StorageTek Virtual Library Extension

Configuring Host Software Guide

Release 1.5

E57794-06

January 2018

ORACLE®

StorageTek Virtual Library Extension Configuring Host Software Guide, Release 1.5

E57794-06

Contents

List of Examples

List of Figures

Preface

This guide provides information for configuring host software for Virtual Library Extension (VLE).

Audience

This guide is for Oracle or customer personnel responsible for configuring the MVS software host software for Oracle's StorageTek VLE.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc`.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info` or visit `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs` if you are hearing impaired.

1 Introduction

Oracle's StorageTek Virtual Library Extension (VLE) is back-end disk storage for VTSS. VLE provides:

- Support for migrating and recalling VTVs to, and from Oracle Cloud Storage

  For more detailed information, visit:

  – "Network Setup Requirements"

  – "VLE Oracle Cloud Storage"

  Note: Refer to `http://docs.oracle.com/cloud/latest/`for further information on setting up a Cloud account.

- Support for 400 MB, 800 MB, 2 GB, 4 GB, and 32 GB VTV

  Note: For setting up and using 32 GB VTV, refer to ELS 7.3 documentation.

- An additional storage tier in the VSM solution. VTVs can now migrate from VTSS-to-VLE to provide fast access to recent data. Additionally, VTVs can transition from VLE storage to tape media (MVCs) for long term archive. You can control how VTVs are migrated and archived through the existing HSC Management and Storage Classes, providing full backward compatibility with previous configurations.
- Back-end disk storage shared between multiple VTSS systems ensuring high-availability access to data.

  Note: For VLE 1.1 and higher, a VLE is a collection of nodes interconnected with a private network.

To VTCS, a VLE looks like a tape library except that the VTVs are stored in Virtual Multi-Volume Cartridges (VMVCs) on disk. With VLE, you can configure either a VLE and tape, or a VLE only (for example, with Tapeless VSM configurations) back-end VTV storage solution. A VTSS can migrate VTVs to, and recall them from a VLE, just as is done with a real tape library.

Caution:

- If you have a VLE system, HSC/VTCS uses SMC communication services to communicate with the VLE. To ensure that these services are available during VTCS startup, Oracle recommends that you first issue the start command for HSC, then immediately issue the start command for SMC, while HSC is initializing.
- Stopping SMC stops VTCS from sending messages to the VLE, which effectively stops data transfer. Therefore, you should ensure that VTCS activity is quiesced, or VTCS is terminated before stopping SMC.
- You cannot use AT-TLS with the SMC HTTP server if you are using VLE.
- Tapeless VSM configurations, provides only a single-node VLE attached to a specific VTSS. Should the VLE go offline, you will lose access to any VTVs migrated to the VLE that are not resident in the VTSS, until the VLE comes back online.

The VLE solution consists of:

- Virtual Tape Storage Subsystem (VTSS) hardware and microcode
- Virtual Tape Control Subsystem (VTCS) software and Storage Management Component (SMC)
- VLE hardware and software

Network Setup Requirements

If network redundancy is required, each IP connection between VSM 5/6 and VLE, VLE-to-VLE and VLE-to-SMC must be configured on separate subnets.

VLE Hardware and Software

The VLE, which is a factory-assembled unit in a Sun Rack II Model 1242, includes the following hardware:

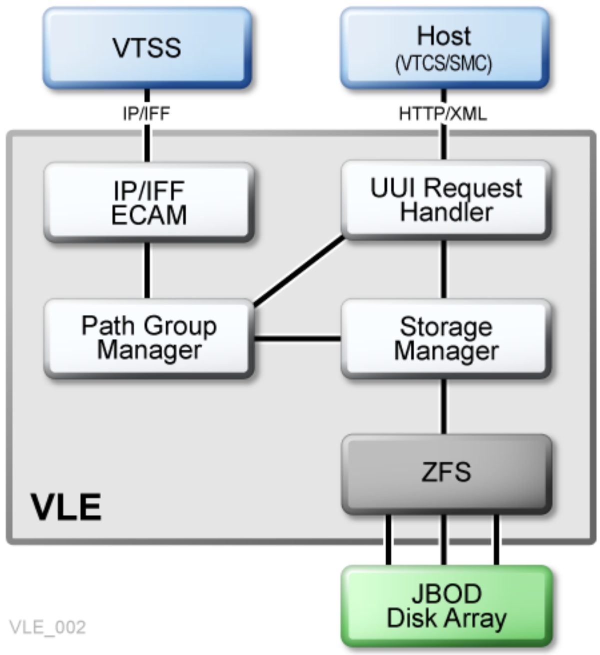
- A server built on a Sun Server X4-4 platform.
- Four motherboard 10 Gb ports, two of which can be used for data transfer and other purposes. Two are dedicated for management, service, and support.
- A service (ILOM) port.
- Four dual port 10 Gb Fibre Optic (six ports available) plus two 10 Gb Copper ports.
- One or more Oracle Storage Drive Enclosure DE2-24Cs (DE2-24C) that contain disk (HDDs) in a ZFS RAID array, scalable in effective capacities starting at 200 TB for a single JBOD VLE (assuming a 4 to 1 compression ratio when the data is migrated to the VLE).
- A DVD drive.

The VLE software consists of:

- Oracle Solaris 11 Operating System.

- ZFS file system and MySQL database.
- The VLE application software.

Figure 1–1 VLE Subsystem Architecture



As Figure 1–1 shows, the VLE application software is comprised of:

- HTTP/XML is the data protocol for host to VLE communications.
- The Universal User Interface (UUI) Request Handler, which processes UUI requests from and produces responses to Storage Management Component (SMC) and Virtual Tape Control Software (VTCS). The UUI Request Handler determines which VLE components are used to service a request.

  UUI Request Handler calls:

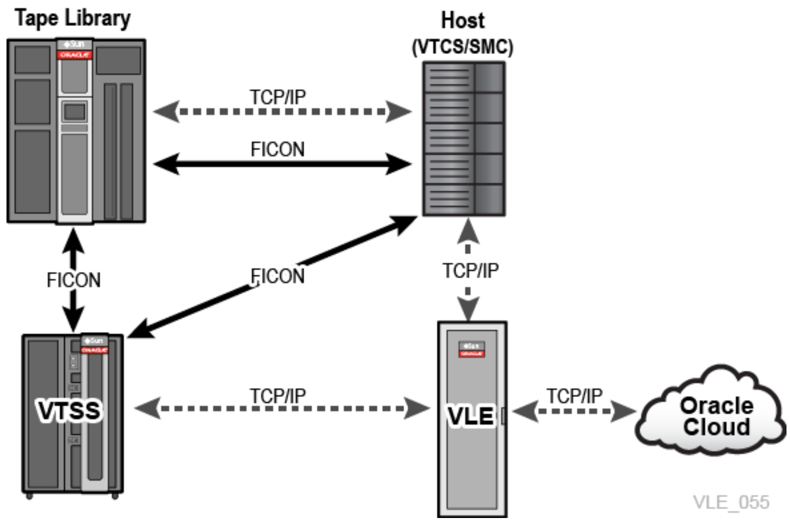
  – The PathGroup Manager to schedule VTV migrates and recalls. The PathGroup Manager manages all Path Groups, where each Path Group manages a single VTV data transfer between the VTSS and the VLE.

  – The Storage Manager to schedule all report generation.
- The VLE Storage Manager component manages the VMVC/VTV data and meta data on the VLE. The VLE Storage Manager stores VTV data on and retrieves it from the ZFS on the JBOD array.
- TCP/IP/IFF is the data protocol for host to VLE communications, where the IP/IFF/ECAM component handles communications between the VTSS and the VLE.

Single Node VLE Configuration

Figure 1–2 shows a single node VLE configuration.

Figure 1–2 Single Node VLE in a VSM System



As Figure 1–2 shows (where 1 is the MVS host and 2 is the library):

- Multiple TCP/IP connections (between the VTSSs' IP ports and the VLEs' IP ports) are supported, as follows:
  - A single VLE can connect up to 8 VTSSs, so VTSSs can share VLEs.
  - A single VTSS can connect to up to 4 VLEs to increase buffer space for heavy workloads.
- A single VTSS can be attached to:
  - Only RTDs
  - Only other VTSSs (clustered)
  - Only VLEs
  - Any combination of the above.
- TCP/IP is the only supported protocol for connections between the VLE and the VTSS and for connections between the VLE and hosts running SMC and VTCS.

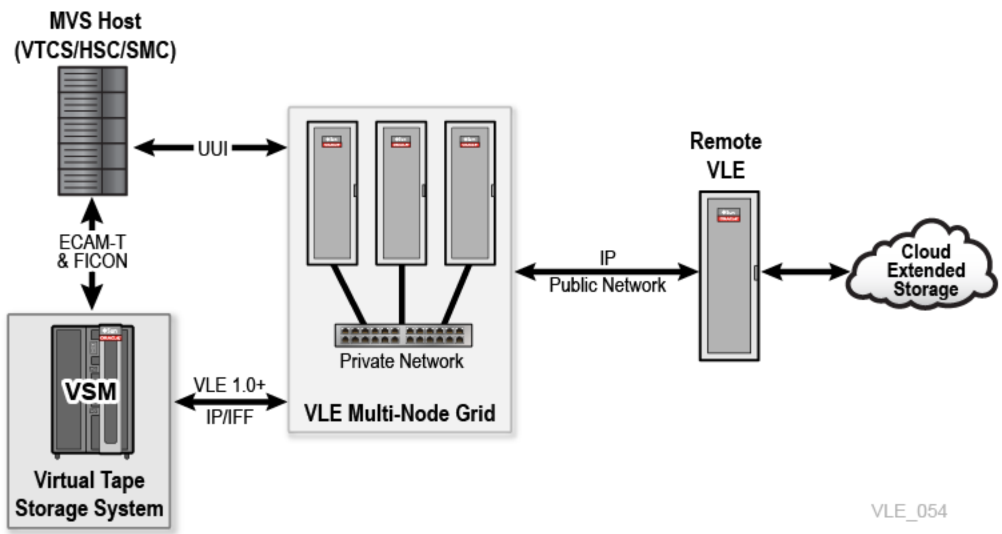
Multi-Node VLE Systems

Multi-node VLE systems enable massive scaling of the VLE storage system. You can construct multi-node systems that can consist of one to 64 nodes, with multiple nodes interconnected by a private network. A multi-node VLE appears to SMC/VTCS as a single VLE. The VLE ships with 4 TB JBODs, so a single VLE can scale between 200 TB (for a one JBOD system) and 100 PB (for a fully populated 64-node VLE).

> **Note:** These are effective capacities, assuming 4:1 compression. VLE was **architected** for up to 64 nodes, but has only been *validated* for up to seven nodes.

Figure 1–3 shows a VLE multi-node complex, where, the nodes are cross-connected into a dedicated 10 GE switch so that each node can access any other node in the complex:

Figure 1–3 VLE Multi-Node Complex



VLE-to-VLE Data Transfer

The VLE storage system can manage data transfers independently of the VTSS, which frees VTSS resources for front-end (host) workload, improving the overall VTSS through-put. For example:

- If your migration policies specify that there should be two VLE copies of a VTV (either in the same or separate VLEs), then the first migrate to a VLE will cause data to be transferred from the VTSS and all subsequent VLE migrates for the VTV may be achieved through a VLE to a VLE copy. This reduces the VTSS cycle times required to migrate all copies of a VTV.
- If your environment runs:
  - VLE 1.2 or higher
  - VTCS 7.1 (with the supporting PTFs) or VTCS 7.2 and higher

    Then you can use VTCS to define more VLE devices than there are VTSS to VLE paths through the `CONFIG STORMNGR VLEDEV` parameter. If you use this addressing scheme, then the VTSS resources used to migrate all of the VTV copies to VLE are reduced even further because the path from the VTSS to the target VLE is only reserved when the data transfer is direct from the VTSS to the VLE. For all VLE VRTD actions, a path from the VTSS is only reserved when VTSS data transfer is required.

VTV Encryption

The encryption feature enables encryption of VMVCs written to the VLE system. Encryption is enabled on a *per node basis*, through an *encryption key* stored on the node, backed up on a USB device. Encryption is entirely managed through the VLE GUI; the host software has no knowledge of encryption, as the VLE de-encrypts VTVs that are recalled to the VTSS.

VTV Deduplication

Deduplication eliminates redundant data in a VLE complex. Deduplication, which is controlled by the `STORCLAS` statement `DEDUP` parameter, increases the effective VLE capacity and is performed by the VLE before the VTV is written to a VMVC.

To assess deduplication results, enable deduplication, monitor the results with the `SCRPT` report, and fine tune deduplication, as necessary. The `SCRPT` report provides the approximate "reduction ratio" for the deduplicated data, which is uncompressed GB divided by used GB. The Reduction Ratio, therefore, includes *both* VTSS compression *and* VLE deduplication. A larger reduction ratio indicates more effective compression and deduplication.

For example, the VTSS receives 16 MB of data, compresses it to 4MB, and writes the compressed data to a VTV. VLE subsequently deduplicates the VTV to 2 MB and writes it to a VMVC. Thus, the reduction ratio is 16 MB divided by 2 MB, or 8.0:1.

Early Time To First Byte (ETTFB)

Early Time To First Byte (ETTFB), also known as the concurrent tape recall/mount feature, allows the VTSS to use a VTD to read data as it being recalled from VLE:

- ETTFB is set globally through `CONFIG GLOBAL FASTRECL`.
- If `CONFIG GLOBAL FASTRECL=YES`, you can disable ETTFB on per VTSS basis through `CONFIG VTSS NOERLYMNT`.

`CONFIG GLOBAL` **and** `CONFIG VTSS` apply to *both* ETTFB for RTDs and ETTFB for VLE.

ETTFB is applicable only to VSM5 systems.

Frame Size Control

Frame Size Control specifies the use of Jumbo Frames on each copy link:

> **Note:** The entire infrastructure between the VSM and VLE, or between VLEs must support Jumbo Frames to work. If any portion of the infrastructure between these connections does not support Jumbo Frames, it will not work.

- *If* your TCP/IP network supports Jumbo Frames, enabling this option can improve network performance.
- You enable Jumbo Frames by selecting the `Jumbo Frames` check box on the `Port Card Configuration Tab`. Selecting this box sets the MTU (Maximum Transmission Unit) value to 9000 for the port.
- It is recommended that Jumbo Frames be enabled on links that are set for VLE-to-VLE transfer.

Oracle Cloud Extended Storage

VLE 1.5.2 and higher offers a connection from the VLE to the Oracle Cloud. VLE can be configured to optionally migrate and recall customer data directly to and from the Oracle Cloud. VLE configuration options support any combination of data storage in the local VLE disk pool and/or the Oracle Cloud.

VLE supports three Oracle Cloud options: Oracle Cloud Object Storage, Oracle Cloud Archive Storage, and Encryption within Oracle Cloud. See "VLE Oracle Cloud Storage" for further explanation of the supported Oracle Cloud options.

2

Configuring the MVS Host Software

This chapter provides the MVS host software configuration for VLE, as described in the following sections:

- "Key Configuration Values"
- "MVS Host Software Configuration Tasks"

Key Configuration Values

The following sections describe values required for software configuration that must match values that are typically already set in the hardware configuration, and recorded in the `IP_and VMVC_Configuration.xls` worksheet.

Subsystem Name

The subsystem name of the VLE, which is set by the VLE installation scripts, is specified, as follows:

- Either the VTCS `CONFIG TAPEPLEX STORMNGR` parameter or the `CONFIG STORMNGR NAME` parameter
- The VTCS `CONFIG RTD STORMNGR` parameter
- The SMC `STORMNGR NAME` parameter
- The SMC `SERVER STORMNGR` parameter
- The HSC `STORCLAS STORMNGR` parameter

VTSS Ethernet Port Addresses

The VTSS Ethernet port addresses are required to configure the VTSS-to-VLE IP connection through the `CONFIG RTD IPIF` parameter. For VSM 5s, this value must match the values specified on the VSM5 IFF Configuration Status screen. For VSM 6s, this must be unique for each VTSS, but does **not** correspond to an actual value on the VSM 6 TCP/IP ports.

IP Addresses of VLE Ports for Host (UUI) Communication

The IP addresses of VLE port for host (UUI) communication are required for the SMC `SERVER IP` parameter.

VMVC Volsers

Required to define VMVCs to SMC/VTCS, method of definition depends on the software version. See "Defining the VLE VMVCs to the MVS Host Software and Including VMVCs in an MVC Pool."

VMVC Reclamation Threshold

For more information, see "Specifying the Reclamation Policy for VMVCS."

VTV Deduplication

The `STORCLAS DEDUP` parameter specifies whether VTV data migrated to VMVCs in the specified `STORMNGR` is deduplicated. For example:

```
STORCLAS NAME(VLEDEDUP)STORMNGR(VLE1) DEDUP(YES)
```

This `STORCLAS` statement specifies to deduplicate data in Storage Class `VLEDEDUP` that is migrated to `VLE1`. For more information, see *ELS 7.3 Command, Control Statement, and Utility Reference.*

Deduplication increases effective VMVC capacity and is performed by the VLE before the VTV is written to a VMVC. Oracle recommends, therefore, that you initially enable deduplication, then monitor the results with the `SCRPT` report and fine tune deduplication, as necessary.

Early Time to First Byte (ETTFB)

ETTFB (also known as the concurrent tape recall/mount feature) allows host applications to read data while VTVs are being recalled from *either* VMVCs or RTDs. ETTFB is done by overlapping the VTV recall and mount phases, which allows the application to read the VTV data sooner. If the application attempts to read a part of the VTV that has not been recalled, the applications' I/O request is blocked until the required VTV data has been recalled. With ETTFB for VLE, application access to the first byte occurs in under a second, making the VLE a true extension of the VTSS. Therefore, VLE ETTFB is a good choice for applications that serially access the VTV data. VLE ETTFB is generally *not* a benefit for those applications that stack multiple files on a single VTV, including HSM and image management applications. In these types of applications, the desired data is usually not at the beginning of the VTV, but rather, at some random location in the VTV.

ETTFB is disabled by default. You can globally enable ETTFB through the `CONFIG GLOBAL FASTRECL` parameter. If you globally enable ETTFB, you can disable it for individual VTSSs through the `CONFIG VTSS NOERLYMNT` parameter.

VTVs that have incurred an ETTFB recall error have an error flag set in their VTV record in the CDS. These VTVs are subsequently not selected for ETTFB. If you want to reset the error flag, do any of the following:

- Enter a `VTVMAINT SCRATCH(ON)` command for the VTV.
- Migrate the VTV to a new MVC copy.
- Import the VTV.
- Create a new version of the VTV.
- Scratch the VTV.

MVS Host Software Configuration Tasks

Adding VLE to a VSM system requires the tasks described in the following sections:

- "Acquire the ELS supporting PTFs for VLE"
- "Updating the SMC OMVS RACF Security Entry"
- "Modifying the SMC SCMDS file"
- "Updating the VTCS CONFIG Deck to Define VLE"
- "Defining the VLE VMVCs to the MVS Host Software and Including VMVCs in an MVC Pool"
- "Updating the MVS Host Software Policies"

For more information on the commands and control statements referenced in this chapter, see *ELS 7.x Command, Control Statement, and Utility Reference*.

Acquire the ELS supporting PTFs for VLE

For ELS 7.2 and later, support is included in the base level. For ELS 7.1, get the latest SMP/E receive HOLDDATA and PTFs (L1H16J6, L1H1674) and SMP/E APPLY with GROUPEXTEND.

Updating the SMC OMVS RACF Security Entry

The VLE requires SMC to have an OMVS RACF security entry to have a TCP/IP connection to the host.

OMVS is a segment associated with the RACF userid. The SMC started task must have a userid associated with OMVS, either in the RACF STARTED class definition or the ICHRIN03 LNKLST module. The userid associated with the SMC task needs to have an OMVS segment defined to it within RACF, as follows:

```
ADDUSER userid
DFLTGRP(groupname)OWNER(owner)OMVS(UID(uidnumber))
```

Or, if the userid already exists but does not have an OMVS segment:

```
ALTUSER userid OMVS(UID(uidnumber))
```

Modifying the SMC SCMDS file

SMC manages all communication between VTCS and VLE, so SMC must know how to connect to the VLE server. To do so, add an SMC STORMNGR statement for each VLE system plus one or more SMC SERVER statements that define the TCP/IP control paths for the VLE. For 7.0 and later, you may want to do this in your SMC CMDS file, as shown in Example 2–1.

Example 2–1 SMC Commands for VLE

```
TAPEPLEX NAME(TMVSA)LOCSUB(SLS0)
SERVER NAME(ALTSERV) TAPEPLEX(TMVSA) +
HOSTNAME(MVSX) PORT(8888)
STORMNGR NAME(VLE1)
SERVER NAME(VLE1)+ STORMNGR(VLE1)IP(192.168.1.10)PORT(60000)
```

Example 2–1 contains:

- A TAPEPLEX statement, which defines a single TapePlex, TMVSA, with an HSC/VTCS running on the same MVS host (SLS0).
- A SERVER statement, which defines a backup HSC/VTCS subsystem (ALTSERV) running on another host.
- A STORMNGR command that defines a VLE (VLE1).
- A second SERVER command that defines a UUI communication path to the VLE, where:
  - The server name is VLE1.
  - The STORMNGR parameter value is VLE1.
  - The IP parameter value is the VLE port IP address of 192.168.1.10 for UUI communications.
  - The PORT parameter value is 60000; this value is always used for the SERVER PORT parameter for SMC communication with a VLE.

Updating the VTCS CONFIG Deck to Define VLE

You must update VTCS CONFIG deck to define the VLE and the connectivity from the VTSS systems to the VLE. VTCS can drive VLE, as follows:

- *For VTCS 7.0 and later*, the CONFIG TAPEPLEX statement defines the TapePlex that VTCS is running under and provides the list of defined VLEs on the CONFIG TAPEPLEX STORMNGR parameter, as shown in Example 2–2.

Example 2–2 VTCS 7.0 CONFIG VLE

```
TAPEPLEX THISPLEX=TMVSA STORMNGR=VLE1
VTSS NAME=VTSS1 LOW=70 HIGH=80 MAXMIG=8 MINMIG=4 RETAIN=5
RTDPATH NAME=VL1RTD1 STORMNGR=VLE1 IPIF=0A:0
RTDPATH NAME=VL1RTD2 STORMNGR=VLE1 IPIF=0A:1
RTDPATH NAME=VL1RTD3 STORMNGR=VLE1 IPIF=0I:0
RTDPATH NAME=VL1RTD4 STORMNGR=VLE1 IPIF=0I:1
RTDPATH NAME=VL1RTD5 STORMNGR=VLE1 IPIF=1A:0
RTDPATH NAME=VL1RTD6 STORMNGR=VLE1 IPIF=1A:1
RTDPATH NAME=VL1RTD7 STORMNGR=VLE1 IPIF=1I:0
RTDPATH NAME=VL1RTD8 STORMNGR=VLE1 IPIF=1I:1
VTD LOW=6900 HIGH=69FF
```

In Example 2–2, note:

- The CONFIG TAPEPLEX statement, which defines TMVSA as the TapePlex that VTCS is running under, and the names of all connected VLEs, (which, in this example, is a single VLE called VLE1).
- The CONFIG RTDPATH statements, which define a single VLE RTD for each path from the VTSS to the VLE. In this example, the CONFIG RTDPATH statements for VTSS1 specify:
  - The name of the RTDPATH.
  - The connections to the defined VLEs (STORMNGR=VLE1).
  - The IPIF value for each VTSS-to-VLE port connection in *ci:p* format where:
    - * *c* is 0 or 1.
    - * *i* is A or I.
    - * *p* is 0 through 3.

> **Note:** For VSM 5s, this value must match the values specified on the VSM5 IFF Configuration Status Screen. For VSM 6s, this must be unique for each VTSS but does *not* correspond to an actual value on the VSM 6 TCP/IP ports.

- VTCS 7.1 and later, systems can, of course, drive VLE 1.5.1 as VTCS 7.0 does. In this mode, however, the number of VLE RTD targets is limited by the number of paths from a VTSS. Additionally, the VLE RTDs are assigned to fixed VTSS paths. The path from a VTSS to the VLE is always reserved by VTCS regardless if any VTSS-to-VLE data transfer is occurring.

  However, with VTCS 7.1 and later, you can define a VLE with more VLE RTD targets than there are paths from the VTSS to the VLE, which means:

  - The path from the VTSS to the VLE is *not* reserved unless a VTSS-to-VLE data transfer is required.
  - More VLE RTD operations can occur simultaneously. For example, an audit of a VMVC does *not* require a data transfer between the VTSS and the VLE.

  As shown in Example 2–3, the VLEs are defined through a `CONFIG STORMNGR` statement, *not* the `CONFIG TAPEPLEX STORMNGR` parameter. The `CONFIG STORMNGR` statement specifies the VLEs that VTCS connects to. Additionally, for each VLE, the `CONFIG STORMNGR VLEDEV` parameter defines the number and the names of the RTD devices that the VLE emulates. The more devices defined (up to the maximum of 96 devices per VLE), the greater the level of concurrent activities VTCS can schedule on the VLEs.

Example 2–3 VTCS 7.1 CONFIG VLE

```
TAPEPLEX THISPLEX=TMVSC
STORMNGR NAME=VLE1 VLEDEV(S000-S05F)
STORMNGR NAME=VLE2 VLEDEV(S000-S05F)
VTSS NAME=VTSS1 LOW=70 HIGH=80 MAXMIG=8 MINMIG=4 RETAIN=5
RTDPATH NAME=VL1RTD1 STORMNGR=VLE1 IPIF=0A:0
RTDPATH NAME=VL1RTD2 STORMNGR=VLE1 IPIF=0A:1
RTDPATH NAME=VL1RTD3 STORMNGR=VLE1 IPIF=0I:0
RTDPATH NAME=VL1RTD4 STORMNGR=VLE1 IPIF=0I:1
RTDPATH NAME=VL1RTD5 STORMNGR=VLE2 IPIF=1A:0
RTDPATH NAME=VL1RTD6 STORMNGR=VLE2 IPIF=1A:1
RTDPATH NAME=VL1RTD7 STORMNGR=VLE2 IPIF=1I:0
RTDPATH NAME=VL1RTD8 STORMNGR=VLE2 IPIF=1I:1
VTD LOW=6900 HIGH=69FF
```

In Example 2–3, note:

- The `CONFIG TAPEPLEX` statement now simply defines `TMVSC` as the TapePlex that VTCS is running under. It does *not* define the connected VLEs.
- The `CONFIG STORMNGR` statements, which define the VLEs configured in this system - `VLE1` and `VLE2`, specify the number of VLE devices through the `VLEDEV` parameter.

  In this example, each VLE has the maximum of 96 emulated devices, allowing VTCS to schedule up to 96 processes on each VLE. The VLE device addresses are in the form of `S`*xxx* (where *xxx* is a hexadecimal value).

  Example: S000-S05F represents 96 emulated devices.
- The `CONFIG RTDPATH` statements for VTSS1, which specify:

- The name of the RTDPATH
- The connections to the defined VLEs (STORMNGR=VLE1, STORMNGR=VLE2)
- The IPIF value for each VTSS to VLE port connection in *ci:p* format where:
  * *c* is 0 or 1
  * *i* is A or I
  * *p* is 0 through 3

> **Note:** For VSM5s, this value must match the values specified on the VSM5 IFF Configuration Status screen. For VSM 6s, this must be unique for each VTSS, but does **not** correspond to an actual value on the VSM 6 TCP/IP ports.

Specifying the Reclamation Policy for VMVCS

VLE MVC media (VMVCs) is subject to fragmentation and must be reclaimed just like real MVCs. The VMVC reclaim process, however, uses far fewer resources than a standard reclaim. The reclaim threshold for a VMVC is specified through the CONFIG RECLAIM VLTHRES parameter. The lower that you set VLTHRES, the more frequent VTCS will run reclaim on the VMVCs, leading to greater, effective capacity of the VMVS (less fragmentation).

Defining the VLE VMVCs to the MVS Host Software and Including VMVCs in an MVC Pool

VMVC volsers must be defined both to the MVS host software and to the VLE. The VMVCs are defined to the VLE as a part of the VLE configuration. The following sections describe how to define the VMVCs to the MVS host software.

Creating VMVC Volume Pools (7.0 and Later)

1. Code HSC POOLPARM or VOLPARM statements to define the VMVC pools.

   For example, to define two separate pools for VLE1 and VLE2:

   ```
   POOLPARM NAME(LEPOOL1)TYPE(MVC)
   VOLPARM VOLSER(VL0000-VL880)

   POOLPARM NAME(LEPOOL2)TYPE(MVC)
   VOLPARM VOLSER(VL2000-VL2880)
   ```

2. Run SET VOLPARM to validate the POOLPARM or VOLPARM statements.

   ```
   SET VOLPARM APPLY(NO)
   ```

   APPLY(NO) validates the statements without loading them. If you like the results, proceed to the next step. Otherwise, rework the volume definitions along with this step, and if the definitions are valid, proceed to the next step.

3. Run SET VOLPARM to load the POOLPARM or VOLPARM statements.

   ```
   SET VOLPARM APPLY(YES)
   ```

Updating the MVS Host Software Policies

The following sections tell how to update the MVS host software policies to direct data to the VLE system.

Creating Storage and Management Classes for VLE

Management Classes specify *how* VTCS manages VTVs. The HSC `MGMTclas` control statement defines a Management Class and its attributes. For example, the `DELSCR` parameter of the `MGMTclas` statement specifies whether VTCS deletes scratched VTVs from the VTSS. Management Classes can also point to *Storage Classes*, which specify *where* migrated VTVs reside. The HSC `STORclas` control statement defines a Storage Class and its attributes. You specify the VLE system as the destination for migrated VTVs through the `STORCLAS STORMNGR` keyword. **For example:**

```
STOR NAME(VLOCAL) STORMNGR(VLESERV1) DEDUP(YES)
STOR NAME(VREMOTE) STORMNGR(VLESERV2)DEDUP(YES)
```

The preceding statements define a "local" Storage Class (`VLOCAL`) on the `VLE1` and a "remote" Storage Class (`VREMOTE`) on the `VLE2`. As these `STORCLAS` statements specify, all migrations to storage class `VLOCLAL` or `VREMOTE` must go to the specified VLEs. Deduplication is specified for both Storage Classes.

You can be less restrictive than this, if desired. For example, if you define an MVCPOOL that contains both VMVCs and MVCs, you can setup the migration policies to migrate to a VLE. However, if the VLE becomes full or not available, continue to migrate to real tape media (MVCs). For example, the MVC pool `DR` is defined, as follows:

```
POOLPARM NAME(DR)TYPE(MVC)
VOLPARM VOLSER(VL0000-VL0100)
VOLPARM VOLSER(ACS000-ACS099)
```

Pool `DR`, therefore, contains both MVCs and VMVCs. A Storage Class that specifies pool DR will migrate first to VMVCs and only use MVCs if VMVCs are not available.

Example:

```
STOR NAME(DRCLASS) MVCPOOL(DR)DEDUP(YES)
```

This method is valuable if you have a configuration where both an ACS and a VLE are connected to the VTSS systems.

Next, to specify migration to VLE, you specify the VLE Storage Classes you defined through the `MGMTCLAS MIGPOL` parameter. For example:

```
MGMT NAME(M1) MIGPOL(VLOCAL,VREMOTE)
MGMT NAME(M2) MIGPOL(DRCLASS)
```

Management Class `M1` migrates one VTV copy to the "remote" VLE, one copy to the "local" VLE. Management Class `M2` migrates a single VTV copy to the Storage Class that points to the "mixed" MVC pool that contains both MVCs and VMVCs.

Note: In addition to directing migration to a VLE also consider:

1. If you are running at ELS 7.0 or higher, you can use HSC `MIGRSEL` and `MIGRVTV` to fine tune migration to VLE. Using these statements, you can cause migration of data in a Management Class to start to one Storage Class before another. This method is typically used to ensure that a critical DR copy is made as soon as possible. For more information, see *Configuring HSC and VTCS.*
2. On a VLE 1.1 and later system, if multiple VLEs are connected to each other and to the VTSS, by default, VTCS preferences VLE-to-VLE connections to make multiple VTV copies. You can control this behavior, as described in Controlling VLE-to-VLE Copy.

Controlling VLE-to-VLE Copy

For VLE-to-VLE connections, if a VTV copy resides on both a VTSS and one VLE, and you want to migrate it to a connected VLE, the default is to use the VLE-to-VLE connection. For example, a DR scenario with a local VLE (LOCVLE) and remote VLE (REMVLE) connected to VTSSA. You want to migrate two VTV copies:

- First, a local copy from VTSSA to LOCVLE.
- Second, a copy using VLE-to-VLE copy from LOCVLE to REMVLE using VLE-to-VLE Replication (Versus VTSS-to-VLE migration).

To make the VTV copies as desired, do the following:

1. Create a STORCLAS statement that sends a VTV copy to LOCVLE.

   ```
   STORCLAS NAME(FORLOCAL) STORMNGR(LOCVLE)
   ```

2. Create a STORCLAS statement that sends a VTV copy to REMVLE.

   ```
   STORCLAS NAME(FORREMOT) STORMNGR(REMVLE)
   ```

3. Create MGRVTV statements that specify that migrates to Storage Class FORLOCAL occur before migrates to Storage Class FORREMOT.

   ```
   MIGRVTV STOR(FORLOCAL) INITIAL
   MIGRVTV STOR(FORLOCAL) SUBSEQNT(360)
   ```

   Finally, create a MGMTCLAS statement that specifies two VTV copies, one to the local site and the other to the remote site:

   ```
   MGMTCLAS NAME(DRVLE) MIGPOL(FORLOCAL,FORREMOT)
   ```

Routing Data to VLE

To route data to VLE, first create an SMC POLICY command that specifies a VLE Management Class. Next, create SMC TAPEREQ statements that route the desired workload to the SMC VLE policy. For example:

```
POLICY NAME(VLEDR) MEDIA(VIRTUAL) MGMT(DRVLE)
```

```
TAPEREQ DSN(HR.**) POLICY(VLEDR)
```

The preceding example assigns the VLEDR policy to all tape datasets with an HLQ of HR.

3 VLE Oracle Cloud Storage

Oracle's Cloud Extended Storage is an option that allows the customer additional storage capacity. Refer to `http://docs.oracle.com/en/cloud/iaas/storage-cloud/index.html` for further information on setting up a Cloud account, and "Network Setup Requirements".

- `http://docs.oracle.com/cloud/latest/trial_paid_subscriptions/CSGSG/toc.htm`
- For up-to-date Cloud information, see:

  `http://docs.oracle.com/cloud/latest/`
- For further assistance, see:

  `http://docs.oracle.com/en/cloud/iaas/storage-cloud/index.html`

> **Note:** The recommendations for the for VLE 1.5.3 are the following.

- Limit to a maximum of 16 connections for mapping to Cloud VMVCs
- No deduplication on the Cloud VMVCs (deduplication on the disk based VMVCs is supported)
- A vlolser name should be comprised of upper-case letters (A to Z) and numeric (0-9) values, and must have a length of six characters. A volser name cannot have lower-case letters, or less than 6 characters.

VLE supports three Oracle Cloud options: Oracle Cloud Object Storage, Oracle Cloud Archive Storage, and Encryption within Oracle Cloud.

The main difference between Oracle Cloud and Oracle Cloud Archive Storage is cost. Oracle Cloud Archive Storage is significantly less expensive than Oracle Cloud. See an Oracle sales representative or available online documentation for current cost options.

Encryption, if VTV data is stored in the Oracle Cloud, is offered for both Archive and non-Archive Cloud offerings. There is no cost delta for encryption. Customers using encryption may experience an approximate ten percent performance penalty when recalling encrypted VTVs. See the following for discussion on the functional differences between the Oracle Cloud offerings:

- "Oracle Storage Cloud Service – Object Storage"
- "Oracle Storage Cloud Service – Archive Storage"
- "Oracle Cloud Encryption (Support for VLE 1.5.3 and Higher)"

Oracle Storage Cloud Service – Object Storage

With support for VLE 1.5.2 and higher, storing data in the Oracle Cloud is much like storing data in the VLE local disk pool. The following steps outline what is needed to configure a VLE for storing a virtual tape volume (VTV) in Oracle Cloud.

The following information is required:

Note: The Oracle CSE must retrieve the customer's Oracle Cloud account information to create the initial connection between the VLE and Oracle Cloud.

- Account Name
- User Name
- User Password
- Authorization URL

MVC ranges are determined by the customer. They are used to configure VTCS host software and provided to the Oracle support team for configuration of the VLE. If the VLE will store VTV data on its local disk pool as well as the Oracle Cloud, there must be two VMVC pool ranges defined and configured in the VLE:

- A vMVC range for VLE local disk pool storage
- A vMVC range for the VLE Oracle Cloud storage

Once VMVC definitions are configured in the VLE, VTV Migrate, Recall, and VLE Copy operations can be expected behave much the same as all VLE operations that use VLE local storage pool. The performance of VLE to cloud data transfer performance is subject to IP bandwidth and delay as well as Oracle Cloud performance capabilities.

Oracle Storage Cloud Service – Archive Storage

With support for VLE 1.5.3 and higher, storing data in the Oracle Cloud is much like storing data in the VLE local disk pool but there are some exceptions regarding a recall of data stored in the Cloud Archive. The steps for setting up the VLE for using the Oracle Storage Cloud Service – Archive Storage is similar to the steps for Oracle Cloud.

The following information is required:

Note: The Oracle CSE must retrieve the customer's Oracle Cloud account information to create the initial connection between the VLE and Oracle Cloud. Cloud Archive account information is the same as Oracle Cloud account information.

- Account Name
- User Name
- User Password
- Authorization URL

MVC ranges are determined by the customer. They are used to configure VTCS host software and given to the Oracle support team for configuration of the VLE. The customer must provide up to three vMVC ranges when using Cloud Archive:

- A vMVC range for VLE local disk pool storage
- A vMVC range for the VLE Storage Cloud
- A vMVC range for the VLE Cloud Archive storage

When creating vMVCs on the VLE the Oracle support person selects an 'archive' flag for vMVCs that will use Cloud Archive. This is what triggers the 'archive' functionality within the Oracle Cloud. Once VMVC definitions are configured in the VLE, VTV Migrate, Recall, and VLE Copy operations are possible for all three vMVC ranges but there some exceptions for the Cloud Archive range of vMVCs:

Migrate

VTV migrate operations perform the same for VTVs migrated to VLE local disk pool or VTVs migrated to the Oracle Cloud Service. Once a VTV is migrated to Oracle Cloud Object Storage, it automatically moves to Oracle Cloud Archive Storage.

Restore and Recall

Once a migrated VTV is moved to Oracle Cloud Archive Storage, you must manually restore the VTV before it can be recalled by VLE. This involves moving the VTV from Oracle Cloud Archive Storage back to Oracle Cloud Object Storage.

Use a `RESTORE_VTV` request to manually restore a VTV from Oracle Cloud Archive Storage. Use a `Route` command to issue this request for the appropriate VLE storage manager.

Depending on your configuration, use one of the following methods to process the `RESTORE_VTV` request:

- In an MVS mainframe configuration:
  - Issue the SMC `Route` command from an MVS console.

    ```
    F ELS73SMC, ROUTE DVTGRD13 RESTORE_VTV VOLUME=5B1307 VTV=CV1234
    ```

  - Issue the SMC `Route` command from the `SMCUUUI` utility. Include the `Route` command in the `UUIIN` data set. Refer to the *ELS command, Control Statement, and Utility Reference* for more information.
  - Issue the SMC `Route` command from the VSM GUI.

    ```
    ROUTE DVTGRD13 RESTORE_VTV VOLUME=5B1307 VTV=CV1234
    ```

- In a VSM 7 Open Systems Attachment (OSA) configuration, issue an oVTCS `Route` command from the VSM GUI.

  From the **VSM Console** menu, select the Command Line Interface and issue the `Route` command in the command window.

  Refer to the *VSM GUI User's Guide* for more information.

Displaying Progress

Issue a `QUERY_RESTORE` request to display progress for the VTVs that are in the restore process. For example:

```
ROUTE DVTGRD13 QUERY_RESTORE VOLUME=5B1307 VTV=CV1234
```

Progress is displayed. For example:

```
Restore initiated via SMCUUI Interface:
```

- `Archived`
- `In Progress`
- `Complete - Restored`
- `Complete - Not Archive`

Once a `Complete` response is received, the VTV can be recalled normally.

> **Note:** Once a VTV is restored, it will remain in Oracle Storage Cloud Service – Object Storage for 24 hours; then it will return to Archive state. Oracle service level agreement (SLA) to restore a VTV is 4 hours. Multiple `RESTORE_VTV` commands can be initiated at the same time.

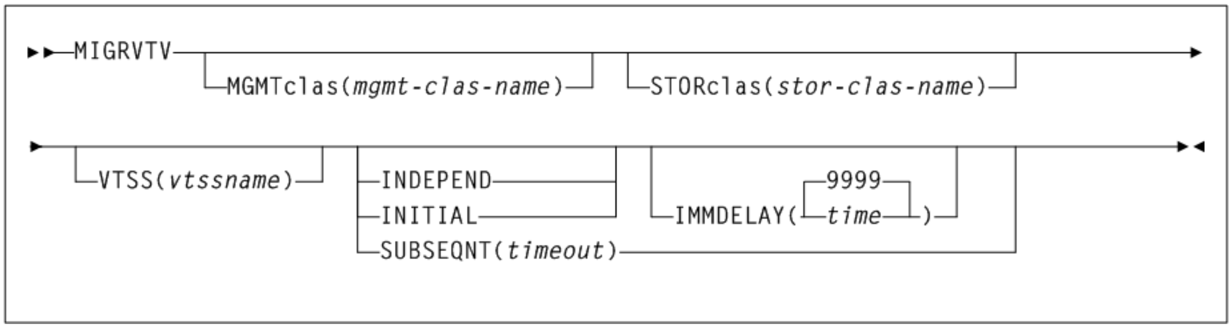
Cloud Archive Guidelines

VLE to Archive Cloud data transfer performance is subject to IP bandwidth and delay as well as Oracle Cloud performance capabilities. The following are general guidelines for the use of Cloud Archive.

- Determine all VTVs to be restored.
  - Determine list of required datasets.
  - Use TMC to determine list of required VTVs.
  - Use `VTVRPT` to determine list of VTVs (and MVC) that only have an Archive Cloud copy.
- If the VTV has not been restored:
  - VLE reports that the VTV on the VMVC is unavailable.
  - VTCS automatically attempts to use another VMVC/MVC copy.
  - Error messages are generated only if all VTV copies are unavailable.
- Learn the costs of the Cloud account.
  - It is not just gets and puts.
  - Early deletes, accessing metadata.
- VLE accesses the Cloud metadata for various operations.
  - Accessing Cloud metadata does have a small cost.
  - Accessing Cloud metadata does not require the VTV to be restored.
- Do *not* perform VLE to VLE copy with a Cloud VMVC as the source.
  - Preference the Cloud VMVC Cloud to be the lowest.
  - Un-configure VLE to VLE connections so that the Cloud cannot be used as a source.
- For recalls, configure VTCS to use Cloud copy as the least preferred copy.
- Avoid `DRAIN`ing an MVC mapped to a Cloud.
- Use `RECLAIM MOVEDATA(NONE)` on Cloud VMVCs.

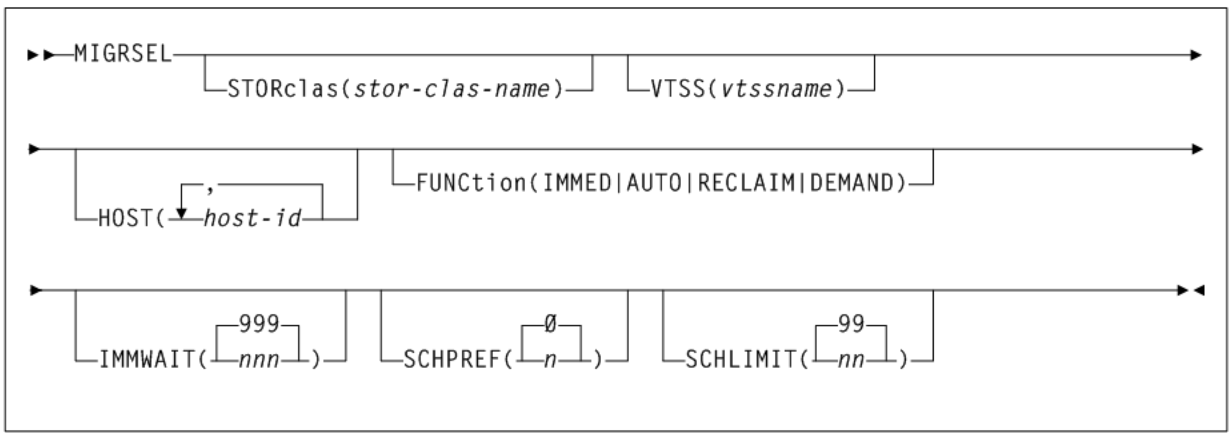
- Use `STORCLAS VLEDELET(RECLAIM)` to reduce average Cloud storage usage.
- Avoid Migrate Causing VLE to VLE copy from a Cloud VMVC.

Figure 3–1 MIGRVTV syntax



  – `MIGRVTV STORCLAS(SCVLE) INITIAL`
  – `MIGRVTV STORCLAS(SCCLOUD) SUBSEQNT(120)`

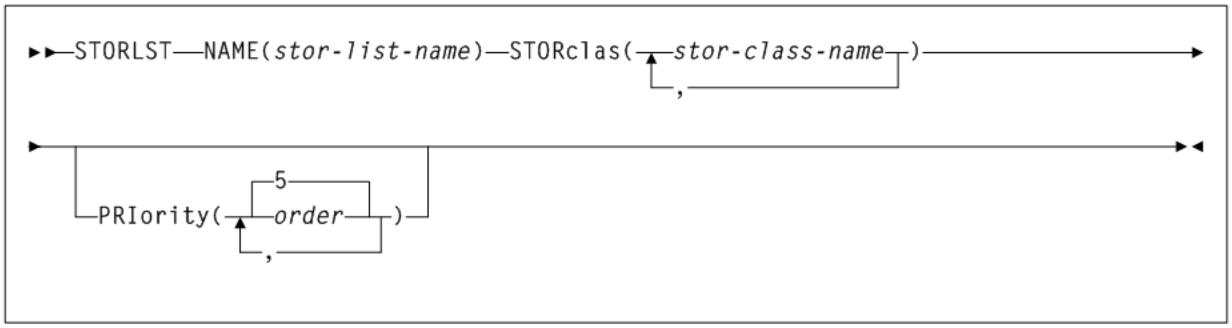
Figure 3–2 MIGRSEL syntax



  – `MIGRSEL STORCLAS(SCTAPE) SCHPREF(9)`
  – `MIGRSEL STORCLAS(SCVLE)  SCHPREF(9)`
  – `MIGRSEL STORCLAS(SCCLOUD) SCHPREF(0)`

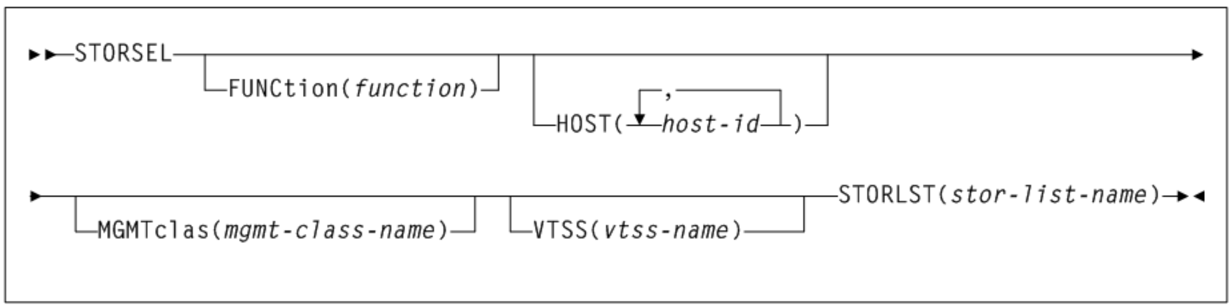
- Avoid Migrate Recalling from a Cloud VMVC.
  – `STORLST NAME(CLDLAST) STORCLAS(SCVLE, SCTAPE, SCCLOUD) PRIORITY(9,8,0)`

Figure 3–3 STORLST syntax



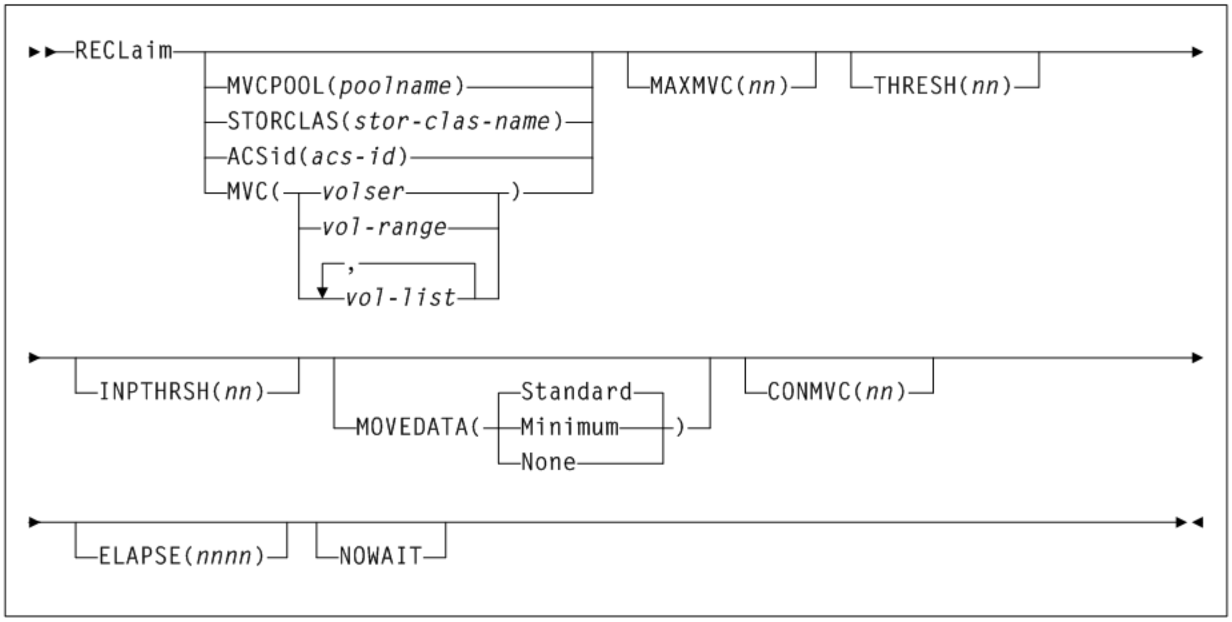
- STORSEL FUNC(SPECIFIC) STORLST(CLDLAST)
- STORSEL FUNC(RECALL) STORLST(CLDLAST)
- STORSEL FUNC(EXPORT) STORLST(CLDLAST)
- STORSEL FUNC(CONSOLID) STORLST(CLDLAST)

Figure 3–4 STORSEL syntax



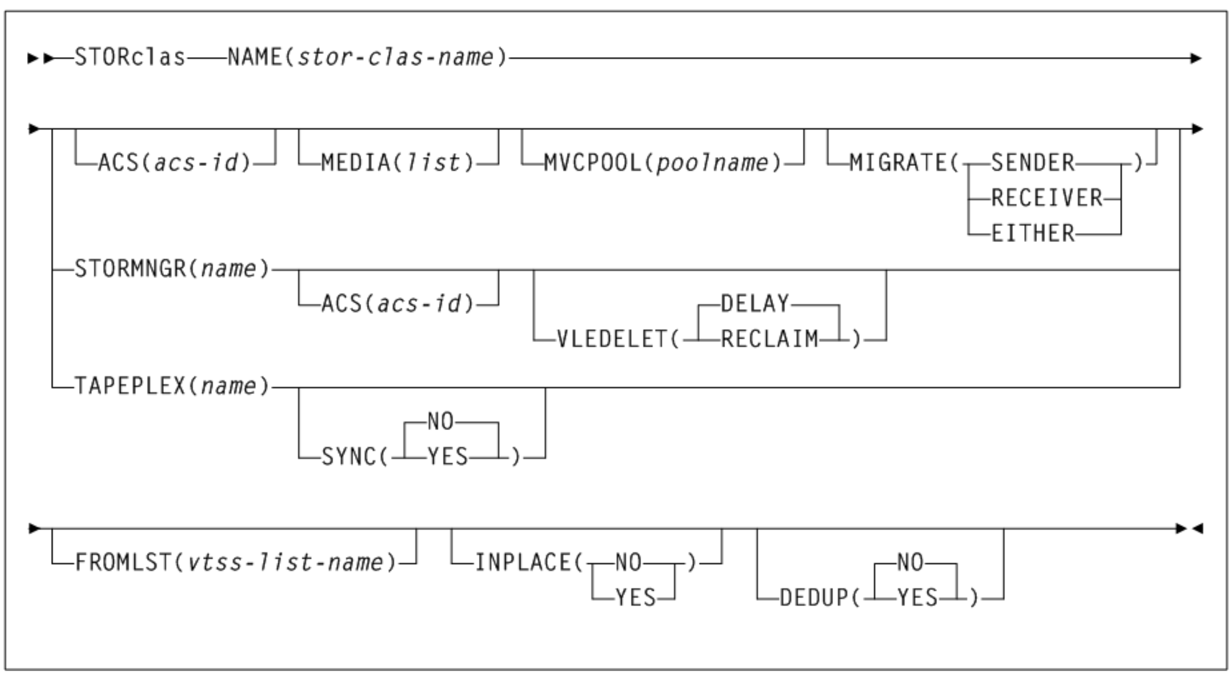
- VLE Reclaim and Explicit Reclaim:
  - Run the `RECLAIM` command
  - `RECLAIM STORCLAS(SCCLOUD)` … `MOVEDATA(NONE)`

Figure 3–5 RECLaim syntax



- VLE `STORCLAS`
  - `STORCLAS (SCCLOUD)` … `VLEDELET(RECLAIM)` …
  - Cloud does not support VTV deduplication (`STORCLAS DEDUP` parameter).
  - Saves the customer money while lowering average monthly storage usage.

Figure 3–6 STORclas syntax



Oracle Cloud Encryption (Support for VLE 1.5.3 and Higher)

Oracle Storage Cloud Service – Object Storage and Oracle Storage Cloud Service – Archive Storage support Encryption. Controlling Encryption in either Oracle Cloud offerings is controlled at the vMVC boundary, that is, if a vMVC is created with the Encryption flag set, all of the VTVs in that vMVC will be encrypted. Migrate and recall operations for encrypted VTVs behave exactly the same for each of the respective Clouds (Archive and non-archive) as described above. The only behavior difference is a performance decrease of 10% for encrypted VTVs. The steps for setting up the VLE for using the Oracle Cloud Encryption are very similar to steps above for Oracle Cloud and Oracle Cloud Archive.

Note: The Oracle CSE must retrieve the customer's Oracle Cloud account information to create the initial connection between the VLE and Oracle Cloud. Cloud Archive account information is the same as Oracle Cloud account information.

The following information is required:

- Account Name
- User Name
- User Password
- Authorization URL

MVC ranges are determined by the customer. They are used to configure VTCS host software and given to the Oracle support team for configuration of the VLE. The customer will need to provide up to three vMVC ranges when using Oracle Cloud with Encryption:

- A vMVC range for VLE local disk pool storage
- A vMVC range for the VLE Oracle Storage Cloud (with or without Encryption)
- A vMVC range for the VLE Cloud Archive (with or without Encryption)

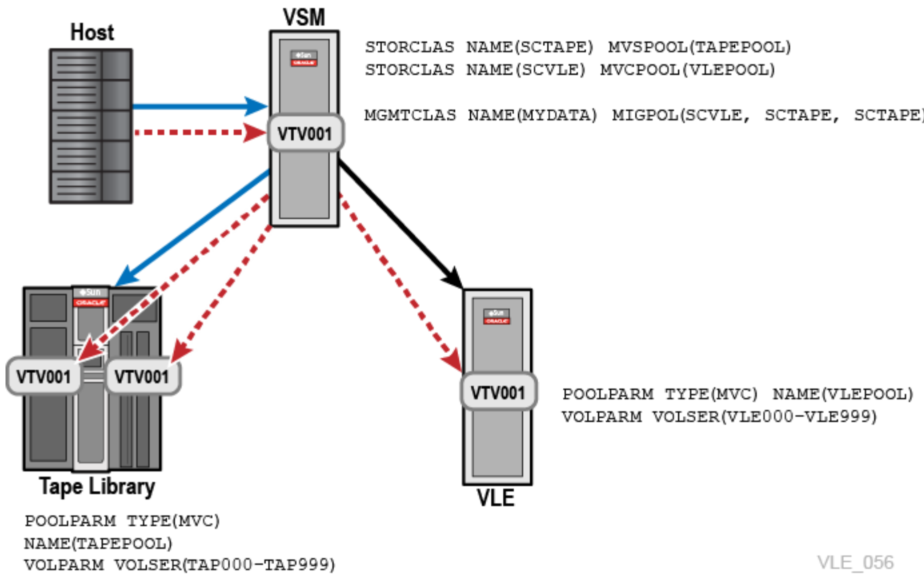
When creating vMVCs on VLE an Oracle support person sets an encryption flag for any vMVCs that will contain encrypted VTVs. Other than the performance there is no difference in the way VTV data is stored (Migrate) and retrieved (Recall) from a VLE or host perspective.

Once VMVC definitions are configured in the VLE, VTV Migrate, Recall, and VLE Copy operations for Encrypted vMVCs behave exactly as described in respective the Oracle Cloud and Cloud Archive in the previous sections. The Oracle Cloud website can be reviewed for information pertaining to the Encryption feature as it is handled within the Oracle Cloud.

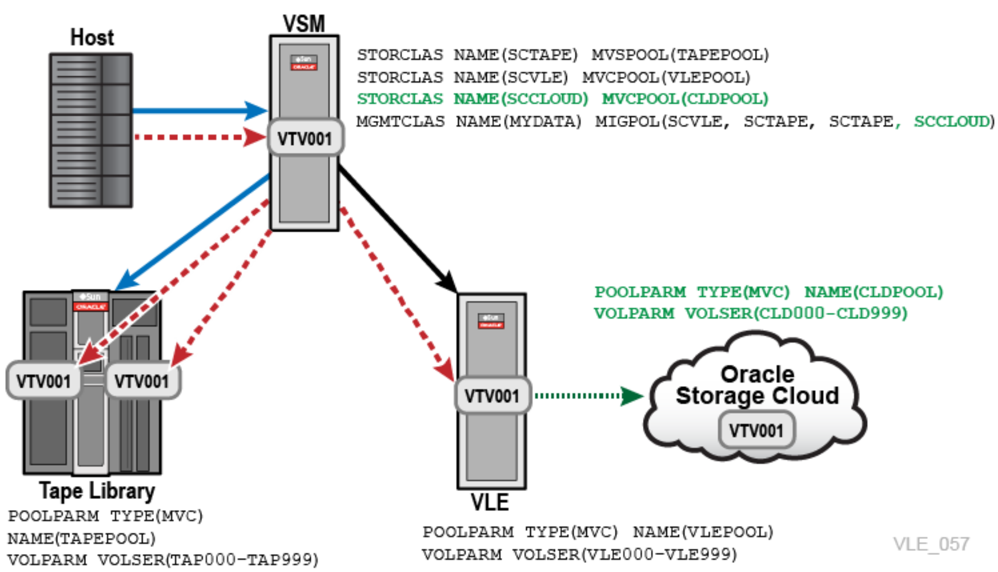
Oracle Cloud Examples Showing VTCS Parameters

The following examples illustrate how a simple VLE configuration might be setup for Oracle Cloud and Cloud Archive including VTCS host parameters.

Example 3–1 Basic VSM/VLE Setup



Example 3–2 Basic VSM/VLE Oracle Cloud setup



Example 3–3 Basic VSM/VLE Oracle Cloud Archive setup