CA Cloud Storage for System z

z/OS Administration Guide

Release 1.1.00

CA Technologies Product References

The CA Cloud Storage for System z guides refer to the following CA products and components:

- CA 1® Tape Management (CA 1)
- CA Allocate™ DASD Space and Placement (CA Allocate)
- CA Earl® (CA Earl)
- CA Chorus Software Manager™ (CA CSM)
- CA MIM™ Resource Sharing (CA MIM)
- CA Tape Encryption
- CA TLMS® Tape Management (CA TLMS)
- CA Vtape™ Virtual Tape System (CA Vtape)

Contact CA Technologies

Contact CA Support

For your convenience, CA Technologies provides one site where you can access the information that you need for your Home Office, Small Business, and Enterprise CA Technologies products. At http://ca.com/support, you can access the following resources:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

Providing Feedback About Product Documentation

If you have comments or questions about CA Technologies product documentation, you can send a message to techpubs@ca.com.

To provide feedback about CA Technologies product documentation, complete our short customer survey which is available on the CA Support website at http://ca.com/docs.

Contents

Chapter 1: About this Guide 9

Audience 9
Conventions 9

Chapter 2: Introduction 11

Limited-Use CA Vtape License Requirement 11
Multisystem Considerations 11

Chapter 3: Understanding the Basic Components 13

CA Vtape 13
Virtual Volumes 14
Virtual Volume Compression 15
Support for Cloud Storage Gateways 15
Tape Mount Interception 16
- Close Processing 17
- Sense Processing 17
- Multivolume Output Processing 17
- Multiple File Processing to Virtual Volumes 17
- Uncataloged Data Sets 18
- Temporary Data Sets 18
- Unit Affinity and GDG Base Referral Processing 18
- Reference Backward Processing (Refer Backs) 18
- Concatenation Processing 18
- Special Processing Conditions 19

Split Maintenance-Level Protection 19
System Logger 20
- Internal Logger 20
- External Logger 20

Volume Pool Definitions 20

Chapter 4: ISPF Interface 21

Start the ISPF Interface and Access the Main Menu 22
Scratch Pool Header Information 23
Using the Tape Device Status Display Panel 24
Using the Volume Pool Display Panel 25

Using the Virtual Volume Display Panel 26
Virtual Volume Flag Fields 29
Label Types 30
Using the Dataset and DataClass List Panels 31

Chapter 5: Reports 33

Using SVTSUTIL Batch Commands to Generate Reports 33
ANALYZE=COMPRESSION 33
LIST=MODULE 34
Use CA Earl with CA Vtape 35
Sample CA Earl Components 36
CA Earl Component Modifications 37
CA Vtape Supplied CA Earl Reports 37
ERPT0120 and ERPT0130 Virtual Mount Performance Analysis 38
ERPT0300 Active Virtual Volume Report 39
ERPT0301 Scratch Virtual Volume Summary Report 40

Chapter 6: Reactivate Accidentally Scratched Virtual Volume 43

Reactivate Accidentally Scratched Virtual Volume 43

Chapter 7: Backing Up and Restoring CA Vtape Control Data Sets 45

Back Up Control Data Sets 45
Back Up BSDS 46
Restore or Recover CA Vtape Control Data Sets 47
Recover BSDS from Global VCAT 47
Restore BSDS from Backup 48
Recover Global VCAT from BSDS 48
Redefine Local VCAT 49

Chapter 8: Using Bidirectional Replication for Disaster Recovery 51

Exploit Bidirectional Replication for Disaster Recovery 51
Bidirectional Replication Example 52
Bidirectional Replication Data Flow 53
Bidirectional Replication Configuration 54
Configure parmlib 55
Configure NFS Exports 55
Create Subdirectories for NFS Export Mounting 56
Update FSTAB Entries to Map the Mount Points to NFS Exports 57
Use cacloud vol_mkdir to Define VOLSER Ranges to Linux Systems 58

Test Configuration 59

Chapter 9: Recovering CA Vtape Subsystem 61

Prepare for Disaster Recovery 61
Recover at the Disaster Recovery Data Center 62

Chapter 10: Troubleshooting 65

Virtual Devices Will Not Vary Online 65
Virtual Volume Shortage 66
SO8B Abends 66
GTF Trace 67
IPCS 67
    IPCS Parameters to Print the Logger 67
Self-Documenting Error Recovery Routines 70

Appendix A: Expanding and Converting Control Data Sets 71

Expand and Convert Control Data Sets 71

Appendix B: Health Checks 73

VTAPE_CDS_SEPARATION 73
VTAPE_MODULE_CONSISTENCY 74
VTAPE_PARM_zIIP_STATUS 75
VTAPE_PARM_zIIP_CONFLICT 76

Appendix C: Performing Riverbed Disaster Recovery 79

How to Prepare and Perform a Riverbed SteelStore Appliance Cold Disaster Recovery 79
    Review Riverbed SteelStore Documentation for Cold Recovery 80
    Prepare for Cold Disaster Recovery 80
    Perform a Cold Disaster Recovery Test 81
    Perform Cold Disaster Recovery 82
How to Prepare and Test a Riverbed SteelStore Warm Disaster Recovery 85
    Review SteelStore Riverbed Documentation for Warm Recovery 86
    Prepare for Warm Disaster Recovery 86
    Isolate Virtual Volume Volser Ranges 87

Chapter 11: Configure Appliances for Peer Replication 89

    Peer Replication Disaster Recovery Scenarios 91

Appendix D: Configure Disk Appliances for Replication 93
Configure EMC Data Domain for Replication 93
Configure NetApp for Replication 93
Glossary 95

Chapter 1: About this Guide

This guide contains information to assist in deploying CA Cloud Storage for System z components on the Linux on System z platform.

This section contains the following topics:

Audience (see page 9)
Conventions (see page 9)

Audience

Users of this guide should be experienced mainframe technicians with knowledge of their mainframe tape systems, tape related software, and security configuration.

Conventions

The following conventions are used throughout this guide to document features, functions, and other aspects of the system:

- Variable text, most commonly used for data set names and console commands, is entered in italics.

  For example, VVE_SCRATCH=*volser* where *volser* is the Virtual Volume VOLSER.

- Commands are entered in uppercase and lower-case. The uppercase portion is the minimum number of characters that must be entered for CA Vtape to recognize the command. The lower-case portion is provided for clarity.

- Features, functions, and components of CA Cloud Storage for System z are capitalized. These include, for example: Virtual Volumes and Virtual Devices.

Chapter 2: Introduction

This product allows you to use your existing hardware to create a virtual tape solution in the cloud. For system requirements, see the *Release Notes* that are on the Bookshelf.

This section contains the following topics:

Limited-Use CA Vtape License Requirement (see page 11)
Multisystem Considerations (see page 11)

Limited-Use CA Vtape License Requirement

New and existing CA Vtape customers require the Limited-Use CA Vtape License to implement CA Cloud Storage for System z.

Multisystem Considerations

In a multisystem environment, the systems cannot concurrently access the physical tape data sets. A tape created on one system can only be read or modified on a different system after the first system releases the tape. Each CA Vtape subsystem can access a specific Virtual Volume as often as required. However, only one subsystem can access a specific Virtual Volume at a time.

You can simultaneously run up to eight CA Vtape subsystems within a single Logical Partition (LPAR). Each subsystem runs in its own set of address spaces and must be assigned a unique range of Virtual Device addresses. Once assigned and varied online by one of the subsystems, other subsystems in the same LPAR cannot use the same Virtual Device addresses.

The same Virtual Device address range can be defined on multiple LPARs for the use of a single CA Vtape subsystem on each LPAR. Do not define the devices as shared in the operating system hardware configuration definitions or in an automatic tape switching product such as CA MIA or IBM Tape Auto Switch. Virtual Devices 0100-010F on LPAR A are not the same Virtual Devices as 0100-010F on LPAR B because separate subsystems control them. Each of these devices can be online and actively servicing virtual tape mounts on both LPARs at the exact same time. The only restriction is that they cannot mount the same Virtual Volume simultaneously.

A CA Vtape Complex is created when multiple CA Vtape subsystems in the same LPAR or in different LPARs share the control data sets and the network appliance where the Virtual Volumes reside. The CA Vtape subsystems participating in a CA Vtape Complex can read or write any Virtual Volume data with any other subsystem participating within the same CA Vtape Complex.

A SYSPLEX is not required to support a CA Vtape Complex. Some CA Vtape features are not fully exploited unless CA MIM or IBM's GRS can propagate enqueues across all the LPARs where CA Vtape is implemented.

Chapter 3: Understanding the Basic Components

This section contains the following topics:

CA Vtape (see page 13)
Virtual Volumes (see page 14)
Virtual Volume Compression (see page 15)
Support for Cloud Storage Gateways (see page 15)
Tape Mount Interception (see page 16)
Split Maintenance-Level Protection (see page 19)
System Logger (see page 20)
Volume Pool Definitions (see page 20)

CA Vtape

CA Vtape simulates tape devices within the operating system. Although these devices do not physically exist, they appear to the operating system as if they do. A CA Vtape Virtual Device responds as if a real device is attached. It has functionality equivalent to the microcode that normally controls a hardware tape device.

Virtual Devices are defined to the operating system as a part of the CA Vtape installation. You use the IBM Hardware Configuration and Definition (HCD) software to create the definitions. These are software, not hardware, definitions requiring only an HCD activation to use. An IPL is not necessary.

The CA Vtape subsystem uses between two and nine address spaces on MVS that communicate with the CA Cloud Storage for System z server running on Linux.

The MVS address spaces are:

- The Primary address space – SVTS, which manages the control data sets, queues, resources, startup, shutdown, and operator communications.
- 1-8 Virtual Device Controllers – SVTSAS.SVT*n*V*m*. These are sub address spaces that act as Virtual Device controllers. These controllers provide subtasks that emulate Virtual Tape Drives attached to MVS. Individual Virtual Devices or the entire Virtual Controller and all of its devices can be restarted just like physical tape drives.

On Linux, the CA Cloud Storage for System z Server provides services to read and write Virtual Volume files as standard Network File System (NFS) files.

Virtual Devices use buffers and dataspaces to simulate tape devices. When an application issues reads or writes to a Virtual Volume, data movement occurs between the buffers used by the application and those of the Virtual Device. This movement occurs asynchronously and independently of the I/O process running in the application.

When an application dismounts a Virtual Volume and releases a Virtual Device, the Virtual Volume and Virtual Device are immediately available for other requests.

Virtual Volumes

Define an exclusive range of Volume Serial (VOLSER) numbers as new scratch tapes to the tape management system before using CA Vtape the first time. These same Virtual VOLSERs are also added to CA Vtape. Most tape management systems support the definition of these new scratch tapes as Virtual Volumes or as residing in a nonexistent location. Consult your tape management system documentation for details.

The tape management system manages the Virtual Volumes as if they are real tape volumes. The only communication that is required between CA Vtape and the tape management system is to synchronize which volumes are in scratch status. This synchronization process is accomplished by adding a CA Vtape utility to the tape management system scratch process. Some tape management systems, like CA 1, have an automatic scratch notification feature that can be used instead of the CA Vtape utility program.

All the protection, management, and control services that tape management systems for real tape processing typically provide are also provided for CA Vtape Virtual Volume processing.

Virtual Volume Compression

Virtual Volume compression lets you simulate Improved Data Recording Capability (IDRC) for the CA Vtape Virtual Volumes. Achieve IDRC simulation by compressing the data:

- when the Virtual Device Engine receives it from the application, and
- prior to writing the data to a file on the Storage Gateway Appliance

Compression has multiple advantages:

- Compressed Virtual Volumes occupy less DASD space.
- The effective increase in Virtual Volume capacity allows CA Vtape to manage more application data.
- Compressed data requires fewer I/Os to read or write. The expense is CPU overhead to perform the actual data compression.

Note: The CPU overhead and the compression rate are a function of the data. While our benchmark testing commonly indicated a 2-to-1 compression rate, your rate can vary.

The CA Vtape compression routines use a combination of techniques, including:

- the run length limited (RLL) processing facility
- the hardware compression facility as described in the IBM publication *Enterprise* Systems *Architecture/390 Data Compression, SA22-7208*
- LZ compression through hardware instructions, software instructions, or both

When Virtual Volume Compression is enabled, the Virtual Device Engine periodically analyzes the data that is written to a Virtual Volume to select the most efficient compression technique dynamically.

Support for Cloud Storage Gateways

Storage gateways are networked appliances that allow you to store virtual Volume data at a cloud service provider.

Virtual Volumes are maintained as normal files that can physically reside on various physical devices using iSCSI. The files could be stored on NFS disk, storage gateways, the cloud, or a combination. For cloud gateway support, see the hardware requirements in the *Release Notes*.

Tape Mount Interception

When allocation processing for a new data set occurs at the operating system tape selection exit (Subsystem Interface 78), CA Vtape decides whether to intercept the allocation by accessing Data Set Name Filters and Data Class Filters. For more information about filters, see the chapter "Tape Mount Intercept Filters" in the *z/OS Configuration Guide*.

If the data set to be written matches an entry in the selection filters, CA Vtape marks all devices, except the Virtual Devices, ineligible in the Eligible Device List (EDL). If no matching entry is found in the selection filters, the Virtual Devices are marked ineligible. The System Resources Manager (SRM) then selects an eligible device from the modified EDL to service the allocation request.

If the esoteric device group or generic device type specified in the UNIT parameter does not contain the Virtual Device addresses, CA Vtape leaves the EDL unmodified.

If the UNIT parameter contains a specific device address that is defined as a CA Vtape Virtual Device, this device address is used, even if no data set matches an entry in the selection filters. CA Vtape must service the mount request.

Note: Oracle HSC, CSC, and SMC software also modify the EDL to influence tape allocation requests. If this software is not modified for the presence of CA Vtape, it can prevent CA Vtape from intercepting any allocation request by always marking the Virtual Devices ineligible. For the appropriate Oracle software changes, see the chapter “System Setup” in the *Configuration Guide*.

After SRM selects a Virtual Device for a new tape data set, the Virtual Device Engine detects the actual mount request and creates the Virtual Volume dataspace.

When allocation processing occurs at the operating system tape selection exit (Subsystem Interface 78) for an existing data set residing on a CA Vtape Virtual Volume, all devices except the CA Vtape Virtual Devices are marked ineligible in the Eligible Device List (EDL).

Note: The esoteric or generic specified in the UNIT JCL parameter or in the device type field of the data set catalog entry must contain Virtual Device addresses. If not, CA Vtape cannot intercept the mount request.

SRM selects the actual device from this list and requests a mount for a specific Virtual Volume on the selected Virtual Device.

When allocation processing for a new data set occurs at the operating system tape selection exit (Subsystem Interface 78), CA Vtape decides whether to intercept the allocation by accessing Data Set Name Filters and Data Class Filters.

Close Processing

All data is written to the Virtual Volume file at dismount, close, or any tape mark processing is allowed. This ensures data integrity and tape position synchronization between the application and the Virtual Volume file at appropriate times.

Sense Processing

CA Vtape provides sense and error recovery information emulating a simple 3480 or 3490 device. The sense data emulates non-automated drives without autoloaders.

Multivolume Output Processing

CA Vtape forces an End-of-Volume condition during output processing for the following conditions:

- The end of a Virtual Volume.
- The number of physical blocks that are written to the Virtual Volume exceeds 500,000.
- The number of files that are written to a 400 MB or 800-MB Virtual Volume reaches 256 or reaches 9999 for all higher capacity Virtual Volumes.

Multiple File Processing to Virtual Volumes

Multiple file processing or stacking to Virtual Volumes is supported, but not typically needed for most application data sets.

Stacking to Virtual Volumes makes sense when running backup jobs which write a data set for each data set or table being backed up. For example, a job backing up 100 data sets or tables writing one data set or table per Virtual Volume would select 100 scratch VOLSERs, update those 100 VOLSERs in the Global VCAT, and allocate 100 files. If the data sets or tables fit on five full Virtual Volumes if stacked, the time spent performing these tasks would be reduced by 95 percent. These types of jobs run faster when stacking than not stacking.

CA Vtape limits the number of files per Virtual Volume to 256 for 400-MB and 800-MB Virtual Volumes. Higher capacity Virtual Volumes have a limit of 9999 files per Virtual Volume. When the file limit is reached, CA Vtape indicates an end of volume condition and performs a volume switch.

If more than one data set is written to a Virtual Volume, all data sets assume the same group number as the first volume data set. The group number determines the volume pool from which the virtual VOLSER is selected.

Uncataloged Data Sets

Uncataloged data sets written to Virtual Volumes are supported in the same way as all uncataloged tape data sets. To mount this volume again, a VOL=SER reference is required in your JCL or dynamic allocation.

Temporary Data Sets

Temporary data sets can be written to Virtual Volumes, whether dynamically allocated or explicitly specified by the user. Simply create a Data Set Filter or a Data Class Filter for the system-generated data set names.

Unit Affinity and GDG Base Referral Processing

Unit Affinity and GDG base referral processing are supported.

If unit affinity, explicitly coded or implied, mixes CA Vtape Virtual Devices and non-CA Vtape devices, the affinity is broken but maintained within the device type. For example, if a GDG has generations on 3490E physical tape and generations on Virtual Volumes, one physical drive and one Virtual Device are required to read all generations with a GDG base referral.

Reference Backward Processing (Refer Backs)

Reference backward (refer backs) processing is the referencing of a previous DD statement. Refer backs are supported if all data set names reside on CA Vtape Virtual Volumes.

Concatenation Processing

Concatenation processing can have references in the same job step to a data set both serviced and not serviced by CA Vtape. Use Unit Affinity when concatenating Virtual Volumes. Unit Affinity minimizes the number of Virtual Devices that are allocated and avoids conflicts during the tape allocation process.

Special Processing Conditions

The following special processing conditions exist:

OPTCD=W

Specified in JCL; causes *Tape Write Immediate*, which, on physical 3480s and 3490s, forces a physical write to tape before returning channel end and device end to the host. CA Vtape processes OPTCD=W if requested; however, some records can be in flight and not yet saved on DASD when channel end and device end are returned to the host.

OPTCD=C

Specified in JCL; forces *channel command chaining* and is accepted by CA Vtape.

OPTCD=Z

Specified in JCL; indicates shortened error recovery for physical tapes. This OPTCD is accepted by CA Vtape but does not affect processing.

LABEL=n

The Tape Management System does not support tape label types. That is, LABEL=AL can result in the Tape Management System exhausting all available virtual tapes. Avoid unsupported label types, whether for a specific or a nonspecific volume request.

Split Maintenance-Level Protection

A CA Vtape Subsystem is a main address space and multiple subaddress spaces. To clear error conditions that can cause product outage, you can restart subaddress spaces individually. If maintenance is applied, the link list is changed, or the CA Vtape PROCs are changed, a subaddress space could be restarted with a different code level than the other address spaces in the subsystem.

The Split Maintenance-Level Protection feature addresses the problem. When this feature is active, all module PTF levels are reviewed for incompatibilities during a subaddress space restart. If discrepancies are found, warning messages are displayed.

If the Automatic or Library modes of this feature are selected during startup, the CA Vtape load modules are copied to a private library. This library is used to start and restart the subaddress spaces. This effectively freezes the runtime code maintenance level to prevent any changes from being introduced until the subsystem is restarted.

System Logger

The System Logger was introduced in the operating system to collect data that transactional applications and databases generate. CA Vtape creates log records for internal events. Copy these log records to the system logger and use them for statistical reports and problem analysis. The CA Vtape Logger consists of two customizable components: the Internal and the External Logger.

Internal Logger

The CA Vtape Internal Logger uses a dataspace that is created at startup as a repository for the logger data. The parmlib attribute in the VTPARMS member determines the database size. The default value is 8 MB. Any CA Vtape automatically generated dump or any manually requested dump that is generated by executing the CA Vtape STV*n* DUMP console command includes the Internal Logger data.

External Logger

The CA Vtape External Logger allows you to optionally offload the internal logger data to a system logger log stream. By implementing the External Logger, logged events can be kept for several days in the system logger log stream. You can then copy to physical sequential data sets for archival. This feature is required for statistical reports and strongly recommend for use in diagnosing problems.

Volume Pool Definitions

Volume Pool Definitions make it possible to associate Virtual Volser ranges with specific pool names. The pool name can then be used to associate any of the defined ranges with specific data sets utilizing the Group and Filter Definitions.

Up to eight pools can be defined.

The Group Definitions are updated with the defined pools names. When a scratch mount is intercepted, the filter which caused the intercept assigns a group number and by association a pool name. The Volume Pool Definitions are then used to assign a scratch VOLSER from that pool to the mount.

Consider the following:

- Volume Pooling is required when using control data sets that allow more than 510,800 Virtual Volumes to be defined.
- Multiple pool definitions are not supported under JES3. Under JES3 only one volume pool can be defined.

Chapter 4: ISPF Interface

The CA Vtape ISPF panels let you view information about:

- Tape devices
- Virtual Volumes
- Filter lists

This section contains the following topics:

Start the ISPF Interface and Access the Main Menu (see page 22)
Scratch Pool Header Information (see page 23)
Using the Tape Device Status Display Panel (see page 24)
Using the Volume Pool Display Panel (see page 25)
Using the Virtual Volume Display Panel (see page 26)
Using the Dataset and DataClass List Panels (see page 31)

Start the ISPF Interface and Access the Main Menu

The CA Vtape ISPF panels let you view information about:

- Tape devices
- Virtual Volumes
- Filter lists

Learn about installing and customizing the panels here.

Start the ISPF Interface

To start the ISPF Interface, execute the SVTSMON member from the CCUUEXEC library.

The main menu displays as follows:

```
---------------------------   CA Vtape   ---------------------

     Subsystem ID......... SVT1  (SVTn, where n is 1-8)
     Select one of the following:


     1  Group List         Display output group information
     2  Tape Devices       Display SVTS related tape devices
     3  Virtual Volumes    Manage virtual volumes
     4  Dataset List       Manage the dataset filter list
     5  Dataclass List     Manage the Dataclass filter list



     Enter PF3 to exit


  Option ===>
```

Note: For information about the common scratch pool information that appears on all screens, see Scratch Pool Header Information (see page 23).

From the main menu, you can select the following items:

1 Group List (Disabled)

This option is disabled for CA Cloud Storage for System z. It shows information about the CA Vtape Externalization subgroup queues.

2 Tape Device (see page 24)

This option shows information about the Tape Devices Tape Devices that are defined to or in use by CA Vtape.

3 **Virtual Volumes** **(see page 26)**

This option shows information about Virtual Volumes.

4 **Dataset List** **(see page 31)**

Shows information that you can use to display and manage the data set list filters. This option is deactivated when you use the parmlib-based enhanced filters.

5 **Dataclass List** **(see page 31)**

This option shows information that you can use to display and manage the data class entries. This option is deactivated when you use the parmlib-based enhanced filters.

Scratch Pool Header Information

Many panels have the following common information beneath the command line:

```
                        Panel Title        Row 1 to nn of mm

Command ===> ____________________________________________________

             Scratch counts by Volume Pool
Pool1=n/a     Pool2=n/a     Pool3=n/a     Pool4=38
Pool5=1000    Pool6=n/a     Pool7=n/a     Pool8=n/a
```

The information includes the following:

Panel Title

The title line of the page.

nn* of *mm

nn indicates the number of rows of information on this panel and *mm* indicates the total number of displayable rows.

Pool*n*

Displays the number of Virtual Volumes in scratch status in Pool1 through Pool8. "N/A" is displayed for a pool that is not defined.

Using the Tape Device Status Display Panel

Select option 2 from the main menu (see page 22) to view information about all the Virtual Devices owned by this Subsystem.

The following example panel is for a Subsystem with nine Virtual Devices (3513-3517 and 3504-3506):

```
   Menu  Functions  Confirm  Utilities  Help
--------------------------------------------------------------------------------
   File  Help
--------------------------------------------------------------------------------
               CA Vtape Tape Device Status Display    Row 1 to 10 of 10
Command ===>

             Scratch counts by Volume Pool
Pool1=n/a     Pool2=n/a     Pool3=n/a     Pool4=38
Pool5=1000    Pool6=n/a     Pool7=n/a     Pool8=n/a

The following device list contains information on virtual tape and
physical devices allocated to CA-Vtape.
Volume  Dev#  Status      Devt  Jobname   # SIOs  SEQ#   Phy/Virt  %
n/a     3513              349S             14      0     Virtual   0
n/a     3514              349S             4       0     Virtual   0
n/a     3515              349S             14      0     Virtual   0
n/a     3516              349S             14      0     Virtual   0
n/a     3517              349S             4       0     Virtual   0
n/a     3518              349S             14      0     Virtual   0
n/a     3504              349S             14      0     Virtual   0
n/a     3505              349S             14      0     Virtual   0
n/a     3506              349S             14      0     Virtual   0
+-----------------------------------------------+-------------------------
|The cross-system information has been refreshed |
+-----------------------------------------------+
```

For more information about the Scratch Pools, see Scratch Pool Header Information (see page 23).

The information includes the following:

Volume

Displays the volume serial number that is associated with the device.

Dev#

Displays the device number or unit address.

Status

Displays the status of the device.

Devt

Displays the device type.

Jobname

Displays the jobname that has the device allocated.

 # SIOs

Displays the number of Start I/Os to the device.

Seq#

Displays the file sequence number.

Phy/Virt

Displays "Virtual" for CA Vtape Virtual Devices and "Physical" for devices.

%

For Virtual Devices always indicates zero.

Using the Volume Pool Display Panel

Select option 3 from the main menu (see page 22) to view the Virtual Volume information.

The following example panel is for a Subsystem with Virtual VOLSERs 100000-100699 defined in its parmlib and split between Pool1and Pool3:

```
                  CA Vtape Volume Pool Display        Row 1 to 6 of 32
 Command ===>

              Scratch counts by Volume Pool
Pool1=n/a     Pool2=n/a     Pool3=n/a     Pool4=38
Pool5=1000    Pool6=n/a     Pool7=n/a     Pool8=n/a

 Virtual Volume Pool defined volume ranges. Use 'S' to show volumes
 within the selected range. Scratch volumes are not displayed.

     1st Volid  Last Volid  Pool#
_    100000     100099        3
_    100100     100199        3
_    100200     100299        3
_    100300     100399        3
_    100400     100499        1
_    100500     100599        1
```

VOLSERs are displayed in sets of 100, *xxxx*00-*xxxx*99, from the lowest VOLSER defined to the highest VOLSER defined. The number of the pool the set of 100 VOLSERs is defined to is also displayed.

Page up and down the list of 100 VOLSER sets with the ISPF Up and Down PF keys. If your paging default is cursor, you can reposition in the list by typing the number of lines to move on the command line and using the Up and Down PF keys. To display an individual Virtual Volume, type an (S)elect in front of the range of 100 Virtual Volumes containing the desired VOLSER.

The information shown in the Volume Pool Display panel includes the following:

Note: For more information about the Scratch Pools, see Scratch Pool Header Information (see page 23).

1st Volid

Displays the starting VOLSER in a set of 100 VOLSERs. A set always starts with vvvv00.

Last Volid

Displays the last VOLSER in a set of 100 VOLSERs. A set always ends with vvvv99.

Pool#

Displays the number of the pool in the parmlib Volume Pool Definitions Section that this set of VOLSERs is defined to.

Using the Virtual Volume Display Panel

The Virtual Volume Display Panel includes information about the compressed and uncompressed Virtual Volume size. Only Virtual Volumes that are not in scratch status are displayed.

The following example panel displays when (S)elect was typed in front of range 100100-100199 and all VOLSERs in that range were in scratch status except VOLSERs 100171 and 100187:

```
                  CA Vtape Virtual Volume Display Panel     Row 1 to 2 of 2
 Command ===>

             Scratch counts by Volume Pool
Pool1=n/a     Pool2=n/a     Pool3=n/a     Pool4=38
Pool5=1000    Pool6=n/a     Pool7=n/a     Pool8=n/a

 Use S to display additional virtual volume data                          E
   Volid   Tape DSN                                  VRM Cmp%  #MB Size Allo S
 _ 100171+ ZOBKU01.SVTS61.P1802$W1                     2    0    8    8    8 Y
 _ 100187  SVTS120.QASVT4.G5197.PASS                  0    0  324  324  324 Y
 ****************************** Bottom of data ******************************

 The cross-system information has been refreshed
```

The information shown includes the following:

For more information about the Scratch Pools, see Scratch Pool Header Information (see page 23).

Volid

Virtual VOLSER number.

Tape DSN

Data set name associated with the Virtual Volume. If the volume contains multiple DSNs, a plus sign is shown next to its VOLSER. To view these additional DSNs, select the entry by placing an S next to the entry.

VRM

Stands for Version Release Maintenance. It is a value assigned to a level of code where a significant new feature or function was introduced. As Virtual Volumes are created, they are assigned the lowest VRM value required to properly support them. The assigned value is used during subsystem start up to ensure that the level of code being used can properly support the existing Virtual Volumes.

Cmp%

Percentage the Virtual Volume was compressed in the DASD buffer.

#MB

The amount of data written by the application to the Virtual Volume tracked in 4-MB segments.

Size

The compressed size of Virtual Volume data rounded up to a 4 MB boundary.

Allo

The current size of the Virtual Volume file.

Res

A flag that indicates if the Virtual Volume resides in the DASD buffer. This flag will always display a (Y)es for CA Cloud Storage for System z.

The (S)elect line command can be used to display additional information about individual Virtual Volumes. The following panel will be displayed:

```
Menu  List  Mode  Functions  Utilities  Help
  File  Help
------------------------------------------------------------------------------
                CA Vtape Virtual Volume Display Panel      Row 1 to 1 of 1
Command ===>  ____________________________________________________________


Volser . . . : 102246    Flags  . . . : 24800000  Number of TMs   5
Label Type . . . : 02    Group ID . . : 11  S     Dev Type . . : 78048081

Tape DSN                                          Rec Sz  BLK Sz  FM   Seq #
 SVTS.TGRP51.RETPD022                              100    22000   FB    1
****************************** Bottom of data ******************************
```

The information shown in the panel includes the following:

Volser

Virtual Volume displayed.

Flags

A four-byte hexadecimal field comprised of the FLAG1, FLAG2, FLAG3, and FLAG4 fields. For more information about the flag settings and their meanings, see Virtual Volume Flag Fields (see page 29).

Number of TMs

Number of Tape Marks written to the Virtual Volume.

Label Type

For more information about Label Type settings, see Label Types (see page 30).

Group ID

The assigned group.

Dev Type

The Virtual Volume device type.

Tape DSN

The 44 characters of the DSNs written to the Virtual Volume.

Rec Sz

Record size of the data set.

BLK Sz

Block size of the data set.

FM

Format type of the data set.

Seq #

File sequence number of the data set on the Virtual Volume.

Virtual Volume Flag Fields

The following table shows the Virtual Volume flag fields and their meanings:

Byte #	Binary	Hex	Meaning
VVEFLAG1	1000 0000	80	CANNOT BE FREED
	0100 0000	40	MUST BE EXTERNALIZED
	0010 0000	20	VVE IN CACHE
	0001 0000	10	DATACLASS LIST
	0000 1000	08	BYPASS BSDS
	0000 0100	04	PRIMARY
	0000 0010	02	DUPLEX
	0000 0001	01	EXPORT
VVEFLAG2	1000 0000	80	NOT SCRATCH
	0100 0000	40	NL SL LABELED
	0010 0000	20	MULTI VOLUME DSN
	0001 0000	10	GRR MANUALLY DEQUEUED
	0000 1000	08	WRITE PROTECT
	0000 0100	04	MORE THAN 44 DATA SETS
	0000 0010	02	HW COMPRESSION
VVEFLAG3	1000 0000	80	VVE SAVE REQUIRED
	0100 0000	40	RECALLING VVE
	0010 0000	20	VVE INITIALIZED
	0001 0000	10	RETRY LDS

Byte #	Binary	Hex	Meaning
VVEFLAG4	1000 0000	80	LDS ALLOCATED
	0100 0000	40	DIV INDENTIFIED
	0010 0000	20	DIV ACCESSED
	0001 0000	10	DATA SPACE CREATED
	0000 1000	08	MOUNTED
	0000 0100	04	RECALL ACTIVE
	0000 0001	01	MAPPED

Label Types

The following table lists the label types, settings, and meanings:

Label Name	Binary	Hex	Meaning
JFCDSEQN	1000 0000	80	DATA SET SEQUENCE NUMBER Specified
JFCBAL	0100 0000	40	AL: ISO/ANSI (ver 1) ISO/ANSI/FIPS (ver 3)
	0100 1000	48	AUL User labels and AL type labels
JFCBLTM	0010 0000	20	LTM LEADING TAPE MARK **Note:** OPEN/CLOSE/EOV and RESTART must space over a tape mark if one exists.
JFCBLP	0001 0000	10	BLP BYPASS LABEL PROCESSING
JFCSUL	0000 1010	0A	SUL STANDARD and USER LABELs
JFCNSL	0000 0100	04	NSL NONSTANDARD LABEL
JFCSL	0000 0010	02	SL STANDARD LABEL (default)
JFCNL	0000 0001	01	NL NO LABEL

Using the Dataset and DataClass List Panels

CA Vtape supports the following two filter modes:

- Basic
- Enhanced

CA Cloud Storage for System z supports Enhanced mode filters.

To avoid confusion, Enhanced mode deactivates the ISPF Interface Main Menu options 4 and 5.

Chapter 5: Reports

This chapter describes sample jobs and reports that are provided with CA Vtape. The first half of the chapter describes the standard reports that are generated by running the SVTSUTIL utility. The second half of the chapter describes the customizable reports that are generated by running CA Earl.

This section contains the following topics:

Using SVTSUTIL Batch Commands to Generate Reports (see page 33)
Use CA Earl with CA Vtape (see page 35)
CA Vtape Supplied CA Earl Reports (see page 37)

Using SVTSUTIL Batch Commands to Generate Reports

The following SVTSUTIL Batch Commands produce reports:

- ANALYZE=COMPRESSION (see page 33)
- LIST=MODULE (see page 34)

Only one report command can be used per execution of the utility.

ANALYZE=COMPRESSION

Use this command when running the SVTSUTIL program to generate a report showing the effective CPU cost for using hardware compression in the DASD buffer for a generated data sample. The CA Vtape compression routines rely on the use of certain hardware instructions. This report can estimate how much additional CPU overhead is required to compress data using these hardware instructions.

Syntax

```
ANALYZE=COMPRESSION
```

The additional CPU cost is expressed in terms of megabytes (MBs) for each CPU second. This reflects the amount of data the hardware compression instructions themselves can move. Use the report information to estimate the additional CPU cost for hardware compression that is based on the number of MBs of data to process.

You can also use this report to determine how to adjust the MaximumCPURate parmlib attribute. This attribute can reduce or increase the additional CPU cost by reducing or increasing the amount of data compressed. By reducing the amount of data that is compressed, CPU cost can be reduced.

The analysis report tends to vary on different physical machines due to the hardware implementation of the compression instructions.

Sample report output

```
    CA Vtape 12.6        S V T S U T I L    -   Snnn.SVT1     -  ddddddddd  mmm dd, yyyy  hh.mm.ss

--------------------------- Utility Control Statement(s) & Report Log ---------------------------
SVTSU0171I ANALYZE=COMPRESSION

Compression Call Performance Analysis

     Type     Method   MB/Inp CpuSec MB/Sec
     -------- -------- ------ ------ ------
     CMPSC    SVTHC#01     25   .127    201
     CMPSC    SVTHC#02     25   .118    216
     CMPSC    SVTHC#03     25   .105    243
     CMPSC    SVTHC#04     25   .105    243
     CMPSC    SVTHC#05     25   .114    224

        Average Velocity mb/Sec         225

SVTSU0172I ANALYZE=COMPRESSION                                          SVTSUT60,RC=00

SVTSU0173I Number of commands processed:        1
SVTSU0174I Highest condition code:              0
```

LIST=MODULE

The LIST=MODULE SVTSUTIL command provides a Module Revision List (MRL) report pertaining to CA Vtape loadlib members. The loadlib must be defined by a SVTSLOAD, JOBLIB or STEPLIB DD statement, or must be included in the operating system Link List (LINKLST) definition.

The MRL report is also automatically created for the libraries accessed by the SVTS address space during CA Vtape initialization. The MRL report is written to the SVTS JOBLOG.

You can use the SVTSLOAD DD statement to generate MRL reports from previous versions of CA Vtape load libraries or, while executing SVTSUTIL from one CA Vtape loadlib, verify the maintenance level of CA Vtape modules after PTFs or Service Pack installations to a different library.

When the SVTSLOAD DD statement is not specified, SVTSUTIL tries to generate the MRL report from the libraries specified as JOBLIB or STEPLIB DD statements.

When the SVTSLOAD, JOBLIB or STEPLIB DD statement are not included, SVTSUTIL tries to process the operating system LINKLST. SVTSUTIL then generates the report based on the CA Vtape load modules found on the LINKLST definition.

The MRL report is written to the SYSUT1 DD statement. The SYSPRINT DD statement provides operational return codes and messages.

Syntax

```
LIST=MODULE
```

Example of the List Module report

```
    CA Vtape 12.6        S V T S U T I L    -   Snnn.SVT1     - ddddddddd  mmm dd, yyyy  hh.mm.ss

-------------------------- Utility Control Statement(s) & Report Log ---------------------------
SVTSU0171I LIST=MODULE
   LOADLIB DDName: SVTSLOAD (A)
           DSName: CSLVL2.VTAPE.R126.CCUULOAD (B)
      ModuleName Description (C)                          Address   AM  SP  ---- ATTRIBUTES ----
   +0  SLSUX02   SLSUX02  14.48 20110218 CCUUC60-RESERVE
   +0  SVTCHEIO  SVTCHEIO 20.55 20120330 CCUUC60-RO43755
   +0  SVTCJMLW  SVTCJMLW 14.48 20110218 CCUUC60-RESERVE  A052F6C8  31  FC  RE RU
   +0  SVTCMDPA  SVTCMDPA 21.17 20130319 CCUUC60-RO54767
   +0  SVTCMMRL  SVTCMMRL 14.48 20110218 CCUUC60-RESERVE  A0553000  31  FC  RE RU
   +0  SVTDOUT   SVTDOUT  14.48 20110218 CCUUC60-RESERVE
   +0  SVTDSCNV  SVTDSCNV 14.48 20110218 CCUUC60-RESERVE
   Etc.
   Module count: 96 (D)
 END Module Revision Level Log
SVTSU0004I MRL request successfully processed
SVTSU0172I LIST=MODULES                                              SVTSUT60, RC=     0

SVTSU0173I Number of commands processed:        1
SVTSU0174I Highest condition code:              0

Legend:
(A) DD name of the loadlib or loadlibs.  Values could be STEPLIB, JOBLIB, SVTSLOAD, or LNKLIST.
(B) Data set name of the loadlib or loadlibs if concatenated.
(C) Description is composed of five items:
    a.  Module name.
    b.  Time of last assembly in military format of HH.MM.
    c.  Date of last assembly in YYYYMMDD format.
    d.  Product FMID.  CCUU indicates Vtape.  C60 indicates 12.6.0.
    e.  RMID.  RESERVE indicates a base module.  PTF numbers are in XXnnnnn format.
(D) The number of modules found in the loadlib or loadlibs.  The current count is 96.
```

Use CA Earl with CA Vtape

CA Earl, provides you with the capability to design and produce customized reports. Easy access to system information provides flexibility and lets you tailor reports to your desired format.

These jobs are distributed in the HLQ.CCUUECPB data set. A cross reference of supplied CA Earl members can be found in the E$INDEX member of the HLQ.CCUUJCL data set.

For information about CA Earl, see the *CA Earl Reference Guide*.

Sample CA Earl Components

The following list shows the supplied members that are used to produce the various CA Earl reports:

Procedures

Member: EREPORT

JCL procedure used to run CA Earl reports. This procedure requires customization.

CA Earl File Definitions and Record Layouts

Member: EFMTGRP

CA Earl copybook member that provides data and file definitions for accessing the CA Vtape Group (GRP) and Sub-Group (SGRP) records.

Member: EFMTLOG

CA Earl copybook member that provides data and file definitions for accessing the CA Vtape Group log records.

Member: EFMTVVE

CA Earl copybook member that provides data and file definitions for accessing the CA Vtape Virtual Volume Entry (VVE) records.

Copy Books

Member: ERPTOPT

CA Earl copybook member that defines CA Earl execution options.

Member: ERPTHDR

Standard heading line format for reports.

JCL Members Required to Produce CA Earl Reports

Member: EJOB0110

JCL to create general data sets used by EJOB0120 and EJOB0130.

Member: EJOB0120

JCL to produce Report ERPT0120.

Member: EJOB0130

JCL to produce Report ERPT0130.

Member: EJOB0300

JCL to produce reports ERPT0300 and ERPT0301.

CA Earl Report Programs

Member: ERPT0120

CA Earl program that creates a daily Virtual Mount Performance Analysis Report.

Member: ERPT0130

CA Earl program that creates a monthly Virtual Mount Performance Analysis Report.

Member: ERPT0300

CA Earl program that creates the Active Virtual Volume Report. The report details the Virtual Volumes currently in use.

Member: ERPT0301

CA Earl program that creates the Scratch Summary Report. The report provides a count of the Virtual Volumes in scratch status.

CA Earl Component Modifications

If you modify the delivered samples and product maintenance is applied that overlays the modified members, your modifications are lost.

Instead of modifying the delivered samples, copy them to new members and modify the new members. The new members can be in the existing product data sets or in new, run-time only data sets.

CA Vtape Supplied CA Earl Reports

The following examples are CA Earl reports that CA Vtape provides.

ERPT0120 and ERPT0130 Virtual Mount Performance Analysis

ERPT0120 produces a daily report by reading the log stream produced by CA Vtape. ERPT0130 produces a monthly report by reading a set of generational data sets produced by ERPT0120.

Example reports

```
dd/mm/yy                                                CA Vtape                                                PAGE    13
hh.mm.ss       ERPT0120                    Copyright(c) yyyy CA. All Rights Reserved

               EOD DETAIL                     VIRTUAL MOUNT PERFORMANCE ANALYSIS

-------------------------------------------------------------------------------------------------------------------------------
SYSTEM     DATE         TIME         TIME      DEVICE  VIRTUAL         #MOUNT        MOUNT       #SCRATCH     #CACHE-HIT    #CACHE-MISS
         REQUESTED    REQUESTED    COMPLETED          VOLSER                      #SECONDS
-------------------------------------------------------------------------------------------------------------------------------
XE61     2003/03/26 23:29:41.38 23:34:22.74 350F    100534              1          281                                          1
XE61     2003/03/26 23:30:27.93 23:30:37.13 3501    100538              1            9                          1
XE61     2003/03/26 23:31:51.00 23:31:56.69 3507    100546              1            5                          1
XE61     2003/03/26 23:35:58.77 23:36:07.02 350F    100543              1            8                          1
XE61     2003/03/26 23:37:24.80 23:37:30.50 350D    100543              1            5                          1
XE61     2003/03/26 23:37:42.06 23:37:47.29 350E    100543              1            5                          1
XE61     2003/03/26 23:38:13.65 23:40:35.61 3500    100547              1          142                                          1
XE61     2003/03/26 23:38:13.66 23:38:44.64 3501    100568              1           31                                          1
XE61     2003/03/26 23:38:13.92 23:40:23.57 350E    100548              1          128                                          1
XE61     2003/03/26 23:38:14.14 23:39:32.37 3507    100550              1           78                                          1
XE61     2003/03/26 23:41:58.81 23:42:32.44 3509    100564              1           33                                          1
XE61     2003/03/26 23:42:35.88 23:45:15.87 350A    100552              1          159                                          1
XE61     2003/03/26 23:42:41.53 23:43:47.18 3500    100555              1           66                                          1
XE61     2003/03/26 23:42:48.10 23:42:54.93 3506    100559              1            7                          1
XE61     2003/03/26 23:50:08.36 23:50:13.01 3509    100560              1            4                          1
XE61     2003/03/26 23:55:28.61 23:55:35.93 350D    100566              1            7                          1
                                                                  -------------                -------------  -------------  -------------
XE61                                                                    664                          182            373            109
                                                                  -------------                -------------  -------------  -------------
                                                                  -------------                -------------  -------------  -------------
GRAND TOTAL                                                             664                          182            373            109
                                                                  -------------                -------------  -------------  -------------

dd/mm/yy                                                  CA Vtape                                                PAGE     1
hh.mm.ss       ERPT0120                    Copyright(c) yyyy CA. All Rights Reserved
               EOD SUMMARY                    VIRTUAL MOUNT PERFORMANCE ANALYSIS

-------------------------------------------------------------------------------------------------------------------------------
SYSTEM     DATE      HOUR        #MOUNT       MOUNT      #SCRATCH     SCRATCH     #CACHE-HIT    CACHE-HIT    #CACHE-MISS   CACHE-MISS
         REQUESTED                          #SECONDS                 #SECONDS                  #SECONDS                   #SECONDS
-------------------------------------------------------------------------------------------------------------------------------
XE61     2003/03/26   11            1          11                                          1          11
XE61     2003/03/26   12            3           8                                          3           8
XE61     2003/03/26   18            7           3            1           5              6           3
XE61     2003/03/26   19          105          13           35           4             64           5             6          152
XE61     2003/03/26   20          155          26           56           5             78           5            21          158
XE61     2003/03/26   21          163          24           34           6             97           6            32          100
XE61     2003/03/26   22          110          32           31           7             58           7            21          139
XE61     2003/03/26   23          120          32           25           7             66           5            29          117
                         -------------            -------------            -------------            -------------
XE61     2003/03/26              664          25          182           6            373           6           109          126
                         -------------            -------------            -------------            -------------
                         -------------            -------------            -------------            -------------
XE61                             664          25          182           6            373           6           109          126
                         -------------            -------------            -------------            -------------
                         -------------            -------------            -------------            -------------
GRAND TOTAL                      664          25          182           6            373           6           109          126
```

ERPT0300 Active Virtual Volume Report

The volume serial numbers shown on this report represent Virtual Volumes that contain active data. You can create an optional output file that allows follow-up processing or extra reporting. The DD statement FILEOUT presence triggers the file creation.

Active Virtual Volumes identified as being in group 0 with a data set name of Volume Unusable - Manual Action Required were selected to be reused by CA Vtape but were rejected as scratch tapes by your tape management system. Identify any action required to recover data if necessary.

The VCAT DD statement is required to resolve subgroup information for the active Virtual Volumes. If you have a single system image, or if your group definitions are identical across all images, a single run of the report generates accurate subgroup information. If you maintain different group definitions on different system images, you may have to run the report against more than one VCAT to obtain accurate results.

Example reports

```
DD/MM/YY                                          CA Vtape 12.6                                                    PAGE     1
HH.MM.SS       ERPT0300                        All Rights Reserved
                                  A C T I V E  V I R T U A L  V O L U M E  R E P O R T

-------------------------------------------------------------------------------------------------------------------------------
    Sub Last Reference System   Volume      Cache  Extrn                                                  Tape     Cmp Last Modified
Grp Grp      Date        ID     Serial VRM   Y/N    Y/N  1st Dataset Name                                 Sz/MB     %      Date
-------------------------------------------------------------------------------------------------------------------------------
 01 S       2013/08/05 CAnn,1   597240 0004   Y      Y   KUCSL02.STAR.DR.#1417684.AL1.BSDS01               956    0.00    2013/08/05
 01 S       2013/08/05 CAnn,1   597241 0004   Y      Y   KUCSL02.STAR.DR.#1417684.AL2.A0071489             952    0.00    2013/08/05
 01 S       2013/08/05 CAnn,1   597256 0004   Y      N   IDI.SYSTEST.T03319A.IC1.P1E.@@1P1E.G0001V00        36    0.00    2013/08/05

(Continues)

 01 S
                                   3,356                                                          2,321,284    0.00

(Each group/subgroup has a total line and each group has a total line)

GRAND TOTAL
                                  68,936                                                         32,803,436    0.00

MM/DD/YY                                          CA Vtape 12.6                                                    PAGE     1
HH.MM.SS       ERPT0300                        All Rights Reserved
                                  A C T I V E  V I R T U A L  V O L U M E  R E P O R T
                                                  Report Summary

GRAND TOTAL   Active Volumes =   68,936     In Cache =    1,902 Externalized =   67,682
              Total Tape Storage(MB) =  32,803,436    Avg Cmp % =  0.00
```

ERPT0300 Active Virtual Volume Optional Output File Record Layout

The record layout for the 90-byte LRECL output file is:

Position	Length	Format	Description
01	6	A/N	Volume Serial
07	1	Y/N	Cache Indicator
08	1	Y/N	Externalization Indicator
09	1	Binary	Group
10	1	A/N	Sub Group
11	44	A/N	Data Set Name
55	3	Binary	Tape size
58	3	Binary	Compression percent
61	8	A/N	System ID - Left Justified
69	10	A/N	Reference Date (YYYY/MM/DD)
79	10	A/N	Last Modified (YYYY/MM/DD)
89	2	Blank	Filler

ERPT0301 Scratch Virtual Volume Summary Report

This report lists the following:

- The total number of available Virtual Volumes in the scratch pool.
- The number of scratch pool Virtual Volumes flagged with the Write Protect status.
- The total number of Virtual Volumes in scratch status in the Global VCAT.

The *Available in Scratch Pool* report line is the total number of scratch volumes residing in the scratch pool. The value indicates the number of scratch volumes available with one exception. If Virtual Volumes were write protected at the pool level (SVT*n* SET WRITPROT,ON,VVP=*vvvvnn*) then these volumes are included in this total. If Virtual Volumes were write protected at the volume level (SVT*n* SET WRITPROT,ON,VVE=*vvvvvv*) then the volumes are not included in this total.

The *Available in Scratch Pool* value is printed as asterisks (*********) if the GLOBAL or VCAT DD statements are not defined, or they reference the wrong Global VCAT or Local VCAT.

The *Virtual Volumes in Write Protect* report line indicates the number of Scratch Virtual Volumes with write protection turned on. Scratch Virtual Volumes that are write protected cannot be written to so they cannot be used as scratch tapes. To make these VOLSERs available for use as scratch tapes, use the SVT*n* SET WRITPROT console command to turn off write protection.

Note: For details on printing a list of write protected VOLSERs, see Obtain the Optional ERPT0301 Detail Scratch Virtual Volume Report (see page 41).

The *Virtual Volumes in Scratch Status* report line is the total number of Virtual Volumes that are marked as scratch volumes. This value includes volumes that are marked with the write protect flag and volumes that are not marked with the write protect flag.

Example of the Summary Scratch Virtual Volume report

```
Summary Scratch Virtual Volume Report

DD/MM/YY                                          CA Vtape r12.6                                                PAGE     1
HH.MM.SS       ERPT0301                        All Rights Reserved
                                  S C R A T C H  V I R T U A L  V O L U M E  R E P O R T
                                                 Report Summary

GRAND TOTAL  Available in Scratch Pool         =          5235
             Virtual Volumes in Write Protect  =            21
             Virtual Volumes in Scratch Status =          5256

             Note: - Write Protect Scratch Virtual Volumes are not available for mount processing unless
                     Write Protect flag is turned off (refer to SVTn SET WRITPROT command).
```

Obtain the Optional ERPT0301 Detail Scratch Virtual Volume Report

To view the Detail Scratch Virtual Volume Report, remove the DUMMY from the optional FILEOUT DD. You can also edit the CA Vtape CA Earl member ERPT0301 in the CA Vtape CCUUEARL library and uncommenting the Detail Scratch Virtual Volume Report lines. The report contains a list of VOLSERs in scratch status in the Global VCAT. If a VOLSER has write protection on, *Write Protected* appears next to the VOLSER.

Example of editing the CA Vtape CCUUEARL library member ERPT0301:

```
*****************************************************************
*       =====>> Detail Scratch Virtual Volume Report <<=====   *
*       To generate a Detail Scratch Virtual Volume Report,    *
*       remove the '!' comment characters from the following   *
*       7 lines (do not remove any '!*' lines).                *
*****************************************************************
REPORT
COPY ERPTHDR USING 'ERPT0301'
TITLE @  39 'S C R A T C H  V I R T U A L  V O L U M E  R E P O R T'
TITLE 'Report Detail'
TITLE '              '
PRINT
     @2 RECORD_OUT
```

Example of the Detail Scratch Virtual Volume report

```
DD/MM/YY                                          CA Vtape r12.6                                          PAGE     1
HH.MM.SS       ERPT0301                           All Rights Reserved

                                  S C R A T C H  V I R T U A L  V O L U M E  R E P O R T
                                                  Report Detail

-------------------------------------------------------------------------------------------------------------------------------
  Volume
  Serial
-------------------------------------------------------------------------------------------------------------------------------
  100000
  100001
  100002  Write Protected
  100003
  100004
  100005
  100006
```

ERPT0301 Detail Scratch Virtual Volume Optional Output File Record Layout

The record layout of the 80-byte LRECL output file is:

Position	Length	Format	Description
1	1	Blank	Filler
2	6	A/N	Volume Serial
8	1	Blank	Filler
9	15	A	Status (Write Protected | Blank)
24	58	Blank	Filler

Chapter 6: Reactivate Accidentally Scratched Virtual Volume

This section contains the following topics:

Reactivate Accidentally Scratched Virtual Volume (see page 43)

Reactivate Accidentally Scratched Virtual Volume

When a Virtual Volume is scratched, CA Cloud Storage for System z renames the file on the Linux Server. Three days after the volume is scratched, executing the cacloud scr_sync command deletes these files from the Linux Server. You can reactivate an accidentally scratched Virtual Volume up until the point its file is permanently deleted from the Linux Server.

To reactivate a scratched Virtual Volume, follow these steps:

1. Issue the SVT*n* X vol_info volser* console command to determine what versions of the scratched VOLSER remain. If no versions are returned, the Virtual Volume cannot be reactivated. If one or more versions are returned, use the scratch date in the file name to identify the required version.

   If the Virtual Volume has been reused, an active version is returned also:

   ```
   SVT4X2205I cacloud vol_info 105979*  481
   Scanning /var/lib/cacloud/vault_01/vv_*/105979*
   2.0G vv_105/105979.vve                  <- Active
   2.0G vv_105/105979.scr-20131017-215934  <- Scratch
   ```

2. Reactivate the Virtual Volume, if the required scratch version still exists.

3. Copy the data set to another Virtual Volume to free up the needed VOLSER, If an active version of the Virtual Volume exists.

   Note: After copying the data sets to a new Virtual Volume, you may need to recatalog them.

4. Scratch the needed VOLSER in the Global VCAT by executing the SVTSUTIL batch command VVE_SCRATCH=*volser* where *volser* is the Virtual VOLSER.

   Note: Now the Linux Server has two scratch versions of this VOLSER.

   ```
   SVT4X2205I cacloud vol_info 105979*  481
   Scanning /var/lib/cacloud/vault_01/vv_*/105979*
   2.0G vv_105/105979.scr-20131018-100738  <- Now scratched
   2.0G vv_105/105979.scr-20131017-215934  <- Scratch
   ```

5. Log on to the Linux system and enter the following command to change to the appropriate directory:

   `cd /var/lib/cacloud/vault_01/vv_`*`xxx`*

   Where *xxx* is the first three characters of the VOLSER.

6. Enter the following command to display the scratched Virtual Volume files:

   `ls -lh `*`volser`*`*`

   Where *volser* is the Virtual VOLSER.

7. Enter the following command to rename the scratch Virtual Volume file to an active Virtual Volume file:

   `sudo mv `*`volser`*`.scr-`*`yyyymmdd-hhmmss`* `volser.vve`

   Where *volser* is the Virtual VOLSER, *yyyymmdd* is the date, and *hhmmss* is the time of the appropriate scratched version of this VOLSER.

8. Update the tape management system record for the reactivated Virtual Volume If needed, so that it reflects the correct data set name and other information. If the record is in scratch status, update it with an expiration date in the future to reactivate it.

Note: When a reactivated Virtual Volume is displayed in the ISPF Interface (SVTSMON), no data set name, group number, creation date, or other information is displayed. This information is not required for CA Vtape to read the Virtual Volume successfully and provide the application with its data set.

Chapter 7: Backing Up and Restoring CA Vtape Control Data Sets

This section contains the following topics:

Back Up Control Data Sets (see page 45)
Restore or Recover CA Vtape Control Data Sets (see page 47)

Back Up Control Data Sets

CA Vtape control data sets are critical system files. Keeping these files on a high performance, reliable DASD, helps improve overall performance and minimize potential system outages.

The z/OS control data sets for a CA Vtape Complex must be available at your disaster recovery site. These data sets consist of:

Global VCAT

The Global VCAT contains multisystem control information about Virtual Volume status. A hardware reserve is used to serialize access to the Global VCAT. This can require you to isolate this data set on a single DASD volume.

All CA Vtape subsystems participating in a CA Vtape Complex share this file.

The Global VCAT is not backed up for disaster recovery purposes. Instead, it is recovered from the BSDS1 control data set.

BSDS (the Bootstrap Data Set)

The BSDS mirrors similar information that is stored in the Global VCAT. This allows either data set to be recovered from the other. Place these data sets on separate DASD volumes, preferably in separate DASD subsystems so that they do not end up on the same physical HDA. This ensures that a single hardware failure does not damage both data sets.

All CA Vtape subsystems participating in a CA Vtape Complex share this file.

Back up the BSDS regularly for disaster recovery purposes.

Local VCAT

The Local VCAT file records information specific to a single CA Vtape Subsystem instance. Each subsystem must have a unique Local VCAT. This data set is used primarily as a work area for transient information and to contain the attribute settings that are loaded from parmlib.

For disaster recovery, define a new Local VCAT.

The typical actions at a disaster recovery site are to:

1. Define and initialize a new Local VCAT.
2. Restore the BSDS from the most current backup.
3. Recover the Global VCAT from the restored BSDS.

Back Up BSDS

Back up the BSDS at about the same time as your tape management database. Taking backups of these data sets close together ensures that Virtual Volume information that is stored in the BSDS matches information that is stored in the tape management database. A synchronized set of backups is not required. If these backups are too far out-of-sync, it can complicate your disaster recovery efforts. CA Vtape can require manual intervention to access Virtual Volume files.

Back up the BSDS with CA Disk, DFSMSdss, FDR, or any other backup product that implements concurrent or snapshot copy. Concurrent or snapshot copy lets you create backups without having to stop CA Vtape or suspend tape processing.

Maintain backups on DASD or on physical tapes that provide access to the backup data set even when CA Vtape is inoperative. Do not back up the BSDS to media that requires CA Vtape to be active to restore it.

To ensure that a media failure does not prevent restoring the most current Virtual Volume information:

- Use RAID devices for a backup to DASD.
- Duplex a backup to tape.

Sample of a CA Disk backup job:

```
//JOBNAME   JOB …
//BACKUP    EXEC DMS,MI=002
//SYSIN     DD *
  FIND      DSN=HLQ.BSDS1
  BACKUPCC  RETPD=14
```

Sample of a DFSMSdss backup job:

```
//JOBNAME       JOB …
//BACKUP        EXEC PGM=ADRDSSU,REGION=4M
//SYSPRINT      DD SYSOUT=*
//INPUT         DD DISP=SHR,DSN=HLQ.BSDS1
//OUTPUT        DD DISP=(,CATLG),DSN=HLQ.BSDS.BACKUP(+1),
//LABEL=(1,SL),UNIT=REAL3490
//SYSIN         DD *
    DUMP DATASET(INCLUDE(HLQ.BSDS1)) -
    PHYSINDDNAME(INPUT) -
    OUTDDNAME(OUTPUT) -
    CANCELERROR OPTIMIZE(4)  CONCURRENT WAIT(20,20) -
    TOL(ENQF)
//
```

Restore or Recover CA Vtape Control Data Sets

This section describes how to restore the control data sets from a backup and how to recover the Global VCAT and BSDS1 from each other.

Recover BSDS from Global VCAT

The BSDS mirrors information that is stored in the Global VCAT. You can use an intact Global VCAT to recover a damaged BSDS data set. Mirroring the Global VCAT lets you restore the BSDS to a state that is more consistent with the information stored in the tape management database.

Follow these steps:

1. Bring down all the CA Vtape subsystems that are part of the CA Vtape complex sharing the BSDS you want to recover.
2. Customize and submit the RECBSDS member from the HLQ.CCUUJCL library. The job follows these steps:
   a. Deletes the old BSDS.
   b. Defines a new BSDS using the Global VCAT as a model.
   c. REPROs the Global VCAT into the new BSDS.
3. Start one CA Vtape subsystem in the CA Vtape complex and execute the console command SVTn D A. This command displays the correct scratch totals and devices.
4. Start the remaining CA Vtape subsystems in the CA Vtape complex.

Restore BSDS from Backup

If both the BSDS and the Global VCAT are corrupted or lost, restore the BSDS from your latest backup. After it is restored, use the BSDS data set to recover the Global VCAT.

Shut down all the CA Vtape Subsystems that are part of the CA Vtape Complex sharing the BSDS being recovered, before recovering the BSDS.

Sample CA Disk restore job

```
//JOBNAME  JOB …
//RESTORE  EXEC RESTORE
//SYSIN    DD *
  RESTORE  DSN=HLQ.BSDS1
//
```

Sample DFSMSdss restore job

```
//JOBNAME       JOB …
//RESTORE       EXEC PGM=ADRDSSU,REGION=4M
//SYSPRINT      DD SYSOUT=*
//INPUT         DD DISP=SHR,DSN=HLQ.BSDS.BACKUP(+0)
//OUTPUT        DD DISP=SHR,UNIT=3390,VOL=SER=XXXXXX
//SYSIN         DD *
  RESTORE DATASET(INCLUDE(HLQ.BSDS1)) -
    PHYSINDDNAME(INPUT) -
    OUTDDNAME(OUTPUT) -
    REPLACE
//
```

Recover Global VCAT from BSDS

If the BSDS is available and intact, use it to recover the Global VCAT.

Follow these steps:

1. Bring down all the CA Vtape subsystems that are part of the CA Vtape complex sharing the Global VCAT to be recovered.
2. Customize the sample JCL found in HLQ.CCUUJCL(RECGLVC).

   If the Global VCAT is not accessible, the first step of the job is required to define a new Global VCAT. If the Global VCAT is accessible, the first step of the job can be commented out and NOINIT can be added to the recovery commands in the second step to reduce runtime.

   RECOVER=GLOBAL,...,NOINIT

   Note: For information about the RECOVER=GLOBAL command, see the chapter "SVTSUTIL Batch Commands."

3. Submit the customized RECGLVC job to perform the recovery of the Global VCAT from the BSDS. Carefully review the job return codes and messages for any errors and take the appropriate corrective action.
4. Start one CA Vtape subsystem in the CA Vtape complex and execute the console command SVT*n* D A. The correct scratch totals and devices should be displayed.
5. Start the remaining CA Vtape subsystems in the CA Vtape complex.

Redefine Local VCAT

Define new Local VCATs for each CA Vtape VTS Subsystem in the CA Vtape VTS complex.

Follow these steps:

1. Edit the HLQ.CCUUJCL data set member DEFVCAT.
2. Add a valid JOB statement and follow the comments in the JCL member to complete its customization.
3. Submit the member to define and initialize a Local VCAT for one CA Vtape subsystem.
4. If the job ends with a nonzero return code, take the necessary corrective action and resubmit the job.
5. Repeat steps 1 through 4 for each CA Vtape subsystem that is started.

Chapter 8: Using Bidirectional Replication for Disaster Recovery

This section contains the following topics:

Exploit Bidirectional Replication for Disaster Recovery (see page 51)
Bidirectional Replication Example (see page 52)
Bidirectional Replication Data Flow (see page 53)
Bidirectional Replication Configuration (see page 54)
Configure parmlib (see page 55)
Configure NFS Exports (see page 55)
Create Subdirectories for NFS Export Mounting (see page 56)
Update FSTAB Entries to Map the Mount Points to NFS Exports (see page 57)
Use cacloud vol mkdir to Define VOLSER Ranges to Linux Systems (see page 58)
Test Configuration (see page 59)

Exploit Bidirectional Replication for Disaster Recovery

Bidirectional replication lets you mirror the disk storage changes between a primary and a secondary appliance. This feature is available in many appliances. You can also emulate a bidirectional replication using two pairs of unidirectional replication appliances. Depending on the storage vendor, each appliance can take on the role of primary for some files and secondary for others.

CA Cloud Storage for System z (CS4z) can take advantage of bidirectional replication for disaster recovery purposes. Appliances that support bidirectional replication typically allow the NFS exports at the secondary site to be mounted read-only by other Linux systems. You can use this capability to read Virtual Volumes that are created at either site.

The topics in this section provide an example that demonstrates how you can configure CS4z to exploit bidirectional replication for disaster recovery.

Bidirectional Replication Example

This example describes how you can exploit a bidirectional replication. Use CA Cloud Storage for System z to write Virtual Volumes at one data center and read them at another.

Assume that you have two separate CA Cloud Storage for System z complexes running and configured at a production data center (PDC) and disaster recovery center (DRC):

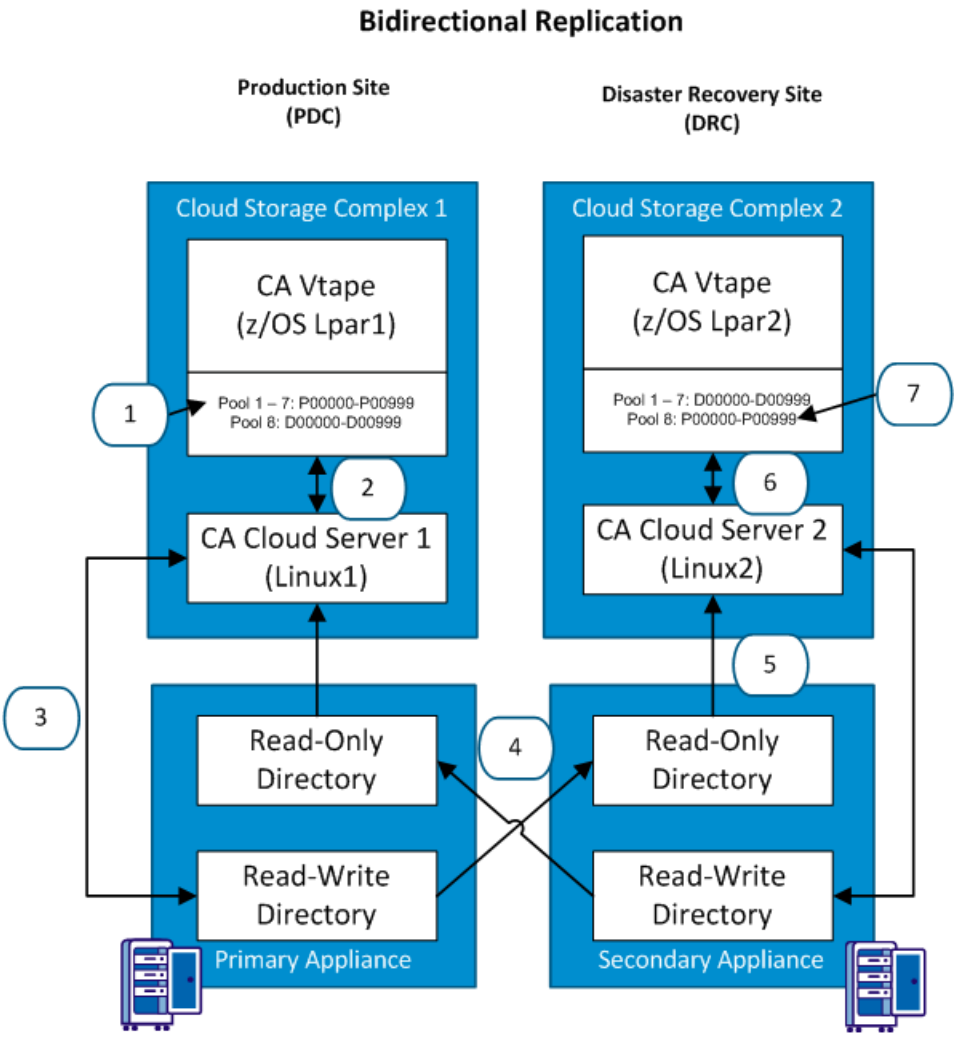
- Configure the PDC to write Virtual Volumes to a locally attached disk storage appliance, the primary appliance.
- Configure the DRC to write Virtual Volumes to its own locally attached disk storage appliance, the secondary appliance.
- Configure the NFS exports on the primary appliance to replicate changes to the secondary appliance
- Configure the NFS exports on the secondary appliance to replicate changes to the primary appliance.

To accomplish bidirectional replication:

- Use a pair of EMC Data Domain appliances with two MTree-replicated directories.
- Use a pair of NetApp appliances with SnapMirror technology.
- Use four Riverbed SteelStore appliances which support unidirectional replication.

Bidirectional Replication Data Flow

This example shows the bidirectional replication data flow from the PDC to the DRC.



The data flow process is:

1. (PDC) CA Vtape services a scratch mount request with a Virtual Volume. For example, VOLSER P00001.
2. (PDC) The volume is written to the CA Cloud Connector Vault Directory on Linux.
3. (PDC) The volume data is stored on the mounted read/write directory on the Primary Appliance.
4. (PDC/DRC)The Primary Appliance replicates the data to the read/only directory on the Secondary Appliance.

5. (DRC) With the read/only directory on the appliance mounted to the CA Cloud Connector Vault Directory, the volume data is now available.
6. (DRC) CA Vtape VTS can access the data from the CA Cloud Connector Vault Directory.
7. (DRC) CA Vtape VTS can now mount the Virtual Volume (VOLSER P00001) for read requests.

Bidirectional Replication Configuration

You begin by isolating the VOLSER ranges that each CA Vtape VTS complex use. You want each complex to read volumes that the other complex creates. For ease of identification each complex uses a separate and unique VOLSER range.

In the CA Vtape VTS parameter library, you:

- Define the VOLSER ranges in a pool.
- Define the pool name to a group number.
- Define the group number in a data set name or SMS data class filter.

When a scratch mount is requested, the matching filter determines the group policy. The group policy contains the pool name, which controls the volser range that is used for the scratch mount.

The VOL_MKDIR command lets you associate volser ranges with mount points and defines which volumes are placed on which mount points. Through FSTAB on Linux, you define what those mount points are.

Configure parmlib

As an example, you configure the parmlib as follows:

PDC	DRC
<DataSetFilters>	<DataSetFilters>
IncludeDS=IncludeDS1	IncludeDS=IncludeDS1
<IncudeDS1>	<IncludeDS1>
Group21='PROD./'	Group31='PROD./'
<Group21>	<Group31>
Description = ' Used for PDC files'	Description = ' Used for DRC files'
Pool=Pool1	Pool=Pool1
; Volume pools	; Volume pools
<Pool1> ; PDC files	<Pool1> ; DRC files
Volser=P00000-P00999	Volser=D00000-D00999
<Pool8> ; DRC files	<Pool8> ; PDC files
Volser=D00000-D00999	Volser=P00000-P00999

You configure parmlib so that the PROD files created at the PDC are directed to the P00000-P00999 VOLSER range. These same PROD files that are created at the DRC are directed to the D00000-D00999 VOLSER range. Next, you map the VOLSER ranges to NFS exports that replicate between the primary and secondary appliances.

Configure NFS Exports

The administration and configuration software varies with the appliance, so we do not provide specific steps to define the replicated NFS exports. For information about setting up replication for your appliances, see your appliance documentation.

As an example, you perform the following steps:

1. Define the NFS exports on the primary appliance so that the PDC can create Virtual Volumes. The primary appliance is locally available to the Linux System at the PDC.
2. Create:
   a. An NFS export for PROD./ files with a name like pdc_mp21 for Production Data Center Group 21 Virtual Volumes mount point. This NFS export is mounted as read/write on the Linux system.
   b. Configure the NFS export to replicate data to the secondary appliance.
   c. An NFS export for PROD./ files with a name like drc_mp31 to match the name of the NFS export from the secondary appliance. This NFS export is mounted as read only on the Linux system.
3. Define the NFS exports on the secondary appliance so that the DRC creates Virtual Volumes. The secondary appliance is locally available to the Linux System at the DRC.
4. Create:
   a. An NFS export for PROD./ files with a name like drc_mp31 for Disaster Recovery Center Group 31 Virtual Volumes mount point. This NFS export is mounted read/write on the Linux system.
   b. Configure the NFS export to replicate data to the primary appliance.
   c. An NFS export for PROD./ files with a name like pdc_mp21 to match the name of the NFS export from the primary appliance. This NFS export is mounted as read only on the Linux system.

Create Subdirectories for NFS Export Mounting

Create the subdirectories in the CA Cloud Connector Virtual Vault at both sites as mount points for the NFS exports that you configured.

To create these subdirectories, issue the Linux mkdir command.

For example, you issue the following commands on each system:

```
sudo mkdir /var/lib/cacloud/vault_01/mp_21
sudo mkdir /var/lib/cacloud/vault_01/mp_31
```

Note: mp_21 is used for PDC Virtual Volumes at both sites. mp_31 is used for the DRC Virtual Volumes at both sites.

Update FSTAB Entries to Map the Mount Points to NFS Exports

Configure the FSTAB entries at each site to map the NFS exports to the directory.

Follow these steps:

1. Issue the CA Cloud Storage for System z Linux command to configure the FSTAB entries:

   ```
   sudo cacloud setup
   ```

2. Select the FSTAB menu option and then map the mount points using the table that is provided at the end of this procedure.

3. Issue the following Linux commands to mount, unmount, or display the NFS appliance directories:

   - To mount the NFS exports that are based on the current FSTAB entries:

     ```
     sudo mount –a
     ```

   - To unmount specific directories:

     ```
     sudo unmount /path-to-directory
     ```

   - To display the FSTAB entries that are currently mounted:

     ```
     mount
     ```

4. Verify that the FSTSAB entries are defined correctly at both sites, then reboot the Linux systems using the Linux command:

   ```
   sudo /sbin/reboot
   ```

5. Verify that the FSTAB entries are again properly mounted on each system using the Linux mount command in Step 3.

For Step 2, use the following table:

PDC FSTAB Entries	DRC FSTAB Entries
This mount point id used at the PDC for production files that replicate to the secondary appliance located at the DRC. It is mounted read/write in FSTAB and contains the Virtual Vault subdirectories. *primary_url*:/../pdc_mp21 /var/lib/cacloud/vault_01/mp_21 nfs rsize=131072,size=131072,nfsvers=3,intr,bg	The DRC uses this mount point to read volumes on the secondary appliance that were replicated from the PDC. It is mounted read-only in FSTAB and contains the Virtual Vault subdirectories that the PDC creates and replicates. *secondary_url*:/../pdc_mp21 /var/lib/cacloud/vault_01/mp_21 nfs ro,rsize=131072,size=131072,nfsvers=3,intr,bg

PDC FSTAB Entries	DRC FSTAB Entries
The PDC uses this mount point to read volumes on the primary appliance that are replicated from the DRC. It is mounted read-only in FSTAB and contains the Virtual Vault subdirectories created and replicated by the DRC. *primary_url*:/../drc_mp31 /var/lib/cacloud/vault_01/mp_31 nfs ro,rsize=131072,size=131072,nfsvers=3,intr,bg	This mount point is used at the DRC for production files that replicate to the target appliance at the PDC. It is mounted read/write in FSTAB and contains the Virtual Vault subdirectories. *secondary_url*:/../drc_mp31 var/lib/cacloud/vault_01/mp_31 nfs rsize=131072,size=131072,nfsvers=3,intr,bg

Use cacloud vol_mkdir to Define VOLSER Ranges to Linux Systems

Now that the NFS exports are mounted, issue the CA Cloud Storage for System z Linux command to define VOLSER ranges to the mount points on each of the Linux Systems.

Follow these steps:

1. On the PDC Linux system, issue the following command to define the VOLSER range directory and create a soft link to the mount point:
   ```
   sudo cacloud vol_mkdir mp_21/vv_p00
   ```
2. On the DRC Linux system, issue the following command to define the VOLSER range directory and create a soft link to the mount point:
   ```
   sudo cacloud vol_mkdir mp_31/vv_d00
   ```
3. On both systems, issue the following CA Cloud Storage for System z Linux command:

   ```
   sudo cacloud vol_mkdir --fix
   ```

   Note: The EMC Data Domain MTree replication uses periodic point-in-time snapshots to replicate changes between appliances. This technique is designed to reduce discrepancies in disk changes. However, because transfers occur periodically, the source and target appliances can become asynchronous between replications.

Test Configuration

After you set up the two CA Vtape VTS systems in a bidirectional configuration, run tests to verify that Virtual Volumes are replicated and accessible between the primary and secondary appliances:

1. Run a job at the PDC that allocates a scratch Virtual Volume for a data set like PROD.TEST. CA Vtape VTS mounts a VOLSER in the P00000-P00999 range on the primary appliance. At some point after the tape is unmounted the disk storage appliance replicates the Virtual Volume to the secondary appliance.
2. After the Virtual Volume replication completes, run a job at the DRC that reads the VOLSER from the secondary appliance.
3. Submit the same job at the DRC. The job mounts a Virtual Volume in the D00000-D00999 range on the secondary appliance. At some point, after the tape is unmounted, the disk storage appliance replicates the Virtual Volume to the primary appliance.
4. Run a job at the PDC that reads the VOLSER from the primary appliance, after the Virtual Volume replication completes.

Note: If the user catalogs are not mirrored or shared between the systems, code the VOLSER in the JCL to mount it.

Chapter 9: Recovering CA Vtape Subsystem

When recovering a CA Vtape Subsystem, ensure that the Virtual Volume data has been replicated to the cloud or a secondary network appliance.

This section contains the following topics:

Prepare for Disaster Recovery (see page 61)
Recover at the Disaster Recovery Data Center (see page 62)

Prepare for Disaster Recovery

At the primary data center (PDC) backup the product libraries, including the parmlib, the tape management system catalog, and the BSDS.

Consider the following backup issues when planning for disaster recovery:

- Back up copies of the product libraries, including parmlibs, the tape management system catalog, and the BSDS1 control data set.
- Back up the BSDS control data set and the tape management catalogs at relatively the same time, to keep them synchronized. You do not have to stop CA Vtape to perform the BSDS backup.
- Schedule these backups to run as often as your business requires. At a minimum, run the backup once a day after the disaster recovery data is created. Note the date and time when the backup started.

The backups provide point in time snapshots of control information about the Virtual Volumes for your DRC. Virtual Volumes that are scratched or modified after the backups are made are not recorded in the tape management system or CA Vtape control data sets.

Recover at the Disaster Recovery Data Center

The recovery process restores the CA Vtape control data sets and the tape management system back to the point in time when the last available backup was created. The following elements must be available at the disaster recovery location:

- Access to replicated networked devices containing Virtual Volume files.
- A Linux Server with the CA Cloud connector installed and properly configured.
- Latest copy of the backups that are performed at the PDC. See Prepare for Disaster Recovery (see page 61).

Follow these steps:

1. Restore the product libraries and the parmlib.
2. Define and initialize the Local VCAT by submitting job DEFVCAT located in HLQ.CCUUJCL.

   Note: The SVTPARMS DD must reference a data set that contains an SUTPARMS member. The SUTPARMS member must contain the CA Vtape DSN prefix and Virtual Volume Size information that is used to configure the PDC installation.
3. Restore the BSDS and perform all the steps in the section Recover the Global VCAT (see page 48).
4. Verify that the SVTS JCL procedure is located in your DRC proclib. If not, copy it from HLQ.CCUUPROC and customize it.
5. Review the OSDRIVE parmlib member and verify that the Virtual Device addresses conform to the ones defined at the DRC.
6. Review the OSPOOLS parmlib=b member and ensure that it is configured correctly for your DR requirements.

   For example, you can edit the member and can move the VOLSER ranges defined in Pool1-7 to Pool8. The VOLSER ranges in Pool8 may need to be moved to Pool1-7. This reserves volser ranges for DR recovery. Volser ranges that are placed in Pool8 are not used for scratch mount requests.
7. Review the OSPARMS parmlib member and verify that the *PipeUnitAddress* attribute addresses conform to the ones defined for the CA Cloud Connector DR location.
8. If the CA Cloud Connector is not installed, download it and install the cacloud and ctcl rpms on your Linux for System z system.
9. Start the CA Cloud Storage for System z setup tool by issuing the command:

   ```
   cacloud setup
   ```
10. Select CONFIG from the Main Menu, and page down to the CTC definitions. If needed, update the definitions to match the definitions that are defined at the DRC.
11. Save your changes and leave the editor.

12. Select QUIT from the Main Menu to leave Setup.
13. Start the Linux Server by issuing the command:

    ```
    cacloud start
    ```

14. Start CA Vtape by issuing the command:

    ```
    S SVTSCE
    ```

15. Restore the tape management system catalog.
16. You can now submit jobs requiring Virtual Volumes.

Chapter 10: Troubleshooting

This chapter describes problems and conditions that you can solve using the CA Vtape console or batch commands. If you cannot completely resolve the condition, follow the additional steps in this chapter to gather debugging documentation and to protect the system.

This section contains the following topics:

Virtual Devices Will Not Vary Online (see page 65)
Virtual Volume Shortage (see page 66)
SO8B Abends (see page 66)
GTF Trace (see page 67)
IPCS (see page 67)
Self-Documenting Error Recovery Routines (see page 70)

Virtual Devices Will Not Vary Online

Several problems can occur when you try to vary the Virtual Devices online. Two of the most common are as follows:

- If the Virtual Devices were not properly defined with HCD, the following message is issued:

  ```
  IEE313I nnn              UNIT REF. INVALID
  ```

  Review the HCD definitions and correct the errors.

  Also verify within IPCS. From option 6, issue the following command:

  ```
  SETD ACTIVE
  LISTUCB ucb
  ```

  where *ucb* is the device address you are investigating.

  In the UCBXPX at offset +0C, you find the SIDA and SCHNO fields. If the values in these fields are 0000s, then CA Vtape successfully allocates this address. If these fields contain anything other than 0000s, then you need to define another range.

CA Vtape uses this safety mechanism to ensure that no physical devices are associated with the UCB. After CA Vtape uses these devices, other resources can use the UCB only after the next IPL.

- If you try to vary the Virtual Devices online without the SVTS STC active, you get the following message:

  IEE025I UNIT *nnn* HAS NO LOGICAL PATHS

If you do not define enough Virtual Volumes, CA Vtape displays the SDSF SYSLOG Scratch Shortage message, as shown in the following example:

```
 Display  Filter  View  Print  Options  Help
-------------------------------------------------------------------------------------------------------------------------------
 SDSF SYSLOG   760.102 P390 P390 12/10/1998 LINE   4,489     COLUMNS   1 132
 COMMAND INPUT ===> _                                        SCROLL ===> CSR
N 4000000 P390     98344 16:45:50.39 JOB00862 00000090  $HASP373 ISPCNT1A STARTED - INIT    C - CLASS C - SYS P390
N 0000000 P390     98344 16:45:50.45 JOB00862 00000081  IEF403I ISPCNT1A - STARTED - TIME=16.45.50
N 2000000 P390     98344 16:45:50.84 JOB00862 00000090 *IEF233A M 0570,PRIVAT,SL,ISPCNT1A,IEBGENER,ISPCNT3.S3480.VTAPE.GENZV5
W 8000000 P390     98344 16:46:11.80 STC00859 00000090 *07 SVTSV0424W 0570,Scratch shortage, Reply R(etry) or C(ancel)
N 0000000 P390     98344 16:47:00.15 TSU00852 00000290  IEA630I  OPERATOR ISPCNT3  NOW ACTIVE,   SYSTEM=P390    , LU=NMMAF001
NC0000000 P390     98344 16:47:00.57 ISPCNT3  00000290  SVTS ADD VVP=899000
NR0000000 P390     98344 16:47:03.10 TSU00852 00000090  SVTSX0100I Command Completed Successfully
NC0000000 P390     98344 16:47:38.35 ISPCNT3  00000290  R 07,R
NR8000000 P390     98344 16:47:38.42 ISPCNT3  00000090  IEE600I REPLY TO 07 IS;R
```

Virtual Volume Shortage

If you do not define enough Virtual Volumes for scratch processing, the system issues a SVT*n*V0424W *devn*, Pool *n*, Scratch shortage, Reply R(etry) or C(ancel) message.

Before replying, Retry the new scratch Virtual Volumes, and, assuming a scratch volume range is available in the Tape Management System, add that range to CA Vtape with the following console command:

SVT*n* ADD VVP=

For more information, see Console Commands.

Run the corresponding Scratch Synchronization job immediately to release as many Virtual Volumes as possible. For more information, see Scratch Tape Synchronization in the *Configuration Guide*.

S08B Abends

S08B abends are related to the IBM Data-In-Virtual (DIV) access method. These abends typically occur when accessing a control data set, such as the Global VCAT or Local VCAT, when the data set has not been initialized. An error message that indicates a problem may precede abends. Looking up the S08B reason code in the IBM System Codes manual allows you to relate the S08B abend to the preceding error messages.

GTF Trace

The Virtual Devices are provided with GTF Trace capability with output formatted exactly like a physical volume GTF trace.

IPCS

CA Vtape comes with an IPCS VERBX exit to use during problem determination. Customize and run the IPCS job that is distributed in HLQ.CCUUJCL to obtain detail information about control block structures residing in the Global and Local VCAT and the content of the Internal and External Logger.

IPCS Parameters to Print the Logger

Following is the syntax and option descriptions for running the Logger print engine that is implemented as part of the IPCS command.

```
VERBX SVTSIPCS '[SVTS(SVTn)] +
               [VOLUMES(ALL| NONE | ONLY(nnn))] +
               NOLOG | [LOG(see note)] | [LOGONLY(see note)]'
               PRINT
```

Valid Subcommands are as follows:

```
ACTIVE | NOACTIVE |,LEVEL(n) |,DDNAME(nnnnnnnn)
```

SVTSIPCS Operands

Operands are as follows:

SVTS(SVT*n*)

Specifies the subsystem ID number of the CA Vtape subsystem for which to display information. This value correlates to the last character of the CA Vtape subsystem (SVT*n*, where *n* must be between 1 through 8). The operand defaults to 1 which is the subsystem ID number of SVT1.

NOLOG

Specifies that no historical logging data be displayed.

VOLUMES

Specifies which Virtual Volumes to include in the report. ALL prints all Virtual Volumes, NONE does not print any, and ONLY(*nnn*) prints the Virtual Volumes that match the first three qualifying alphanumeric characters.

PRINT

Indicates the PRINT keyword that is part of IBM's IPCS VERBX command and if supplied writes to the IPCSPRNT DDNAME. The PRINT keyword must not be included within quotes. Output is written to the SYSTRPRT DDNAME, so specifying DUMMY helps to limit the output.

The following is an example of the IPCSPRNT DDNAME:

```
//IPCSPRNT DD DISP=(,CATLG),DSN=SVTS.TEMP.IPCS.PRINT,
// SPACE=(CYL,(5,5),RLSE),UNIT=SYSALLDA,
 // DCB=(LRECL=137,RECFM=VBA,BLKSIZE=23476)
```

LOG

Appends the Logger report to the regular IPCS report.

LOGONLY

Formats the Logger report exclusively.

ACTIVE

Indicates the default. Specifies that the IPCS SETDEF keyword (either a Dump or the Active system) controls the input to the Logger print engine.

NOACTIVE

When specified, instructs the Logger print engine to ignore the SETDEF keyword. If this option is not specified ACTIVE is generated.

LEVEL(*n*)

Overrides the detail level the Logger print engine formats. The value *n* must be between 1 through 3.

If LEVEL is not specified, all records, regardless of their Logger Detail Level, are included. Specify a lower detail level than the one that the data was recorded at filters all events that are generated for a Logger Level Detail greater than the supplied value. Specifying a greater detail level than the one that the data was recorded at has no effect. Level(3) may generate thousands of lines for only a few minutes of logging. Adjust the IPCSPRNT DD SPACE parameter accordingly.

DDNAME(*xxxxxxxx*)

When specified, indicates that the input to the Logger print engine is supplied by the specified DDNAME *xxxxxxxx* value. This parameter is independent of parameters ACTIVE/NOACTIVE and NOACTIVE and specified to process only input from the supplied DDNAME. The supplied *xxxxxxxx* value must match a DDNAME allocated to the Job Step that is pointing to a valid Log Stream name or a sequential data set containing extracted external Logger data.

The following is a typical example of how to code a Log Stream DDNAME using IBM's LOGR subsystem parameter to filter records based on date and time.

```
//LOGR     DD DISP=SHR,DSN=VTAPE.LOG,SUBSYS=(LOGR,,
 // FROM=(2002/302,12:00),TO=(2002/302,15:23),LOCAL'),
 // DCB=(RECFM=U,BLKSIZE=24576)
```

Examples of the IPCS command

```
VERBX SVTSIPCS PRINT
```

In this example, the IPCS print engine takes all the CA Vtape defaults and writes a copy of the output to the IPCSPRNT DDNAME. The options in effect are VOLUMES(ALL) and NOLOG.

```
VERBX SVTSIPCS 'LOGONLY(NOACTIVE,DDNAME(LOGR))'
```

In this example, the IPCS print engine generates the output that is based on the external Logger events pointed to by DDNAME LOGR and does not include the typical internal control blocks and Virtual Volumes data.

```
SETD DA('SYS1.P390.DMP00017') NOCONFIRM
VERBX SVTSIPCS 'LOGONLY(ACTIVE,LEVEL(1))' PRINT
```

In this example, the IPCS print engine looks in the dump 'SYS1.P390.DMP00017' for the internal Logger dataspace and formats all events at level 1. A copy of the output is written to the IPCSPRNT DDNAME.

```
SETD ACTIVE NOCONFIRM
VERBX SVTSIPCS 'LOGONLY'
```

In this example, the IPCS print engine writes the output to the SYSTRPRT DDNAME and only formats the data from the active running system's internal Logger dataspace.

```
SETD ACTIVE NOCONFIRM
VERBX SVTSIPCS 'LOG(ACTIVE)'
```

In this example, the IPCS print engine prints the traditional internal control blocks and all Virtual Volume information with data from the internal Logger dataspace. The input is the live system and output is written to the SYSTRPRT DDNAME.

Self-Documenting Error Recovery Routines

When an error is detected in any CA Vtape module, error recovery routines take place by issuing messages like SVT*nx*100E and SVT*nx*200E. This condition generates an SVC dump. The SVC dump is eligible to be suppressed by DAE (Dump Analysis and Elimination) and is generated during the first failure only. These self-documenting error recovery procedures are designed to simplify the error data collection process and ensure that the documentation is appropriately collected. As soon you detect that a dump is generated under these conditions, contact CA Support to report the error and receive the corresponding corrective maintenance.

Note: For online technical assistance and a complete list of locations, primary service hours, and telephone numbers, contact CA Support at http://ca.com/support.

Appendix A: Expanding and Converting Control Data Sets

This section contains the following topics:

Expand and Convert Control Data Sets (see page 71)

Expand and Convert Control Data Sets

To accommodate more than 510,800 Virtual Volume Entries in the Global VCAT and BSDS, modify the internal index. Change the index from pointing to individual records, to pointing to VSAM Control Intervals which contain sets of records.

The change in indexing occurs when the Global VCAT and BSDS are increased in size and the CONVERT parameter is coded on the EXPAND CDS,VOLUMES= command. The JCL found in HLQ.CCUUJCL(EXPAND) executes this command.

To index more than 510,800 VOLSERs, activate Volume Pooling. To activate Volume Pooling, update the Parmlib Directory Section to uncomment or add the VolumePoolDefinitions=OSPOOLS attribute. The OSPOOLS parmlib member is then updated with the VOLSER ranges in use by CA Vtape. If you do not know the VOLSER ranges in use, customize and execute HLQ.CCUUJCL(GENVOLPL) to create an OSPOOLS parmlib member with the current VOLSER ranges.

The control data sets are sized for the specific number of VOLSERs they currently index. To determine how large to allocate the new data sets, use the following formula:

7296 + current VOLSERs + additional VOLSERs = total records

For example, you have 500,000 VOLSERs in the Global VCAT and BSDS and are adding 50,000 more. The total number of records equals 7296 + 500,000 + 50,000 or 557296. Update the primary allocation amount to 557296 for the RECORDS parameters in the VSAM LDS DEFINEs in the EXPAND member.

The total records that are divided by 12 equals the number of 3390 tracks that are required to hold one of the control data sets. Following the example above, 557296 records would require 557296 / 12 or 46,441 tracks or 3096 cylinders of DASD for each control data set.

To perform the conversion, the EXPAND job dynamically invokes a recovery of the Global VCAT from the BSDS.

Follow these steps:

1. Calculate the number of records that are required for the new control data sets.
2. Find two DASD volumes with enough free space to allocate the new data sets.
3. Customize HLQ.CCUUJCL(EXPAND) by following the comments in the member:
   a. Add the CONVERT parameter to the EXPAND command.
   b. Update the Global VCAT DEFINE and the BSDS DEFINE RECORDS parameters to the calculated total number of records.
   c. Change the Global VCAT DEFINE VOLUME parameter to point to one of the DASD volumes from step 2.
   d. Change the BSDS DEFINE VOLUME parameter to point to the other volume.
4. Stop all the CA Vtape Subsystems that are sharing the Global VCAT and BSDS to be expanded and converted.
5. Execute the EXPAND JCL.
6. Start one of the CA Vtape Subsystems.
7. Perform any desired testing:
   a. Browse your Virtual VOLSER ranges with the CA Vtape ISPF Interface (SVTSMON CLIST).
   b. Read an existing Virtual Volume.
   c. Write to a scratch Virtual Volume.
8. Start the remaining CA Vtape Subsystems.

Appendix B: Health Checks

These topics describe health checks for the system. The owner for all the health checks is CA_VTAPE.

VTAPE_CDS_SEPARATION

Description

On different DASD volumes, allocate the Global VCAT, BSDS, and ICF Catalog data sets that CA Vtape uses. Allocate any combination of the data sets on the same volume to create a single point of failure, inhibit disaster recovery, but degrade performance due to contention.

Best Practice

The CA Vtape control data sets are critical system files. Keep these files on high-performance reliable media, such as a RAID-compliant DASD, to improve performance and minimize potential system outages. CA Vtape uses Hardware Reserve processing to serialize access to the Global VCAT. The need to perform volume Hardware Reserve may require you to isolate the Global VCAT on a single DASD volume. To ensure recoverability if the hardware unexpectedly fails, maintain current backups of the BSDS and keep the Global VCAT and BSDS on separate DASD volumes.

Allocating the ICF Catalog that CA Vtape uses or any other ICF catalog on the same DASD volume with the Global VCAT or the BSDS causes contention on the concurrent data set volume. This contention degrades performance and could lead to lock outs when other software products that scan catalogs or VTOCs are active in your environment.

We recommend that you relocate all CA Vtape control data sets to separate, high performance, RAID-compliant DASD volumes. Also allocate the CA Vtape control data sets in separate DASD subsystems. The allocation spreads the workload across separate channels and control units. The allocation maximizes performance, recovery, and availability.

Parameters Accepted

None

Debug Support

Yes

Verbose Support

Yes

Reference

For more information pertaining to the DASD placement and recovery of CA Vtape control data sets, see the chapters "Operational Considerations" in the *Configuration Guide* and "Recovering CA Vtape" in this guide.

Messages

SVTH1251E

A combination of the CA Vtape control data sets has been allocated on the same DASD volume. This is a single point of failure and could also degrade performance due to contention.

VTAPE_MODULE_CONSISTENCY

Description

The main address space and the sub address spaces are running with the same load modules, but at different maintenance levels. This can occur when:

- Executing the two PROCs with different loadlibs coded in the STEPLIB DD statements.
- One PROC has a STEPLIB DD coded and the other is using a loadlib in the LNKLST.

Maintenance has been applied to the loadlib and a single address space has been restarted instead of the entire CA Vtape subsystem.

Best Practice

Ensure that all CA Vtape address spaces are started with the same maintenance level library. Changing the CA Vtape Split Maintenance-Level Protection to run in AUTOMATIC mode is recommended. Split Maintenance-Level Protection is controlled by the TaskLib attribute in the Startup Options Section of parmlib.

Parameters Accepted

None

Debug Support

Yes

Verbose Support

Yes

Reference

For more information, see Split Maintenance-Level Protection (see page 19) in this guide and STARTUP OPTIONS in the *Configuration Guide*.

Messages

SVTH1301E

The CA Vtape address spaces are not all running with the same maintenance level.

VTAPE_PARM_zIIP_STATUS

Description

IBM zIIP specialty processors are installed and available for use yet CA Vtape is not currently configured to take advantage of these specialty processors.

Best Practice

Investigate and evaluate the use of zIIP specialty processors for CA Vtape. The use of zIIP specialty processors by CA Vtape increase performance and lower general processor usage.

The CA Vtape use of zIIP specialty processors is controlled by the zIIPExploitation and PercentRunOnZIIP attributes defined in the respective Startup Options and Dynamic Options Sections of the CA Vtape parmlib.

To enable CA Vtape exploitation of zIIP specialty processors, specify the zIIPExploitation attribute as zIIPExploitation=Y, and set the PercentRunOnzIIP attribute to a value greater than 0.

Parameters Accepted

None

Debug Support

Yes

Verbose Support

Yes

Reference

For more information, see the chapters "Operational Considerations" and "The Parameter Library (PARMLIB)" in the *Configuration Guide*.

Messages

SVTH4501I

CA Vtape is not configured to take advantage of the IBM zIIP specialty processors available on this system.

VTAPE_PARM_zIIP_CONFLICT

Description

The zIIPExploitation attribute indicates that SVTS uses the zIIP specialty processors to process virtual I/O activity. However, the PercentRunOnzIIP attribute indicates that no work is scheduled on those processors.

Best Practice

The use of zIIP specialty processors in a CA Vtape environment offers superior performance advantages over non-zIIP processor configurations. Evaluate the zIIPExploitation and PercentRunOnzIIP attributes defined in the respective Startup Options and Dynamic Options sections of the respective CA Vtape parmlib. Change the attributes to enable CA Vtape exploitation of zIIP if a zIIP specialty processor is installed.

To enable CA Vtape exploitation of zIIP, specify the zIIPExploitation attribute as zIIPExploitation=Y, set the PercentRunOnzIIP attribute to a value greater than 0, and restart CA Vtape.

To disable CA Vtape exploitation of zIIP, specify the attribute as zIIPExploitation=N and restart CA Vtape.

Parameters Accepted

None

Debug Support

Yes

Verbose Support

Yes

Reference

For more information, see the chapters "Operational Considerations" and "The Parameter Library (PARMLIB)" in the *Configuration Guide*.

Messages

SVTH4601I

CA Vtape has detected an inconsistency between the settings of the zIIPExploitation and the PercentRunOnzIIP attributes.

Appendix C: Performing Riverbed Disaster Recovery

This section contains the following topics:

How to Prepare and Perform a Riverbed SteelStore Appliance Cold Disaster Recovery (see page 79)
How to Prepare and Test a Riverbed SteelStore Warm Disaster Recovery (see page 85)

How to Prepare and Perform a Riverbed SteelStore Appliance Cold Disaster Recovery

A Riverbed SteelStore appliance is typically configured to replicate files to your cloud provider. You can optionally configure the appliance to perform an asynchronous replication to a dedicated secondary appliance at a remote location.

If you do not replicate files to a secondary appliance at a disaster recovery site, then a cold site recovery of the Riverbed SteelStore appliance is required. The basic steps in a cold recovery are:

1. Acquire and install a new appliance.
2. Import the configuration data (which is backed up from the primary appliance).
3. Download backups from the cloud provider to repopulate.

The *Best Practices Guide* provides disaster recovery recommendations.

Use the following scenario to guide you through the process:

1. Review Riverbed SteelStore Documentation for Cold Recovery (see page 80).
2. Prepare for disaster recovery (see page 80).
3. Perform a disaster recovery test (see page 81).
4. Perform the disaster recovery (see page 82):
    a. Log in to the appliance and repopulate the files (see page 83).
    b. Check for changes in the Linux server configuration (see page 83).
    c. Check softlinks that the Cloud Connector uses (see page 84).
    d. Test access after cold recovery (see page 85).

Review Riverbed SteelStore Documentation for Cold Recovery

Note: Riverbed changed the product name from *Whitewater* to *SteelStore.* Their documentation names may not reflect this change.

Review the following documentation to become familiar with the appliance and recovery on www.riverbed.com:

- *Whitewater Cloud Storage Appliance User's Guide*
- *Riverbed Whitewater Cloud Storage Gateway/Disaster Recovery Best Practices Guide*

Prepare for Cold Disaster Recovery

Note: Riverbed changed the product name from *Whitewater* to *SteelStore.* Their documentation names may not reflect this change.

Follow the disaster recovery preparation steps in the *Riverbed Whitewater Appliance User's Guide*. The Riverbed documentation provides instructions to export the appliance configuration data at your primary site. The exported file includes the license and encryption keys necessary to access data from your cloud provider. In your disaster recovery plans, include instructions for making the exported configuration file available at the disaster recovery location.

For a cold recovery, import the configuration file into the new SteelStore appliance. Once completed, you recover and repopulate the appliance from copies that the cloud provider maintains.

Perform a Cold Disaster Recovery Test

Note: Riverbed changed the product name from *Whitewater* to *SteelStore.* Their documentation names may not reflect this change.

This topic provides an overview of the *Whitewater Cloud Storage Appliance User's Guide* procedure for disaster recovery, but includes considerations. Because the Riverbed instructions may change with new releases, we recommend following the steps that are outlined in the Riverbed Whitewater documentation.

Test the disaster recovery of your appliances. Practicing the recovery process allows you to review procedures and policies and ensures that you are well-prepared for any unplanned outage.

Note: When you perform this test, the *Whitewater Cloud Storage Appliance User's Guide* instructs you to suspend replication to the cloud. This suspension simulates the loss of the primary appliance. After you complete the secondary appliance testing, unsuspend replication to ensure that the Virtual Volumes data resume replication to the cloud.

Follow these steps:

1. Suspend the replication to the cloud from the primary appliance.
2. Follow the procedures in the *Whitewater Appliance User's Guide* to import the primary SteelStore appliance configuration file into the secondary SteelStore appliance. Before starting the import, verify that the Import shared data only check box is selected.

   Important! Once the import process completes, the SteelStore system may display a prompt to restart the storage optimization service. Do *not* restart the storage optimization service.
3. Use SSH to connect to the SteelStore appliance command-line interface.
4. To perform replication recovery testing of the appliance, issue the following commands:

   ```
   whitewater > enable
   whitewater # configure terminal
   whitewater # no service enable
   whitewater # datastore format local
   whitewater # replication dr -test enable
   whitewater # service enable
   whitewater # show service
   ```

   Note: Riverbed changed the product name from *Whitewater* to *SteelStore*. The commands are customizable and may not reflect this change. For more information about these commands, see the *Whitewater Cloud Storage Appliance Command-Line Interface Reference Manual.*

Perform Cold Disaster Recovery

Note: Riverbed changed the product name from *Whitewater* to *SteelStore.* Their documentation names may not reflect this change.

This topic provides an overview of the *Whitewater Cloud Storage Appliance User's Guide* procedure for disaster recovery, but includes considerations. Because the Riverbed instructions may change with new releases, follow the steps in the Riverbed Whitewater documentation.

If the primary SteelStore device is not connected to the cloud provider, configure a new SteelStore appliance as its replacement. Riverbed describes this as a cold site replacement. Riverbed indicates that disaster recovery time can vary from a few seconds to a few hours. The time depends on how many files are stored on the appliance or at the cloud provider.

Follow these steps:

1. Follow the procedures in the *Whitewater Cloud Storage Appliance User's Gu*ide to import the primary SteelStore appliance configuration file into the secondary SteelStore appliance. Before importing, verify that the Import shared data only check box is selected.

   Important! Once the import process completes, the SteelStore system may display a prompt to restart the storage optimization service. Do not restart the storage optimization service.

2. Use SSH to connect to the SteelStore appliance command-line interface.

3. To perform replication recovery of the appliance, issue the following commands:

   ```
   whitewater > enable
   whitewater # configure terminal
   whitewater # no service enable
   whitewater # datastore format local
   whitewater # replication recovery enable
   whitewater # service enable
   whitewater # show service
   ```

   Consider the following:

   - Riverbed changed the product name from *Whitewater* to *SteelStore*. The commands are customizable, but may not reflect this change. For more information about these commands, see the *Whitewater Cloud Storage Appliance Command-Line Interface Reference Manual.*
   - Use the "show service" command to determine when replication should complete.

Log in to Appliance and Repopulate Files

Once the replication recovery completes, log in to the SteelStore appliance using the browser interface and begin repopulating files on the appliance.

Check for Changes in Linux Server Configuration

As the files are repopulated on the SteelStore appliance, log in to the Linux server and review the Linux file system table. Linux uses the FSTAB (or /etc/fstab) to map disk appliances as subdirectories of the Linux file system. FSTAB is used to map the NFS exports of the SteelStore appliance to subdirectories used as mount points for Virtual Volume data.

Because you are configuring a new appliance, some items like the IP address may change. Review FSTAB if there are changes in this area.

Follow these steps:

1. SSH to the Linux server and issue the cacloud setup command to edit or modify the FSTAB entries:

   ```
   sudo cacloud setup
   ```

2. Complete the changes.
3. Reboot the Linux system to ensure that the FSTAB changes take effect.
4. Issue the following commands to verify that your FSTAB entries are properly configured on the Linux server:

   - To display the mount points in effect issue:

     ```
     mount
     df –h
     ```

   - To display mount points that the Cloud Connector uses:

     ```
     cacloud mp_stats
     ```

   - To display volser ranges of mount points that are mapped to the Cloud Connector:

     ```
     cacloud mp_stats mp*/vv*
     ```

     The optional mp*/vv* parameter lets you display volser ranges of mount points that are mapped to the Cloud Connector.

Check Softlinks that Cloud Connector Uses

CA Cloud Storage for System z uses softlinks to map virtual volser ranges to mount points of its Virtual Vault on the Linux server. If you change FSTAB or the mount point subdirectories, you may need to correct the softlinks that point to them.

Follow these steps:

1. Issue the following command to change to the Virtual Vault subdirectory containing softlinks that the Cloud Connector uses:

   ```
   cd /var/lib/cacloud/vault_01
   ```

2. Verify that the softlinks are properly defined.

   Softlinks direct Virtual Volume volser ranges to mount points of the disk appliance. If the links are incorrect, you can correct the links manually. You can also remove broken softlinks then regenerate missing softlinks using cacloud or Linux commands. Use the following softlink commands:

   - To display the softlinks:

     ```
     ls –l vv_*
     ```

   - To remove broken softlinks:

     ```
     sudo rm ./vv_nnn
     ```

   - To direct cacloud to regenerate missing softlinks:

     ```
     sudo cacloud vol_mkdir –fix
     ```

   - To create new softlinks using cacloud:

     ```
     sudo cacloud vol_mkdir mp_nnn/vv_nnn
     ```

   - To create softlinks:

     ```
     sudo ls –s mp_nn/vv_nnn vv_nnn
     ```

Test Access after Cold Recovery

Perform preliminary testing to verify that CA Cloud Storage for System z has access to the recovered appliance.

Follow these steps:

1. Start the CA Vtape subsystem on z/OS.
2. Issue the following CA Vtape subsystem commands to check connectivity to the Cloud Connector and the recovered appliance:
   - To display connectivity with the cloud connector:

     ```
     SVTn linux ping
     ```

   - To display the mount points of the cloud connector:

     ```
     SVTn linux mp_stats
     ```

   Note: For more information about these commands, see the *Linux Installation and Configuration Guide*.
3. Test read access to recovered Virtual Volumes. Run z/OS jobs that read repopulated Virtual Volumes.
4. To test write access, run jobs on z/OS that write new scratch volumes to the recovered appliance.
5. Consider more testing to confirm that the basic functionality of CA Cloud Storage for System z is restored using the new appliance.

   Cold Recovery of the SteelStore appliance is complete.

How to Prepare and Test a Riverbed SteelStore Warm Disaster Recovery

A Riverbed SteelStore warm disaster recovery uses two SteelStore appliances (a primary and a secondary) in a peer replication configuration.

In peer replication, the primary appliance replicates the data and metadata to the cloud provider and also to the secondary appliance. You can use the primary appliance for both backup and access to the stored data. Use the secondary appliance for read-only access. You can mount the NFS exports of the primary using read-write access on the Linux server. You can mount the secondary for read access only.

Warm disaster recovery using peer replication provides a faster disaster recovery time. The secondary appliance provides immediate read access to data without the delays that are required to repopulate the secondary appliance from the cloud. The *Best Practices* provides the disaster recovery recommendations. Planning for Riverbed Cold Disaster Recovery (see page 80) continues to apply even when using peer replication.

The basic steps to configure Riverbed SteelStore for peer replication are:

1. Review the SteelStore documentation (see page 86).
2. Prepare for disaster recovery (see page 86).
3. Isolate Virtual Volume Volser Ranges (see page 87).
4. Configure appliances for peer replication (see page 89):
   a. Configure the primary appliance peer replication (see page 89).
   b. Configure the secondary appliance peer replication (see page 89).
   c. Complete primary appliance configuration (see page 90).
5. Review the Peer Replication disaster recovery scenarios (see page 91).

Review SteelStore Riverbed Documentation for Warm Recovery

The Riverbed SteelStore documentation provides a warm disaster recovery model. The topology uses two SteelStore appliances in a paired primary and secondary configuration for peer replication.

Review the following documentation to become familiar with the appliance and recovery on www.riverbed.com:

- *Whitewater Cloud Storage Appliance User's Guide*
- *Whitewater Cloud Storage Gateway/Disaster Recovery Best Practices Guide*

  Note: Riverbed changed the product name from *Whitewater* to *SteelStore.* The document names may not reflect this change.

Prepare for Warm Disaster Recovery

Follow the disaster recovery preparation steps in the *Whitewater (SteelStore) Cloud Storage Appliance User's Guide*.

The Riverbed documentation provides instructions to export the appliance configuration data at your primary site. If you want to perform a warm disaster recovery (see page 85), the exported file includes the license and encryption keys necessary to access data from your cloud provider. In your disaster recovery plans, include instructions for making the exported configuration file available at the disaster recovery location.

Isolate Virtual Volume Volser Ranges

Using different virtual volser ranges for production and Disaster Recovery testing is an important consideration in warm recovery. Planning makes identifying test volumes to delete at the end of your DR tests easier.

- Use Volume Pools within the CA Vtape parameter library to identify test volumes.

  Up to eight Volume Pools can define virtual volser ranges for new scratch mount requests. If a virtual mount requires a new scratch tape CA Cloud compares the requesting allocated data set or SMS data class name against a set of filters. These filters assign the allocation to a CA Cloud Group. The Group has attributes for controlling how the Virtual Volume is processed.

- Assign Volume Pool1 through Pool7 to groups for new scratch mount requests.

  Volume Pool8 Volser ranges define the Virtual Volumes that the subsystem can mount, but cannot be used for new scratch mount requests.

Chapter 11: Configure Appliances for Peer Replication

This section contains the following topics:

Configure Primary Appliance Peer Replication (see page 89)
Configure Secondary Appliance Peer Replication (see page 89)
Complete Primary Appliance Configuration (see page 90)

Configure Primary Appliance Peer Replication

Stop the primary appliance optimization service and configure peer replication settings with the secondary appliance.

Follow these steps:

1. Use an internet browser to log in to the primary appliance Riverbed SteelStore UI.
2. Select Configure, Maintenance, Storage Optimization Service.
3. Stop the optimization service.
4. Select Configure, Storage, Peer Replication Settings.
5. Select Primary from the drop-down list to assign the appliance role.
6. Copy the Megastore ID. The appliances use this ID to confirm each other.

Configure Secondary Appliance Peer Replication

Configure the appliance settings to identify the appliance as the secondary and to define the communication with the primary appliance.

Follow these steps:

1. Use an internet browser to log in to the secondary appliance.
2. Select Configure, Maintenance, Storage Optimization Service.
3. Stop the optimization service.
4. Select Configure, Storage, Peer Replication Settings.
5. Select Secondary from the drop-down list to assign the appliance role.

6. Configure the secondary appliance settings:
   - Enter the Megastore ID of the primary.
   - Enter a shared secret key. You enter the same shared key when you complete the primary appliance configuration (see page 90).
   - Enter a port number on which the secondary appliance listens for connections from the primary appliance. The primary appliance replicates data to the secondary appliance through this port.
7. Click Apply to save your appliance changes.
8. Select Configure, Maintenance, Storage Optimization Service.
9. Start the optimization service.

Complete Primary Appliance Configuration

Complete the primary appliance configuration by configuring the shared key and secondary appliance URL, then restarting the optimization service.

Follow these steps:

1. Use an internet browser to log in to the primary appliance Riverbed SteelStore (Whitewater) UI.
2. Configure the remaining primary appliance settings:
   - Enter the same shared secret key that you entered on the primary appliance (see page 89).
   - Enter the secondary appliance URL. The URL is the IP address and the port number that the primary appliance uses to communicate with and replicate to the secondary appliance.
3. Click Apply to save your appliance changes.
4. Select Configure, Maintenance, Storage Optimization Service.
5. Start the optimization service.

   The optimization service should begin to show the progress of the primary replicating data to the secondary appliance. The replication can take some time. The time depends on the data amount that is stored on the primary appliance.

Peer Replication Disaster Recovery Scenarios

Your Disaster Recovery procedures should already include the following systems:

- Plans and processes to recover the MVS system
- CA Vtape subsystems, applications, parameter libraries, and the Global VCAT and BSDS control files
- The catalog and tape management systems

Recovering these systems can be critical for business resumption.

The following scenarios describe the basic recovery strategies when running SteelStore Peer Replication.

- If service to the secondary appliance is temporarily interrupted and the primary is unable to replicate to the secondary, then the primary appliance pauses replication to the cloud. The data is stored only locally until service to the secondary appliance is restored.

  If the interruption is only a temporary condition, the primary appliance resumes replication as soon as the secondary appliance becomes available. No further action is required.

- If service to the primary appliance is temporarily interrupted, the secondary appliance state remains unchanged. You can continue to use the secondary appliance to provide read access to stored data.

  If the interruption is only a temporary condition, the primary appliance resumes replication as soon as the primary appliance becomes available. No further action is required.

- If the secondary appliance goes down permanently, then you can disable Peer Replication on the primary appliance to return it to stand-alone mode. The primary appliance retains ownership for management and can then resume replication of data that is stored at the cloud provider.

  To disable Peer Replication on the primary appliance, connect through the command-line interface and enter the following commands:

  ```
  whitewater > enable
  whitewater # configure terminal
  whitewater # no service enable
  whitewater # no replication peer enable
  whitewater # service enable
  whitewater # show service
  ```

  Note: Riverbed changed the product name from *Whitewater* to *SteelStore*. The commands are customizable and may not reflect this change. For more information about these commands, see the *Whitewater Cloud Storage Appliance Command-Line Interface Reference Manual.*

- If the primary appliance goes down permanently, switch the secondary appliance to replace it.

  To return the secondary appliance to stand-alone mode, disable Peer Replication. In this mode, you can switch ownership for management and replication of the data that is stored at the cloud provider to the secondary appliance. Starting the optimization service after disabling Peer Replication causes the appliance to request a switch in ownership of the cloud data. The process can take some time, depending on the amount of data stored.

- After the switch of ownership completes, you can mount the NFS exports of the appliance for read/write access.

- To disable Peer Replication on the secondary appliance, connect through the command-line interface and enter the following commands:

  ```
  whitewater > enable
  whitewater # configure terminal
  whitewater # no service enable
  whitewater # no replication peer enable
  whitewater # service enable
  whitewater # show service
  ```

Appendix D: Configure Disk Appliances for Replication

This section contains the following topics:

Configure EMC Data Domain for Replication (see page 93)
Configure NetApp for Replication (see page 93)

Configure EMC Data Domain for Replication

For replication methods and recommendations, see the *Best Practices Guide*.

To ensure that you can access your data if you lose access to the primary appliance, use data replication from the primary appliance to a secondary appliance. This is especially true for private cloud implementations.

For Data Domain appliances, use the EMC Data Domain Enterprise Manager to configure Configure NFS Exports (see page 55) or Create Subdirectories for NFS Export Mounting (see page 56) with replication.

Follow these steps:

1. Use a web browser to log in to the EMC Data Domain Enterprise Manager.
2. Select the Replication tab to create, edit, or delete replication pairs.

For more information about configuring and setting up replication pairs, see the *EMC Data Domain Administration Guide* on the EMC website.

Configure NetApp for Replication

For replication methods and recommendations, see the *Best Practices Guide*.

To ensure that you can access your data if you lose access to the primary appliance, use data replication from the primary appliance to a secondary appliance. This is especially true for private cloud implementations.

For the NetApp appliances, use NetApp OnCommand System Management software to create and configure snapshot mirroring of volumes between appliances. For more information about the software, see the NetApp website http://www.netapp.com/.

Follow these steps:

1. Select the NetApp appliance to begin configuration of the necessary components.
2. Select Data Protection, SnapMirror.

   Use the SnapMirror Relationship Wizard to create a mirror. When creating a SnapMirror, define the source and target appliances, the synchronization schedule, and network bandwidth to mirror volumes to the remote NetApp appliance.

After you become familiar with setting up snapshot mirrors, see Using Bidirectional Replication for Disaster Recovery (see page 51).

Glossary

A/N

A/N is the abbreviation for alphanumeric.

Automated Class Selection (ACS)

Automated Class Selection (ACS) routine is a procedural set of ACS language statements. Based on a set of input variables, the ACS language statements generate the predefined SMS class name or a list of predefined storage groups names for a data set.

Bootstrap Data Set (BSDS)

Bootstrap Data Set (BSDS) is the backup or mirror data set for the Global VCAT, but does not contain DASD buffer information. You can recover the Global VCAT from the BSDS and also conversely.

CA Graphical Management Interface (CA GMI)

CA Graphical Management Interface (CA GMI) is the graphical management interface product that allows you to view and manage mainframe activity from a Windows PC. It consists of the CA Vantage Windows Client user-interface (referred to as the Windows Client), and the CA Vantage Web Client user-interface (referred to as the Web Client). Both of which interface with a z/OS server component to allow access to basic z/OS server functions. *CA GMI* is included free of charge with many CA products, including CA Vtape.

CA License Management Program (LMP)

The *CA License Management Program (CA LMP)* provides a standardized and automated approach to the tracking of licensed software. CA LMP uses common real-time enforcement software to validate the user configuration.

cell pool

Cell pool is an area of virtual storage that is subdivided into fixed-size storage areas named cells. CA Vtape uses cell pools to document lists of available resources (such as free LDSs and scratch Virtual Volumes), commands for execution, and Recalls in progress.

Channel Command Word

Channel Command Word (CCW) is the detailed instructions that are used to read and write data on a device that is attached to the mainframe.

Channel Path ID (CHPID)

A valid online *Channel Path ID (CHPID)* is required for the Virtual Devices. At startup, CA Vtape automatically selects the CHPIDs for the volume on which the data set referred to with the BSDS1 DD card is allocated. You can override by coding //CHPID DD in the SVTS PROC to point to a different device or by utilizing the ChpidDeviceList attribute of parmlib.

Complex, CA Vtape

A CA Vtape *Complex* is one or more CA Vtape subsystems running on one or more LPARs that share the CA Vtape Global VCAT and BSDS.

Control Data Sets (CDS)

CA Vtape uses a set of files or *control data sets (CDS)* to record system and configuration information. The CA Vtape control data sets consist of the Local VCAT, Global VCAT, and BSDS. CA Vtape control data sets should be considered critical system files. For more information, see the chapter "Control Data Sets" in the *Configuration Guide*.

Data Set Name (DSN)

Data Set Name (DSN) is the name of a data set.

Data-In-Virtual (DIV)

Data-In-Virtual (DIV) is a technique for holding data from linear data sets in memory to improve access performance.

Data-In-Virtual Entry (DVE)

Data-In-Virtual Entry (DVE) is a 4 MB virtual storage segment that CA Vtape uses when partitioning an LDS. This is the standard unit of data transfer between the Virtual Device dataspace and an LDS. Multiple DVEs are chained when writing to physical tape. The number of DVEs used determines the size of the Virtual Volume reported in the ISPF Interface and for reports.

dataspace

Dataspace is the area of storage that is defined for high-speed data storage and retrieval. The use of dataspaces allows the application to process as though it has much more central storage than it actually does. Data is considered mapped into a data space. Data-In-Virtual (DIV) is used to store and retrieve data. Dataspaces are used for each virtual drive and for the Global and Local VCAT data.

Deadly Embrace

Deadly Embrace is:

- A condition where a transaction cannot proceed. The transaction depends on exclusive resources that another transactions locked, which also depend on exclusive resources locked by the original transaction.
- Unresolved contention for resource use.

Eligible Device List (EDL)

Eligible Device List (EDL) is the list of eligible devices.

exclude filter

Exclude filter refers to a Data Set Filter List entry that indicates CA Vtape should not intercept a tape mount done for a Data Set Name matching this entry. An exclude works only if the DSN matched also matches an include entry for the same group. Data classes do not have exclude entries because patterning is not allowed for Data Class includes.

filter list

A *filter list* is a list of Data Set Names or patterns or Data Classes for which CA Vtape should or should not intercept a tape mount. Each filter list entry is assigned to a group number.

Global VCAT

Global VCAT refers to the Global Volume Catalog. This is a VSAM Linear Data Set containing information about resources shared by multiple CA Vtape Subsystems active on one or more LPARs. The shared resources are the Virtual Volumes and the NFS storage containing those volumes.

group

A *group* is a set of attributes or policies that tell CA Vtape how to process Virtual Volumes. *Group* attributes can be used to control things like the VOLSER range used to satisfy a mount request or If compression should be activated. Data class and or data set name filters are used to assign a *group* number to Virtual Volumes during scratch mount processing.

Hardware Configuration and Definition (HCD)

The Virtual Devices are defined to the operating system as a part of the CA Vtape installation. IBM's *Hardware Configuration and Definition (HCD)* software is used to create the definitions. These are software, not hardware, definitions that require only an HCD activation to use. An IPL is not necessary.
For more information, see the section "Define Virtual Devices Using IBM's Hardware Configuration and Definition (HCD) Dialogs" in the *Configuration Guide*.

High-Level Qualifier (HLQ)

High-Level Qualifier (HLQ) refers to the SMP/E installation data sets.

HIPER

HIPER is a SMP/E term that stands for Highly Pervasive problem.

Improved Data Recording Capability (IDRC)

Improved Data Recording Capability (IDRC) refers to the IBM hardware-based data compression option.
Virtual Volume compression lets you simulate *Improved Data Recording Capability (IDRC)* in CA Vtape Virtual Volumes by compressing the data as soon as the Virtual Device Engine receives it from the application.

include filter

Include filter refers to a Data Set Name or pattern, or Data Class Filter List entry that indicates CA Vtape should intercept a tape mount done with a matching Data Set Name or Data Class.

ISPF Customization

After SMP/E installation, *ISPF customization* refers to the panel-driven process of creating the SUTPARMS member used to initialize the Global VCAT, the BSDS, and the Local VCAT.

Local VCAT

Local VCAT refers to the Local Volume Catalog. VSAM Linear Data Set initialized with the SUTPARMS member. Each CA Vtape Subsystem has its own, unique Local VCAT into which parmlib attributes are loaded at startup and that is used as a work file by the running subsystem.

PREFIX

PREFIX refers to the CA Vtape DSN Prefix. This is the data set prefix used for generating the data set names for various CA Vtape data sets such as the DASD Buffer LDSs.

Program Temporary Fix (PTF)

Program Temporary Fix (PTF) is an IBM-sanctioned patch, often implemented using ZAP or SUPERZAP.

Program Temporary Fix in Error (PE)

Program Temporary Fix in Error (PE) is a SMP/E term.

PRP

PRP is a SMP/E term that stands for PTF Resolving PTF in error.

Scratch synchronization

Scratch synchronization is the process by which CA Vtape is informed which Virtual Volumes that the tape management system scratched.

Service Request Block (SRB)

The Virtualization Engine for CA Vtape is designed to run in *Service Request Block (SRB)* mode, a prerequisite of zIIP processing.
For more information, see the section "Exploitation of the zIIP Specialty Processor" in the *Configuration Guide*.

Small Product Enhancement (SPE)

Small Product Enhancement (SPE) is a SMP/E term.

Subsystem, CA Vtape

A CA Vtape *subsystem* is a single set of CA Vtape started tasks running on an LPAR. A CA Vtape *subsystem* can be part of a CA Vtape complex that shares a Global VCAT or BSDS with other CA Vtape *subsystems.* The subsystem can also make up its own, stand-alone complex. The started tasks that make up a CA Vtape *subsystem* are, by default, named SVTS, SVTSAS.SVT*n*PT, SVTSAS.SVT*n*UT, and *SVTSAS.SVTn*V*m* where *n* is the subsystem number and *m* is a number from 1 to 8.

SVTJCL library

SVTJCL library is a work data set into which the customization panels generate the necessary JCL members to define the global or shared resources. Resources include the Global VCAT, BSDS, and DASD Buffer LDSs.

System Management Facility (SMF)

System Management Facility (SMF) is a component of IBM z/OS. SMF provides a standardized method for writing out records of activity to a file (or data set). SMF provides full instrumentation of baseline activities that are running on z/OS, including: I/O, network activity, software usage, error conditions, processor utilization, and so on.

Task Control Block (TCB)

A Task Control Block (TCB) is a data structure in the operating system kernel containing the information to manage a particular process.

Unit Control Block (UCB)

Unit Control Block (UCB) is a control block in storage that describes the characteristics of a particular I/O device on the operating system.

Version Release and Modification (VRM)

CA Vtape implemented the *Version Release and Modification (VRM)* level concept before Release 11.5. The VRM is used to prevent incompatible software levels between subsystems sharing the control data sets from running simultaneously. As each CA Vtape subsystem is started, it compares its VRM to that of the Global VCAT to determine if it is safe to continue the startup. If the VRM levels are incompatible, the subsystem shuts down.

Virtual Device Engine

Virtual Device Engine is another name for the SVTSAS.SVT*n*V*n* subaddress spaces that control the Virtual Devices.

Virtual Devices

Virtual Devices are the virtual tape drives.

Virtual Storage Access Method (VSAM)

Virtual Storage Access Method (VSAM) is an access method for disk files that offers various techniques to access data. Data access includes: sequential, keyed, indexed, and relative record. Some of these techniques support special media access including data compression, caching, and data striping.

Virtual Tape System (VTS)

CA Vtape is installed into an address space and is often known as a *Virtual Tape System (VTS)*.

Virtual Tape Unit (VTU)

A *Virtual Tape Unit (VTU)* is a control block. VTUs are attached when a z/OS command varies Virtual Devices online. They are detached when varied offline. They are attached to the SVTSAS.SVT*n*V*n* subaddress spaces.

Virtual Volume Entry (VVE)

A *Virtual Volume Entry (VVE)* is a control block. Documents a single Virtual Volume. Each VVE uses 4 KBs of space in the Global VCAT.

Virtual Volume Pool (VVP)

A *Virtual Volume Pool (VVP)* is a control block that documents a consecutive series of 100 Virtual Volumes. Example volumes include: 000000-000099, 000100-000199, and so on.

Volume Catalog (VCAT)

Volume Catalog (VCAT) is:

- An LDS used by one or more CA Vtape subsystems to contain control information or temporary data. See glossary items Global VCAT and Local VCAT.
- A DD in the SVTS PROC or utility JCL for the Local VCAT.

Volume Mount Analyzer (VMA)

Refers to the IBM *Volume Mount Analyzer (VMA)*.

write-protected

Write-protected refers to a Virtual Volume or Virtual Volume pool that has been placed in read-only status in CA Vtape.