replication technologies also enable the creation of multiple concurrent point-in-time replicas.
- Replication management software provides two types of interfaces, CLI and GUI.
- CLI works with command lines and GUI works with graphics.

Terminal Questions

1. Define the considerations pertaining to when to perform the restore and restart operations.
2. How is the tracking of change done from source to target? Explain with example.
3. Discuss the need for creating multiple replicas.
4. What are CLI and GUI? Compare with examples.

Answer Keys

Self-assessment Questions	
Question No.	Answer
1	a, b, c, d
2	b
3	a, c
4	d
5	c
6	b
7	b, c
8	b
9	a, b
10	c
11	d

Activity

Activity Type: Offline **Duration:** 30 Minutes

Description:

Prepare a presentation on the following two topics:

1. Considerations made while performing restore and restart operations
2. Tracking changes to source and target during backup

Case Study

A 300 GB database needs two local replicas for reporting and backup. There are constraints in provisioning full capacity for the replicas. It has been determined that the database has been configured on 15 disks and the daily rate of change in the database is approximately 25 percent. You need to configure two pointer-based replicas for the database.

a) Describe how much capacity you would allocate for these replicas?

b) For the same database, discuss the advantages of configuring full-volume mirroring if there are no constraints on capacity.

Bibliography

e-References

- *Replicas*. (2016). *Www-12.lotus.com*. Retrieved 21 July 2016, from http://www-12.lotus.com/ldd/doc/domino_notes/7.0/help7_admin.nsf/f4b82fbb75e942a6852566ac0037f284/340be284ee373ff68525706f0065b61d?OpenDocument

- *Considerations for Restoring the model and msdb Databases*. (2011). *Technet.microsoft.com*. Retrieved 21 July 2016, from https://technet.microsoft.com/en-us/library/ms190749(v=sql.105).aspx

- *Replication: the secret to working offline*. (2016). *Www-12.lotus.com*. Retrieved 21 July 2016, from http://www.12.lotus.com/ldd/doc/domino_notes/Rnext/help6_client.nsf/b3266a3c17f9bb7085256b870069c0a9/a8389ebe34ccfa8f85256c1c0037cb17?OpenDocument

Image Credits

- Figure 4.2.1: Source: Information Storage and Management: Storing, Managing and Protecting Digital Information
- Figure 4.2.2: Source: Information Storage and Management: Storing, Managing and Protecting Digital Information
- Figure 4.2.3: Source: Information Storage and Management: Storing, Managing and Protecting Digital Information
- Figure 4.2.4: Source: https://www.youtube.com/watch?v=JPzUsEeTN8Y
- Figure 4.2.5: Source: https://www.youtube.com/watch?v=JPzUsEeTN8Y
- Figure 4.2.6: Source: https://www.youtube.com/watch?v=JPzUsEeTN8Y
- Figure 4.2.7: Source: https://www.youtube.com/watch?v=JPzUsEeTN8Y
- Figure 4.2.8: Source: https://www.youtube.com/watch?v=JPzUsEeTN8Y
- Figure 4.2.9: Source: https://www.youtube.com/watch?v=JPzUsEeTN8Y
- Figure 4.2.10: Source: http://howtocode.pk/wp-content/uploads/2015/04/mysql-cmd-3.jpg

- Figure 4.2.11 : Source: http://previews.123rf.com/images/sak111/sak1111401/sak111140100134/25401441-hand-pressing-applications-graphical-user-interface-flat-icons-on-tablet-Stock-Photo.jpg

External Resources

- Somasundaram, G. & Shrivastava, A. (2009) *Information storage and management* - Storing, managing and protecting digital information. *Indianapolis, Ind.: Wiley Pub.*

- Dufrasne, B., Eriksson, R., Martinez, L., & Kalabza, W. (2014). *IBM XIV Storage System Architecture and Implementation* (9th ed.). International Business Machines Corporation.

Video Links

Topic	Link
Creating replicas on multiple servers	https://www.youtube.com/watch?v=JPzUsEeTN8Y
Recovery, restore and backup considerations for Microsoft Exchange	https://www.youtube.com/watch?v=OZCCq6GPJ-Q
GUI vs. CLI	https://www.youtube.com/watch?v=D-yHV26DvnY
Discussing CLI and GUI	https://www.youtube.com/watch?v=c7FrxMmELK8
Why do we require CLI	https://www.youtube.com/watch?v=IqaryB7SxJU

Notes: