Virtual Tape Control System

Installing and Configuring VTCS

MVS Software

Version 6.2

E21045-02

Revision 02

Submit comments about this document to STP_FEEDBACK_US@ORACLE.COM

Preface

Oracle's StorageTek Virtual Tape Control System 6.2.0 (VTCS 6.2.0, hereafter referred to as "VTCS") is MVS host software, which together with the portions of Nearline Control System (NCS) 6.2.0 that support VTCS and the Virtual Tape Storage Subsystem (VTSS), comprises Virtual Storage Manager (VSM).

Audience

This guide is for StorageTek or customer personnel who are responsible for installing and configuring VTCS. See *VTCS Command and Utility Reference* for information about the following:

- VTCS and NCS (virtual) commands and utilities
- HSC SMF records for VTCS
- VTD commands

Prerequisites

To perform the tasks described in this guide, you should already understand the following:

- MVS, OS/390, or Z/OS operating system
- JES2 or JES3
- System Management Facility (SMF)
- System Modification Program Extended (SMP/E)
- Nearline Control Solution (NCS)

About This Book

For VTCS 6.2, the title is the same, but the book is different. It's been reorganized/rewritten to delete some artifacts that no longer apply, add some information about new procedures and capabilities, and reworked so that the work flow is more straightforward, logical, and complete. This guide now contains the following sections:

- "Planning for Installation" on page 1
- "Planning VTCS Operating Policies" on page 51
- "Preparing for Installation" on page 85
- "Installing VTCS Base" on page 89
- "Reconfiguring HSC" on page 111
- "Configuring VTCS" on page 117
- "Configuring the Host Software for VLE 1.0" on page 131
- "Completing the VSM Configuration" on page 145
- "VSM Configuration Record" on page 165
- "VSM4 ESCON Configuration" on page 167
- "VSM4 FICON Front-End and Back-End Configuration" on page 187
- "VSM5 FICON Configuration" on page 205

What's New in This Guide?

Revision 02

The VTCS 6.2, Revision 02 of this guide contains technical updates and corrections.

Revision 01

The VTCS 6.2, Revision 01of this guide contains information about the VTCS 6.2 enhancements described in TABLE P-2.

TABLE P-1 VTCS 6.2 Updates to Installing and Configuring VTCS, Revision 01

This Enhancement...	...is described in...	...and requires the following PTFs...
Support for T10000C	■ "VTCS Considerations to Correctly Specify MVC Media" on page 25 ■ "Using the STORclas MEDIA Parameter for MVC Media Preferencing" on page 29	■ HSC/MVS - L1H1516 ■ MVS/SMC - L1A00SW ■ MVS/CSC - L1C10AZ ■ VTCS 6.2 - L1H15I4

Revision O

The VTCS 6.2, Revision Oof this guide contains information about the VTCS 6.2 enhancements described in TABLE P-2.

TABLE P-2 VTCS 6.2 Updates to Installing and Configuring VTCS, Revision O

This Enhancement...	...is described in...	...and requires the following PTFs...
Support for VLE 1.0	■ "Configuring the Host Software for VLE 1.0" on page 131 ■ "VMVC Fragmented Space Threshold - Determines VMVC Eligibility for Reclamation" on page 74	■ L1A00R6 ■ L1A00RZ ■ L1A00SY ■ L1H158F ■ L1H158G ■ L1H158H ■ L1H155S ■ L1H155T ■ L1H15G9 ■ L1H15H0 ■ L1H15H7 ■ L1H15NA ■ L1H15QL

Revision N

Revision N contains technical updates and corrections.

Revision M

Revision M contains technical updates and corrections.

Revision L

Revision L contains technical updates and corrections.

Revision K

The VTCS 6.2, Revision K of this guide contains information about the VTCS 6.2 enhancements described in TABLE P-7.

TABLE P-3 VTCS 6.2 Updates to Installing and Configuring VTCS, Revision K

This Enhancement...	...is described in...	...and requires the following PTFs...
Native IP VTV Replication	"CONFIG Example: Defining Native IP Connections" on page 128	**L1H153L, L1H153M and L1A00QO**
Migration control enhancements	"Controlling Migration Workloads" on page 69	L1H14M8 (SWS620) and L1H14MA (SOS620)
CONFIG GLOBAL MAXVTVSZ parameter	"Maximum VTV Size" on page 60	L1H153L, L1H153M and L1A00QO

Revision J

The VTCS 6.2.0, Revision J of this guide contains information about the VTCS 6.2 enhancements described in TABLE P-4.

TABLE P-4 VTCS 6.2.0 Updates to Installing and Configuring VTCS, Revision J

This Enhancement...	...is described in...	...and requires the following...
VSM5 ESCON channel support	"VSM5 ESCON/FICON Configurations" on page 223	VTSS microcode level D02.07.00.00 or H01.07.00.00.
new VSM5 models	"VSM5 New Models" on page 8	See "VSM5 New Models" on page 8
"Tapeless" VSM	"Tapeless VSM" on page 235	■ L1H14XS - SMS6200 ■ L1H14XT - SOS6200 ■ L1H14Y7 - SWS6200

Revision I

The VTCS 6.2.0, Revision I of this guide contains information about the VTCS 6.2 enhancements described in TABLE P-5.

TABLE P-5 VTCS 6.2.0 Updates to Installing and Configuring VTCS, Revision I

This Enhancement...	...is described in...	...and requires the following VSM4/VSM5 microcode...
■ **Up to a total of 16 simultaneous NearLink I/O transfers**, which can be spread across multiple targets on as many as 14 NearLink ports. ■ **Up to a total of 2 simultaneous NearLink I/O transfer**s are allowed per port.	■ "VSM4 FICON Front-End and Back-End Configuration" on page 187 ■ "VSM5 FICON Configuration" on page 205	D02.06.00.00 or higher

Revision H

The VTCS 6.2.0, Revision H of this guide contains information about the VTCS 6.2 enhancements described in TABLE P-6.

TABLE P-6 VTCS 6.2.0 Updates to Installing and Configuring VTCS, Revision H

This Enhancement...	...is described in...	...and requires the following PTFs...
MVC Initialization	"MVC Initialization on First Mount" on page 80	L1A00OO (SMC), L1H14DE (SMS), L1H14DF (SOS), L1H14DG and L1H14H5 (SWS)

Revision G

Revision G contains technical updates and corrections.

Revision F

The VTCS 6.2.0, Revision F of this guide contains information about the VTCS 6.2 enhancements described in TABLE P-7.

TABLE P-7 VTCS 6.2.0 Updates to Installing and Configuring VTCS, Revision F

This Enhancement...	...is described in...
Maximum 32 RTDs	■ "Maximum RTDs per VTSS" on page 62 ■ "VSM5 FICON Configuration" on page 205
Stacked Migrates	■ "Stacked Migrates" on page 68
T9840D/T10000B Support	■ "VTCS Considerations to Correctly Specify MVC Media" on page 25 ■ "Using the STORclas MEDIA Parameter for MVC Media Preferencing" on page 29

32 RTDs/Stacked Migrates support, which applies to only VSM5s, has the requirements described in TABLE P-8.

TABLE P-8 32 RTDs/Stacked Migrates Support Requirements for VTCS/NCS 6.2

32 RTDs/Stacked Migrates Support requires...	..the following VSM4/VSM5 microcode...	...and the following VTCS/NCS 6.2 PTFs...	...and CDS level...
FICON RTDs and FICON ports for the CLINKs	D02.05.00.00 or higher	L1H13ZF (SOS6200) L1H13ZG (SWS6200	"F" or higher

Note – T9840D support requires the following PTFs:

- NCS/VTCS 6.2- L1H14EP (SOS6200), L1H13ZC (SWS6200), and L1A00P0 (SMC6200).

T10000B support requires the following PTFs:

- NCS/VTCS 6.2 - L1H142E (SOS6200) and L1H142C (SWS6200)

Revision E

The VTCS 6.2.0, Revision E of this guide contains information about the VTCS 6.2 enhancements described in TABLE P-9.

TABLE P-9 VTCS 6.2.0 Updates to Installing and Configuring VTCS, Revision E

This Enhancement...	...is described in...
VTSS Synchronous Replication	"VTCS Replication - Synchronous or Asynchronous" on page 83

Synchronous replication, which applies to only VSM4s and VSM5s, has the requirements described in TABLE P-10.

TABLE P-10 Sychronous Replication Requirements for VTCS/NCS 6.2

Synchronous replication requires...	..the following VSM4/VSM5 microcode...	...and the following VTCS/NCS 6.2 PTFs...	...and CDS level...
FICON ports for the CLINKs	D02.03.00.00 or higher	L1H13QL (SWS6200), L1A00L3 (SMC6200), and L1H13K8 (SOS6200)	"F" or higher

Revision D

The Revision D of this guide contains technical updates and corrections.

Revision C

The Revision C of this guide contains the new information or updates described in Table 11.

TABLE P-11 Updates to VTCS Installation and Configuration Guide, Revision C

This information...	...is described in...	...and is available via...
Up to 14 Nearlink FIPs supported	• "VSM4 FICON Front-End and Back-End Configuration" on page 187 • "VSM5 FICON Configuration" on page 205	For VSM4s/VSM5s: microcode level D02.02.00.00 or higher

TABLE P-11 Updates to VTCS Installation and Configuration Guide, Revision C

This information...	...is described in...	...and is available via...
Standard/Large VTV Pages	"CDS VTCS Level" on page 32 "VTV Page Size" on page 59	■ For VSM3s: microcode level N01.00.77.00 or higher. ■ For VSM4s/VSM5s: microcode level D02.02.00.00 or higher.
400Mb/800Mb/2Gb/4gb VTVs	"CDS VTCS Level" on page 32 "Maximum VTV Size" on page 60	
65000 VTVs per MVC	"CDS VTCS Level" on page 32 "Maximum VTVs per MVC" on page 70	

Revision B

The Revision B of this guide contains the new information or updates described in Table 12.

TABLE P-12 Updates to VTCS Installation and Configuration Guide, Revision B

This information...	...is described in...	...and is available via...
T10000 Encryption transports and media	"VTCS Considerations to Correctly Specify MVC Media" on page 25	6.1 with the following PTFs: ■ L1H13AA (SOS6100) ■ L1H136K (SWS6100) 6.2 with the following PTFs: ■ L1H139D (SOS6200) ■ L1H139C (SWS6200)

Revision A

The VTCS 6.2.0, Revision A of this guide contains information about the VTCS 6.2 enhancements described in TABLE P-13.

TABLE P-13 VTCS 6.2.0 Updates to Installing and Configuring VTCS 6.2, Revision A

This Enhancement...	...is described in...
VSM4 FICON Back-End support	"VSM4 FICON Front-End and Back-End Configuration" on page 187
VTCS Locks in a Coupling Facility	"Storing VTCS Locks in a Coupling Facility (Optional)" on page 35

Contents

Preface iii

Audience iii

Prerequisites iii

About This Book iv

What's New in This Guide? v

Revision 02 v

Revision 01 v

Revision O v

Revision N vi

Revision M vi

Revision L vi

Revision K vi

Revision J vii

Revision I vii

Revision H viii

Revision G viii

Revision F viii

Revision E ix

Revision D ix

Revision C x

Revision B x

Revision A xi

1. Planning for Installation 1

Verifying VSM Software and Hardware Prerequisites 5

VTCS System Software Requirements 5

CDS Locations 6

Third Party Tape Copy Software for Migrating Data to VSM 6

Nearline Hardware Requirements 7

VSM5 New Models 8

How VSM Measures Sizes and Capacities 10

Determining VSM Configuration Values 11

HSC and SMC Definition Data Set Names 11

VTSS Names 12

VTD Unit Addresses 13

VSM Esoterics and Esoteric Substitution 14

General Guidelines and Requirements for VSM Esoterics 14

VSM Esoterics and Esoteric Substitution for SMC TAPEREQ Statements and POLICY Commands 15

VSM Esoterics and Esoteric Substitution for the StorageTek DFSMS Interface 15

Designing VSM Esoterics for Esoteric Substitution 16

VTV Definitions 19

RTD Definitions 21

MVC Definitions 22

Define and Select Nearline Volumes 22

Define Available MVCs with CONFIG 23

Define the MVC Pool 23

Protect MVCs and Nearline Volumes 24

VTCS Considerations to Correctly Specify MVC Media 25

Using the STORclas MEDIA Parameter for MVC Media Preferencing 29

CDS VTCS Level 32

Guidelines for Changing CDS VTCS Levels 34

Storing VTCS Locks in a Coupling Facility (Optional) 35

When to implement VTCS locks in a Coupling Facility Structure 35

Requirements 36

Sizing the Coupling Facility Structure 36

Defining the Coupling Facility Structure to MVS 36

t Reverting to Storing VTCS Lock Data in the CDS 37
Managing Failures/Unavailability of the VTCS lock structure 38
Virtual ACS IDs 39
Virtual ACS Location for VSM2s and VSM3s 39
Virtual ACS Location for VSM4s 42
HSC CDS DASD Space 49
Tape Management System DASD Space 49
VSM Candidate Data Sets 49
HSC COMMPath METHod Value 49
Data Chaining a VTD Read Forward or Write Command 49

2. Planning VTCS Operating Policies 51

VTSS Policies 52
AMT Settings 53
53
Usage Notes 53
Deleting Scratched VTVs 55
Usage Notes 55
VTV Residency Interval before Automatic Migration Candidacy 56
Usage Notes 56
Immediately Migrate VTVs On Dismount 58
Usage Notes 58
MVC Retain Interval 59
VTV Page Size 59
Usage Notes 59
Maximum VTV Size 60
Usage Notes 60
Maximum RTDs per VTSS 62
Usage Notes 62
RTD/CLINK Addresses - Maximum 32 RTDs 63
Maximum and Minimum Concurrent Migration Tasks 67
Usage Notes 67
Stacked Migrates 68
Usage Notes 68

Controlling Migration Workloads 69

Usage Notes 69

VTV Policies 70

Maximum VTVs per MVC 70

Usage Notes 70

Hosts Disabled from Migration, Consolidation and Export by VTV or Management Class 71

Usage Notes 71

Recall VTVs with Read Data Checks 71

MVC Policies 72

MVC Fragmented Space Threshold - Determines MVC Eligibility for Reclamation 73

Usage Notes 73

VMVC Fragmented Space Threshold - Determines VMVC Eligibility for Reclamation 74

Usage Notes 74

Free MVCs Threshold - Starts Automatic Space Reclamation 74

Usage Notes 74

Eligible/Total MVCs Threshold - Starts Automatic Space Reclamation 76

Usage Notes 76

Maximum MVCs Processed Per Reclaim 77

Usage Notes 77

Maximum MVCs Concurrently Processed for Reclamation and Drain 78

Hosts Disabled from Reclamation 78

Usage Notes 78

MVC Retain Interval 79

Usage Notes 79

MVC Initialization on First Mount 80

VTV Replication Policies 81

VTCS Replication - Always or Changed VTV Only 82

Usage Notes 82

VTCS Replication - Synchronous or Asynchronous 83

Usage Notes 83

3. Preparing for Installation 85

Defining A Security System User ID for HSC, SMC, and VTCS 86
Configuring MVS Device Numbers and Esoterics 87
Assigning MVS Device Numbers to VTDs 87
Associating VTD MVS Device Numbers and Esoterics 87
Assigning MVS Device Numbers to RTDs 87
Setting the MVS Missing Interrupt Handler (MIH) Value 88
Specifying the Region Size 88

4. Installing VTCS Base 89
Ensuring that HSC and SMC Are Installed 90
SMC Installation Considerations 90
Reviewing Coexistence Requirements 91
Verifying Base Installation Materials 92
VTCS Base Installation Media Contents 93
Base Media Contents 93
VTCS FMIDs 94
Allocating VTCS Target and Distribution Library Data Sets and Required DDDEF Entries 94
Installing the VTCS Base from Tape 95
t Unloading the SMP/E JCL Library 95
t Receiving the VTCS 6.2.0 FMID from Tape 95
Installing VTCS Base from CD-ROM 96
t Unloading the VTCS Base Installation CD 96
t FTPing VTCS62.pax to USS 96
t Unpacking the VTCS62.pax File 97
t TSO RECEIVEing the LOADSAMP.xmit file 98
t SMP/E Receiving the VTCS Functions from the VTCS Installation CD 98
Applying the VTCS 6.2.0 FMID 99
Accepting the VTCS 6.2.0 FMID 100
Adding SWSLINK to the Authorized Program List 101
Using IEAAPFxx to APF Authorize the SWSLINK 101
Using PROGxx to APF Authorize SWSLINK 101
Modifying the HSC Startup Procedure to include the VTCS 6.2.0 LINKLIB 102
Installing VTCS 6.2 Maintenance 103

5. Installing VTCS Maintenance 105

Verifying Service Media 105

VTCS Service Media Contents 106

Unloading the VTCS Service Tape 107

Unloading the VTCS Service CD-ROM 107

Receiving the VTCS 6.2 Maintenance 108

Applying the VTCS 6.2 Maintenance 109

Accepting the VTCS 6.2 Maintenance 109

6. Reconfiguring HSC 111

Creating or Updating the HSC LIBGEN 112

Verifying the LIBGEN 113

Formatting the New CDS 114

Creating VOLATTR Statements for VTVs 115

Defining MVCs to HSC 115

7. Configuring VTCS 117

Building a Simple CONFIG Deck 119

t To build a simple CONFIG deck: 119

Special Cases: A Gallery of Advanced Uses of CONFIG 123

CONFIG Example: All Hosts Access VTDs in One VTSS, Only Selected Hosts Access VTDs in Second VTSS 124

CONFIG Example: Update Configuration to Add RTDs 125

CONFIG Example: Update Configuration to Add MVCs and VTVs and Change AMTs 126

CONFIG Example: Denying Host Access to a Physically Removed VTSS 127

CONFIG Example: Defining Native IP Connections 128

129

8. Configuring the Host Software for VLE 1.0 131

Key Configuration Values 132

Subsystem Name 132

VLE Data Port and VSM5 IFF3 Card Target IP Addresses 132

IP Addresses of VLE Ports for Host (UUI) Communication 132

VMVC Volsers 132

Common Configuration Tasks (All Examples) 133
- Creating VMVC Volume Pools 133

Example 1: One VTSS Connected to One VLE Appliance, Direct Connect (No Switch) 134
- Configuring the System 136
- Defining Policies 138

Example 2: Four VTSSs Connected to One VLE Appliance, Direct Connect (No Switch) 139
- Configuring the System 141
- Defining Policies 143

9. Completing the VSM Configuration 145

Updating the HSC PARMLIB Member (SLSSYSxx) 146

Adding SMF Parameters for VTCS to SYS1.PARMLIB 147

Connecting MVS/CSC Clients to VSM 148
- To connect MVS/CSC clients to VSM: 148

Using the SMC Client/Server Feature 150

Converting HSC/SMC on Multiple LPARs to SMC Client/Server 153
- To convert from HSC/SMC on multiple LPARs to SMC client/server: 153

Converting from MVS/CSC and LibraryStation 154

Connecting Non-MVS/CSC Clients to VSM 155

Updating the Tape Management System 156

Defining VSM Security 157

Defining MVC Pool Volser Authority 157

Defining VTCS Command Authority 158

Updating HSM 159

Routing Data Sets to VSM 160

The StorageTek DFSMS Interface 161

SMC TAPEREQ Statements 162

HSC User Exits 162

MVS/CSC User Exits 163

Restarting NCS/VTCS 164

A. VSM Configuration Record 165

B. VSM4 ESCON Configuration 167

VSM4 with 32 Ports 169

VSM4 Configuration Examples - 32 Ports 172
VSM4 Configuration Example: 16 Host Ports, 16 RTD Ports 173
VSM 4 Configuration Example: 20 Host Ports, 12 RTD Ports 175
VSM4 with 16 Ports 177
VSM4 Configuration Examples - 16 Ports 179
VSM4 Configuration Example: 8 Host Ports, 8 RTD Ports 180
VSM 4 Configuration Example: 10 Host Ports, 6 RTD Ports 182
CONFIG Example for VSM4 with 10 Host Ports, 6 RTD Ports 183
IOCP Example for Single MVS Host Connected to a VSM4 Via ESCON Directors 184
Logical Paths for VSM 4 with 32 Ports 186

C. VSM4 FICON Front-End and Back-End Configuration 187
VSM4 FICON VCF Card Options 188
VSM4 FICON Front-End and Back-End Configuration Examples 194
VSM4 Configuration Example: 8 VCF Cards, FICON Directors, 16 RTDs 195
CONFIG Example for VSM4 FICON with 8 VCF Cards, FICON Directors, 16 RTDs 196
VSM4 Configuration Example: 8 VCF Cards, 4 CLINKs, FICON Directors for 8 RTDs 197
CONFIG Example for Bi-Directional Clustered VSM4 FICON Back-End 198
IOCP Example for Single MVS Host Connected to a VSM4 Via FICON Directors 200

D. VSM5 FICON Configuration 205
VSM5 FICON VCF Card Options - Maximum 16 RTDs 206
VSM5 FICON VCF Card Options - Maximum 32 RTDs 209
FICON Port Processing 210
FICON Port Operations Best Practices 211
VSM5 FICON Front-End and Back-End Configuration Examples 212
t Implementing Support for Maximum of 32 RTDs 212
VSM5 Configuration Example: 8 VCF Cards, FICON Directors, 32 RTDs 213
CONFIG Example for VSM5 FICON with 8 VCF Cards, FICON Directors, 32 RTDs 214
VSM5 DOP Panels for Maximum 32 RTDs 215
Channel Configuration Status Screen 215
Channel Configuration and RTD Path Validation Screen 216

Real Tape Drive Status Screen 217

IOCP Example for Single MVS Host Connected to a VSM5 Via FICON Directors 218

E. VSM5 ESCON/FICON Configurations 223

VSM5 ICE/VCF Card Options 224

VSM5 Configuration Example: 8 ICE Cards, 16 Host Ports, 16 RTD Ports 229

IOCP Example for Single MVS Host Connected to a VSM5 Via ESCON Directors 231

F. Tapeless VSM 235

How Does Tapeless VSM Work? 236

Example: Mixed Tapeless VSM 237

- Configuring the System 238
- Defining Policies 239

LIBGEN Example for Tapeless ACS 241

Glossary 243

A 244

B 245

C 246

D 248

E 249

F 250

G 251

H 251

I 252

J 253

K 253

L 254

M 255

N 256

O 256

P 257

R 258

S 258

T 260

U 261

V 262

W 263

WTOR Write–to–Operator with reply. 263

Symbols 263

Numerics 263

CHAPTER **1**

Planning for Installation

Planning...yes, we know, the activity we all know we need to do most but want to do least. Most of this is pretty dry stuff, so I'm not going to try real hard to spice it up.

As a time-saver, what you *should* do before diving into the bits and bytes is take a quick look at the checklist in TABLE 1-1. This table is designed to help plan and verify completion of your system's installation and configuration tasks, and if you look at the notes, you'll see that, depending on your situation (new or upgrade install, adding hardware or not, etc.), you may not have to do any more than say "Yep, been there, done that" for a specific task and check it off...

TABLE 1-1 VTCS Installation Summary and Checklist

Task	Notes	✓ to Verify Completion
"Verifying VSM Software and Hardware Prerequisites" on page 5	Ensure that you have the prerequisites. for VTCS 6.2 and the features and hardware you intend to use.	
"Determining VSM Configuration Values" on page 11 "HSC CDS DASD Space" on page 49	Plan configuration values...a must do, because you can't just take default here. **Note especially!** The VSM Extended Format CDS requires additional DASD space!	
"Planning VTCS Operating Policies" on page 51	You have a choice...that is, you can take the defaults and worry about tuning up these knobs and dials later. If you take that route, it's not a bad idea to take a high level pass through this chapter with an eye toward reading it thoroughly when you have the time...	

TABLE 1-1 VTCS Installation Summary and Checklist

Task	Notes	✓ to Verify Completion
Chapter 3, "Preparing for Installation"		
	None of these tasks is difficult, but they are **all** critical. For example, Depending on the default settings of your security system, VSM may not be able to mount and to write to MVCs until you have defined a security system user ID for HSC and TAPEVOL profiles for the MVCs!	
"Defining A Security System User ID for HSC, SMC, and VTCS" on page 86		
"Configuring MVS Device Numbers and Esoterics" on page 87		
"Setting the MVS Missing Interrupt Handler (MIH) Value" on page 88		
"Specifying the Region Size" on page 88		
Chapter 4, "Installing VTCS Base"		
"Ensuring that HSC and SMC Are Installed" on page 90	Basic SMP/E installation. **Note that** order is important, and you have to coordinate VTCS installation with NCS (including SMC) installation. As you can see, you now have a choice of installation media and procedure.	
"Reviewing Coexistence Requirements" on page 91		
"Verifying Base Installation Materials" on page 92		
"Installing VTCS Base from CD-ROM" on page 96 OR "Unloading the SMP/E JCL Library" on page 95		
"Allocating VTCS Target and Distribution Library Data Sets and Required DDDEF Entries" on page 94		
"Applying the VTCS 6.2.0 FMID" on page 99		
"Accepting the VTCS 6.2.0 FMID" on page 100		
"Adding SWSLINK to the Authorized Program List" on page 101		
"Modifying the HSC Startup Procedure to include the VTCS 6.2.0 LINKLIB" on page 102		
Chapter 5, "Installing VTCS Maintenance"		
"Verifying Service Media" on page 105	All the good stuff about applying cumulative maintenance from the service media...something you must do after installing VTCS 6.2 Base...	
"Unloading the VTCS Service CD-ROM" on page 107		
"Receiving the VTCS 6.2 Maintenance" on page 108		
"Applying the VTCS 6.2 Maintenance" on page 109		
"Accepting the VTCS 6.2 Maintenance" on page 109		
Chapter 6, "Reconfiguring HSC" **Tip –** Optionally, you can do all these tasks **before** you install VTCS.		
"Creating or Updating the HSC LIBGEN" on page 112	**Note that** if your system's RTDs are new transports, you must update the HSC LIBGEN by adding a SLIDRIVS macro to define the device addresses you determined.	
"Verifying the LIBGEN" on page 113	...as you do with any new or updated LIBGEN...	

TABLE 1-1 VTCS Installation Summary and Checklist

Task	Notes	✓ to Verify Completion
"Formatting the New CDS" on page 114	If required, this is where you use the HSC SLICREAT macro to format the CDS to the larger size you determined in the planning section.	
"Creating VOLATTR Statements for VTVs" on page 115	In Chapter 7, "Configuring VTCS", you define VTVs and MVCs to VTCS via the CONFIG utility, but you also need to define these entities to HSC.	
"Defining MVCs to HSC" on page 115		

TABLE 1-1 VTCS Installation Summary and Checklist

Task	Notes	✓ to Verify Completion
Chapter 7, "Configuring VTCS"		
"Building a Simple CONFIG Deck" on page 119	A step procedure to create a plain-vanilla, works every time CONFIG deck.	
"Special Cases: A Gallery of Advanced Uses of CONFIG" on page 123	Variations on the theme...	
Chapter 9, "Completing the VSM Configuration"		
"Updating the HSC PARMLIB Member (SLSSYSxx)" on page 146	This is where your "DEF" statements reside, plus other critical items such as the COMMPATH and FEATURES statements. As above, don't overlook this step...	
"Adding SMF Parameters for VTCS to SYS1.PARMLIB" on page 147	Technically speaking, this is optional...but **highly recommended**, because you need the SMF information to know how your system is performing.	
"Connecting MVS/CSC Clients to VSM" on page 148	You only need to do this if you are using MVS/CSC on remote hosts. **Note that**, for VTCS/NCS 6.2, you can simply install SMC 6.2 in your client MVS system, and SMC will route virtual allocation and mount requests to HSC running in a remote server HSC system. For more information, see "Using the SMC Client/Server Feature" on page 150.	
"Connecting Non-MVS/CSC Clients to VSM" on page 155	You only need to do this if you want to connect non-MVS/CSC 4.0 and above clients to VSM and define LibraryStation subpools that contain VTVs. Contact StorageTek Software Support for information on the supported clients.	
"Updating the Tape Management System" on page 156	You might not have to do anything here, unless you're adding VTV ranges or MVCs...but read through this section to make sure everything's set up correctly with your TMS(s)...	
"Defining VSM Security" on page 157	Required to ensure that the correct personnel and applications have access to the VSM resources required.	
"Updating HSM" on page 159	...if you're an HSM user, and are routing HSM jobs to VSM...	
"Routing Data Sets to VSM" on page 160	The part we've all been waiting for. If you're doing an upgrade install, you might not have to do anything here, but read through this in case you have some new or changed jobs coming to VSM. There are some changes in the way things work (hint: it's simpler), which you can probably put to good use...	
"Restarting NCS/VTCS" on page 164	...to make the NCS/VTCS reconfiguration take effect...	

Verifying VSM Software and Hardware Prerequisites

VTCS System Software Requirements

Verify the software prerequisites for VTCS 6.2.0 listed in TABLE 1-2.

TABLE 1-2 VTCS 6.2.0 Minimum Software Requirements

Software Description	Minimum Version/Release
Operating System	▪ z/OS v1.3.0, v1.4.0, v1.5, v1.6, v1.7; ▪ OS/390 v1.2 and higher. ▪ MVS/ESA v5.1 and higher. ▪ JES2 v1.4 and higher. ▪ JES3 v1.4 and higher. JES3 support for local and global configurations is supported. ▪ Sysplex environment is supported.
Nearline Control Solution	NCS 6.2 **Note:** ▪ VTCS 6.2.0 requires HSC 6.2.0 and will not run with previous versions of HSC. ▪ If you are using RMM in an MVS/CSC environment, MVS/CSC can share the tape management catalog with the host(s) running HSC if you have the following installed: • RMM APAR OA03368 • VSM3 microcode N01.00.65 or later or VSM4 microcode D01.00.03 or later. Otherwise, the tape management catalog cannot be shared or VTV scratch mounts will fail.
Expert Performance Reporter (optional software)	ExPR 4.0
Expert Library Manager (optional software)	To use Expert Library Manager (ExLM) with VSM for VTV consolidation using ExLM, ExLM 4.0, HSC 4.0.0, and VTCS 4.0.0. For more information about using ExLM with VSM, see “Using ExLM to Manage VSM Resources” in Chapter 2 of the ExLM *System Administrator’s Guide*.

CDS Locations

Caution – VSM **does not** support copies of the CDS at multiple sites (for example, Primary CDS at one site and Secondary at another). A link failure would allow the two sites to run independently, and VSM cannot enforce separation of all resources. This prevents reconciliation of the two divergent CDSs as can be accomplished in a pure NCS environment.

Third Party Tape Copy Software for Migrating Data to VSM

TABLE 1-3 lists Third Party tape copy software for migrating data to VSM.

TABLE 1-3 Third Party Tape Copy Software for Migrating Data to VSM

Product Name	**Vendor**
Beta55	Beta Systems Software AG
TelTape/390	Cartagena Software Limited
CA-1®/Copycat	Computer Associates International
CA-Dynam®/TLMS/ Copycat	Computer Associates International
MediaMerge	eMag Solutions
FATSCopy	Innovation Data Processing
Tape/Copy	OpenTech Systems, Inc.
Zela	Software Engineering of America
CARTS-TS TapeSaver	UNICOM Systems, Inc.

Nearline Hardware Requirements

Verify the minimum Nearline hardware requirements listed in TABLE 1-4 and TABLE 1-5.

TABLE 1-4 VSM Nearline Hardware Requirements

Hardware	Requirement
LSMs	Any of the following but SL8500, 9310, or 9740 recommended by StorageTek: 4410, 9310, 9740, 9360, and SL8500
Transports and media	VSM RTDs can be a mixture of 9490 (Timberline), 9490EE (Timberline EE), T9840A, T9840B,T9840C, T9940A, T9940B, and T10000 transports (see TABLE 1-5). Each VTSS must have a minimum of two library-attached transports for each media type used for MVCs. For example, if your MVCs are STANDARD and ECART, you need a minimum of *either* two 9490s *or* two 9490EEs as RTDs. If your MVCs are STANDARD, ECART, ZCART, and STKIR, you need a minimum of two 9490EEs *and* two 9840s as RTDs. **Note:** Using T9940s as RTDs requires HSC 4.0.0 and VTCS 4.0.0 with PTF L1H1043 applied. Valid media types for the supported RTDs are: 9490: STANDARD, ECART 9490EE: STANDARD, ECART, ZCART T9840A, T9840B, T9840C: See "VTCS Considerations to Correctly Specify MVC Media" on page 25. T9940A, T9940B: See "VTCS Considerations to Correctly Specify MVC Media" on page 25.

TABLE 1-5 Prerequisites for T10000 Drives as RTDs

Description	**Requirement**
NCS/VTCS	6.0 with the following PTFs: ▪ L1H12ZI (SOS6000) ▪ L1H12ZJ (SWS6000) 6.1 with the following PTFs: ▪ L1H12ZN (SOS6100) ▪ L1H12ZO (SWS6100) 6.2
LSMs	9310 and SL8500 at LMU Compat Level 13
protocol	FICON
VTSSs	VSM4 and VSM5
media	T10000T1 (full capacity 500GB cartridge) T10000TS (120GB sport cartridge)

VSM5 New Models

The VSM5 provides new models that offer the capacities shown in TABLE 1-6.

TABLE 1-6 VSM5 New Model Capacities

VSM Model	Product Family ID	Drive Capacity	1-Array TBE	2-Arrays TBE	3-Arrays TBE	4-Arrays TBE
VSM5-45TB-IFF3	580	450GB		45		
VSM5-68TB-IFF3	580	450GB			68	
VSM5-90TB-IFF3	580	450GB				90
VSM5-1.25TB-IFF3	567	146GB		7.5/11/14		
VSM5-16TB-IFF3	567	146GB			16/18/21	
VSM5-23TB-IFF3	567	146GB				23/25/28
VSM5E -.8TB	567	146GB	.8/1.25			

Note –

- TBE =The approximate maximum effective capacity in Terabytes (TB).
- Models VSM5-45TB-IFF3, VSM5-68TB-IFF3, and VSM5-90TB-IFF3 require the following PTFs:
  - For 6.2:
    - L1A00Q4 - SMC6200
    - L1H14UL - SMS6200
    - L1H14UK - SWS6200
  - These models can contain a maximum of 500,000 VTVs
- VSM5 new models require the following VTSS microcode:
  - **For Models VSM5-45TB-IFF3, VSM5-68TB-IFF3, and VSM5-90TB-IFF3 only**, VTSS microcode level H02.07.
  - **For all other Models**, VTSS microcode level D02.07.
  - Models VSM5-1.25TB-IFF3, VSM5-16TB-IFF3, and VSM5-23TB-IFF3 support ESCON channel cards. For more information, see "VSM5 ESCON/FICON Configurations" on page 223.

TABLE 1-7 shows the supported channel card configurations for:

- VSM5 - All models **including** VSM5c but **not including** VSM5e and VSM5escon. For information on VSM5escon, see "VSM5 ESCON/FICON Configurations" on page 223.
- VSM5e.

TABLE 1-7 VSM5 New Model Supported Channel Card Configurations

VSM Model	**Storage Cluster 0**				**Storage Cluster 1**			
	2	**3**	**4**	**5**	**12**	**13**	**14**	**15**
5	**VCF**		**VCF**		**VCF**		**VCF**	
	VCF	**VCF**	**VCF**		**VCF**	**VCF**	**VCF**	
	VCF	**VCF**	**VCF**	**VCF**	**VCF**	**VCF**	**VCF**	**VCF**
5e	**VCF**		**VCF**		**VCF**		**VCF**	
	ICE		**ICE**		**ICE**		**ICE**	

How VSM Measures Sizes and Capacities

VTCS uses the binary standard rather than the decimal standard in displaying and calculating sizes and capacities for VTVs and MVCs. Thus:

- 1 kilobyte(KB)=1024 bytes
- 1 megabyte(MB)=1000 kilobytes or 1000*1024 byte
- 1 gigabyte(GB)=1000 megabytes or 1000*1000*1024 bytes

Note, however, that VTCS uses the decimal standard in displaying and calculating sizes and capacities for VTSSs. Thus:

- 1 kilobyte(KB)=1000 bytes
- 1 megabyte(MB)=1000 kilobytes or 1000*1000 bytes
- 1 gigabyte(GB)=1000 megabytes or 1000*1000*1000 bytes

Determining VSM Configuration Values

The following sections tell how to determine configuration values for your VSM system. Use TABLE A-1 on page 165 to record these values. This table also provides a record of your site's VSM configuration, which can help you and StorageTek service troubleshoot problems with your VSM system.

Note – Unless otherwise noted, in each of the following sections, the values you determine must match wherever you use them. For example, the unit addresses described in "RTD Definitions" on page 21 must match on the following:

- The HSC SLIDRIVS macro.
- If you will share these transports with MVS, when you assign MVS device addresses to these transports via the HCD facility.

HSC and SMC Definition Data Set Names

Determine the names of the HSC data sets that will contain your VSM system's VOLATTR, MVCPool, MGMTclas and STORclas statements. MGMTclas and STORclas statements must reside in the same file (sequential data set or PDS member) for cross-validation.

VTSS Names

Determine your system's 1 to 8 character VTSS names, which you specify when you run VTCS CONFIG to initially install and configure your VSM system as described in Chapter 7, "Configuring VTCS".

Caution – Note the following:

- The VTSS name can consist of the characters "A-Z", "0-9", "@", "$", and "#".
- You specify the VTSS name *only* via the NAME parameter, which sets the VTSS name in both the VTSS microcode (as displayed in the Subsystem Name field in the LOP or VOP) and in the configuration area of the HSC CDS. After VSM is put into operation, the VTSS name is also stored in each VTV record in the CDS. Each VTV record contains the VTSS name on which that VTV is resident or, if the VTV is migrated, the VTV record contains the VTSS name from which the VTV was migrated.
- Once you set the VTSS name via the NAME parameter, you *cannot* change this identifier in the HSC CDS. That is, the CONFIG utility *will not* let you change the NAME parameter after an initial setting and changing the VTSS name using the Subsystem Name field of the LOP or VOP *cannot* change the VTSS name in the HSC CDS.
- It is especially critical that you *do not* attempt to rename a VTSS that contains data on VTVs, which includes VTSS-resident VTVs and migrated VTVs!
- For an initial setting *only* (not a change), you can set the VTSS name in the NAME parameter only if the VTSS name value in the VTSS microcode is:
  - The factory setting (all blanks).
  - A value of 99999999 (eight 9s).
- Therefore, for an initial setting *only,* if the name in the VTSS microcode *is not* all blanks or 99999999, your StorageTek hardware representative must use the VTSS LOP or VOP to set the VTSS name to 99999999 so you can set the VTSS name to the value you want via the NAME parameter.

Caution –

VTD Unit Addresses

Determine MVS unit addresses for your system's VTDs as follows:

- For each VTSS in your VSM configuration, determine a unique unit address range for the VTDs in that VTSS. Do not use duplicate addresses or overlapping address ranges, either within the VTDs in a VTSS or across VTSSs.
- For each VTSS in your VSM configuration, you must define its VTD unit addresses to VTCS via CONFIG.

  In a multi-host, multi-VTSS configuration, you can configure your VTD unit addresses to restrict host access to VTSSs. Note that the VTVs created and MVCs initially written to from a VTSS are considered that VTSS's resources, so only hosts with access to a VTSS also have access to its VTVs and MVCs. For more information, see Chapter 7, "Configuring VTCS".
- For each HSC host, use the HCD facility to define to MVS the VTDs that host can access as described in "Configuring MVS Device Numbers and Esoterics" on page 87. The unit addresses you specify via the HCD facility *must* match the unit address range you specified for that host via CONFIG.
- If you use MIM or GRS, add VTDs to the list of managed devices.

VSM Esoterics and Esoteric Substitution

The following sections tell how to use VSM esoterics and esoteric substitution to influence VTD allocation for the following methods for routing data sets to VSM:

- "General Guidelines and Requirements for VSM Esoterics" on page 14
- SMC TAPEREQ statements and POLICY commands; see "VSM Esoterics and Esoteric Substitution for SMC TAPEREQ Statements and POLICY Commands" on page 15
- The StorageTek DFSMS Interface; see "VSM Esoterics and Esoteric Substitution for the StorageTek DFSMS Interface" on page 15

General Guidelines and Requirements for VSM Esoterics

- To ensure that virtual requests actually go to virtual, StorageTek recommends that esoterics contain **only** VTDs.
- For any esoteric that you design, you must define the esoteric and associate it with the MVS device numbers for the VTDs you have chosen for that esoteric; see "Associating VTD MVS Device Numbers and Esoterics" on page 87.
- Assigning a Management Class to a VTV **requires** allocating a VTD in a VTSS that can satisfy the requirements of the assigned Management Class. For any jobs that route data to VSM via esoteric substitution **and** assign a Management Class to the data, **ensure** that you specify an esoteric that includes VTSSs that can satisfy the requirements of the Management Class!

VSM Esoterics and Esoteric Substitution for SMC TAPEREQ Statements and POLICY Commands

With the SMC TAPEREQ statements and POLICY commands, you can specify an esoteric on the SMC POLICY statement and specify the policy name on the TAPEREQ statement. StorageTek **strongly recommends** this method because esoterics on a TAPEREQ statement will be unsupported in a future release.

For example, for scratch VTV mounts, if the esoteric VIRTUAL represents all VTDs in your system and the esoteric VTSS1 represents all VTDs in VTSS1, the following POLICY statement allows SMC to allocate to VTSS1 providing that Management Class ACCOUNT is compatible with VTSS1. Otherwise, allocation will go to another VTD in esoteric VIRTUAL that is compatible with Management Class ACCOUNT. **Also note that** the IDAXVOLCNT value prevents prevent job failures when virtual volume count will exceed the MVS default 5:

```
POLICY NAME(VSMPOL1) MGMT(ACCOUNT) ESOT(VTSS1) -
IDASESOT(VIRTUAL) IDAXVOLCNT(20)
```

Next, you must set policy on with the SMC IDAX command:

```
IDAX POLICY(ON)
```

Finally, you select Policy VSMPOL1 with a TAPEREQ for data set with an HLQ of ACCOUNTS:

```
TAPEREQ DSN(ACCOUNTS.**) POLICY(VSMPOL1)
```

See *SMC Configuration and Administration Guide* for more information.

VSM Esoterics and Esoteric Substitution for the StorageTek DFSMS Interface

In the StorageTek DFSMS interface, a management class name can now reference an SMC policy name. StorageTek recommends converting your StorageTek DFSMS ACS routines to return a Management Class to reference an SMC Policy because this method does not require defining esoterics as Storage Groups and allows you to specify additional policy options. You must also specify the SMSDEF MGMTPOL(ON) or MGMTPOL(ALL) parameter. See *SMC Configuration and Administration Guide* for more information.

Note – The StorageTek DFSMS interface cannot assign Management Class to a VTV at VTD allocation. Management Class is assigned by the DFSMS interface at the time the mount is requested.

Designing VSM Esoterics for Esoteric Substitution

VSM Esoterics for JES2 and JES3 without Tape SETUP

In JES2 and JES3 without tape SETUP, for TAPEREQ statements or User Exits, esoteric definition and substitution is as follows:

- Esoteric definition is **optional** with these interfaces in these environments. That is, if you do not use esoteric substitution for VSM, you do not have to define any VSM esoterics.
- As long as the esoteric is a valid esoteric defined to MVS, esoteric substitution works as follows:
  - Allocation determines the common drives between the specified esoteric and the Eligible Device List (EDL).
  - As long as there are sufficient drives in the list of common drives, then allocation continues using this list of common drives.
- If you use esoteric substitution in a multi-VTSS environment, StorageTek recommends that you**:**
  - Define an esoteric for each VTSS.
  - Ensure that each VTSS esoteric represents exactly the entire range of devices for only that VTSS.
  - Ensure that the VTSS esoteric name matches the VTSS name defined via the CONFIG utility.

  This approach allows you to do esoteric substitution at the VTSS level and also allows you to use the same VTSS name defined via the CONFIG utility.

  - You can also define other virtual esoterics. For example, you can define and substitute an esoteric that represents all of the VTDs in all VTSSs or an esoteric that represents all VTSSs that comprise VTSS Clusters.
- For consistency, especially in installations that run both JES2 and JES3, you may want to define an esoteric structure for JES2 such as shown for JES3 with tape SETUP in FIGURE 1-1 on page 17.

VSM Esoterics for JES3 with Tape SETUP

FIGURE 1-1 is an example of a VSM esoteric structure for JES3 with tape SETUP.

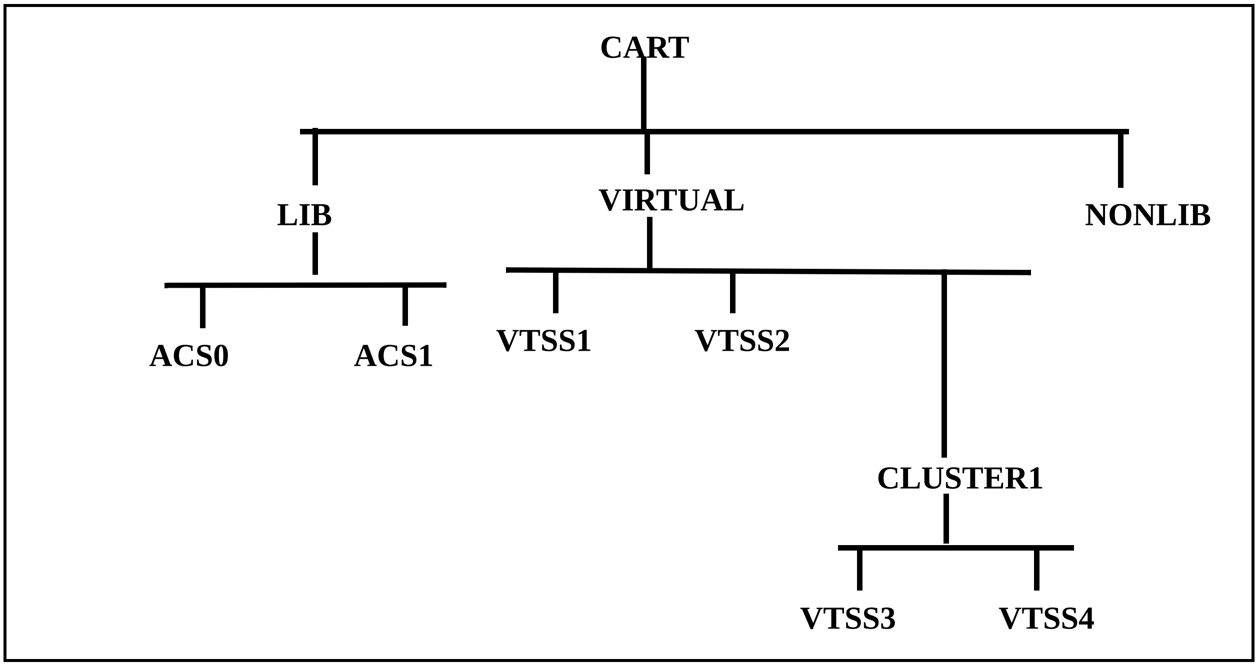


FIGURE 1-1 Example VSM Esoteric Structure for JES3 with Tape SETUP

Defining VSM Esoterics for JES3 with Tape SETUP

FIGURE 1-1 illustrates the VSM requirements and recommendations for defining esoterics in JES3 with tape SETUP as follows:

Requirement

Esoteric definition is **required** with these interfaces in this environment. That is, even if you do **not** use esoteric substitution for VSM, you must define VSM esoterics as described in the following sections.

Recommendation

StorageTek recommends that you begin the esoteric structure with a high-level esoteric that allocates all tape jobs. Your system typically has this esoteric previously defined, which is **CART** in FIGURE 1-1.

Requirement

For Nearline systems, under the high-level esoteric, you must have defined:

- A subesoteric for “library” (Nearline managed) real tape jobs (**LIB** in FIGURE 1-1). Under **LIB** are the subesoterics **ACSO** and **ACS1** that represent drives in two different ACSs. You specify **ACSO** and **ACS1** in the HSC LIBGEN on the ACSDRV parameter of the SLIACS macro.
- A second subesoteric for “non-library” (not Nearline managed) tape jobs (**NONLIB** in FIGURE 1-1).

Requirement

Any esoteric that represents VTDs **must** contain only VTDs and must be defined in the JES3 initialization deck.

Requirement

For multi-VTSS environments, you **must** define an esoteric that represents all VTDs in your system (**VIRTUAL** in FIGURE 1-1 on page 17).

Requirement

StorageTek recommends that you place the esoteric that represents all VTDs in your system (**VIRTUAL** in FIGURE 1-1 on page 17) under the high-level esoteric. The benefit of placing the virtual subesoteric under the high-level esoteric is that this esoteric structure prevents JES3 HWS device allocation optimization from allocating a VTV to a non-virtual device.

Recommendation

For multi-VTSS environments, StorageTek recommends that you define an esoteric for each VTSS **(VTSS1**, **VTSS2**, **VTSS3**, and **VTSS4** in FIGURE 1-1 on page 17). Each esoteric *must* represent the entire range of devices for only that VTSS and the esoteric name *must* match the VTSS name on the VTCS CONFIG statement.

Note – In JES3 with tape SETUP, you can also define other esoterics, such as an esoteric that represents all VTDs in all VTSSs or all VTDs in all VTSSs that comprise a VTSS Cluster (for example, **CLUSTER1** in FIGURE 1-1 on page 17). You can use these esoterics for esoteric substitution and JCL reference.

VTV Definitions

You define your system's VTV volsers to VTCS by volser ranges with the following:

- VTCS CONFIG. as described in Chapter 7, "Configuring VTCS".
- HSC VOLATTR statement as described in Chapter 6, "Reconfiguring HSC".

Determine your system's VTV volser ranges as follows:

- For an initial CONFIG definition, consider defining only enough VTVs for reasonable growth. This method allows for growth without rerunning CONFIG but does not reserve unnecessary space in the CDS, which can impact VTV processing performance. Note that if the CDS does not contain sufficient space to run VTCS CONFIG, you will also have to run HSC RECONFIG. For more information about sizing the CDS for VSM, see "HSC CDS DASD Space" on page 49.

  If your VTV requirements expand beyond initial definition, then rerun CONFIG to define additional VTVs.

- You can only add new VTV ranges. A range can consist of a single volume. You cannot delete or modify existing ranges.
- If you are currently writing files to non–standard length tapes and will route these files to VTVs, you may need additional VTVs because VTVs emulate standard–length cartridges. This may require a change to the JCL volume count parameter.
- NCS/VTCS **does not allow** allocation of unlabeled tapes to VTVs. Unlabeled VTVs can cause the following for scratch VTV allocation requests:
  - If your JCL specifies a virtual esoteric, the NCS Storage Management Component (SMC) fails the allocation.
  - If you have a default esoteric such as CART and specify allocation to virtual (via SMC TAPEREQ or HSC User Exit), the allocation will go to a non-virtual device.
- You must define VTV volsers to your tape management system; for more information, see Chapter 9, "Completing the VSM Configuration".
- Ensure that VTV volser ranges do not duplicate or overlap existing TMS ranges or volsers of real tape volumes, including Nearline volumes, *including* MVCs and Nearline volumes that are regularly entered and ejected from the ACS!
- If VTDs are being used across multiple MVS images and VTV volsers are unique, add a generic entry for SYSZVOLS to the SYSTEM inclusion RNL to insure that a VTV is used by only one job at a time. If you are using automatic tape switching, also add a generic entry for SYSZVOLS to the SYSTEM inclusion RNL to prevent a system from holding a tape device while it waits for a mount of a volume that is being used by another system.

  For more information, see the IBM publication *OS/390 MVS Planning: Global Resource Serialization.*

- If you specify scratch subpools for scratch mounts of VTVs (for example, with the SMC TAPEREQ SUBPOOL parameter or SMC User Exit 01), use the following guidelines:
  - If you need to define new subpools, add SCRPOOL statements to HSC PARMLIB for the VTV volsers.

- HSC mixed-media support lets you mix VTV and real volume types in the same scratch pool. In this case, ensure that the mount request specifies a VTD as transport type (for example, via TAPEREQ MEDIA (VIRTUAL). In addition, if you are routing data to a specific VTSS (for example, by using esoteric substitution as described in "VSM Esoterics and Esoteric Substitution" on page 14) and the request specifies a subpool, ensure that the subpool contains scratch VTVs.

Tip – Note the following:

- You can use Query to display the available scratch count of a subpool.
- You can dynamically reload SCRPOOL statements via the SCRPDEF command. For more information, see *HSC System Programmer's Guide for MVS.*
- The Warn SCRatch, Display SCRatch, and Display THReshld commands are enhanced to let you manage and monitor scratch VTVs. For more information, see Chapter 2, "Commands, Control Statements, and Utilities," in *HSC Operator's Guide for MVS.*

- By default, VTCS assigns a Management Class to VTVs only on scratch mounts. You can, however, specify that VTCS assigns a Management Class whenever VTCS mounts a VTV (for read or write).

Caution – If you specify that VTCS assigns a Management Class whenever VTCS mounts a VTV, these attributes can change, which can cause undesirable or unpredictable results. For example, if an application writes data set PROD.DATA to VTV100 with a Management Class of PROD, then writes data set TEST.DATA to VTV100 with a Management Class of TEST, then the VTV (and both data sets) has a Management Class of TEST. Similarly, it is possible to write SMC TAPEREQ statements or SMS routines that assign different Management Classes to the same data set (for example, based on jobname), which can also cause a VTV's Management Class to change.

RTD Definitions

RTDs, which are Nearline transports, require LIBGEN definitions.

If your system's RTDs are new transports, determine 4-digit hexadecimal MVS unit addresses for these transports. The addresses you choose must be the same for all hosts in the configuration. You will use these addresses to:

- Add a SLIDRIVS macro to define RTD device addresses during the HSC LIBGEN update as described in Chapter 6, "Reconfiguring HSC".
- Run the HCD facility to assign MVS device numbers to these transports as described in "Configuring MVS Device Numbers and Esoterics" on page 87.

Caution – Note the following:

- StorageTek **strongly recommends** that you define your RTDs to MVS (as normal 3490 tape drives), even if you do not intend to vary them online to MVS. This prevents the RTD addresses used in CONFIG and LIBGEN from accidentally being used for other devices. If you do not do this, and subsequently use the addresses for other MVS devices, you will cause problems with LOGREC processing, because VTCS will write records using the RTD addresses, and MVS will write records for other devices with those same addresses.

  If you define your RTDs to MVS, however, **do not** define them to NCS using `UNITATTR` with `MODEL(IGNORE)`, which causes SMC to **not** send the RTD mount request to VTCS.
- StorageTek **requires** that RMM users define their RTDs to MVS. RMM causes problems if it sees the IEC501 mount message generated for RTDs by VTCS and if the device in the mount message is not defined to MVS.

- Specify the MVS device numbers on the CONFIG VTSS RTD DEVNO parameter as described in Step 6 on page 96, and you do this whether your system's RTDs are new or existing transports.

  You also specify the RTD identifier on the CONFIG VTSS RTD NAME parameter. To help identify the RTDs connected to each VTSS, StorageTek recommends that you choose RTD identifiers that reflect the VTSS name (specified on the VTSS NAME parameter) and the RTD's MVS device number (specified on the RTD DEVNO parameter).

  In configurations where multiple VTSSs are connected to and dynamically share the same RTD, in each VTSS definition you can either assign unique RTD identifiers or use the same RTD identifier.

Note –

- You can specify that Nearline transports can only be used as RTDs. For more information, see "Creating or Updating the HSC LIBGEN" on page 112.
- Ensure that you use the drive operator's panel or T10000 Virtual Operator Panel (VOP to enable the SL PROT (Standard Label Protect) function on the RTDs (ESCON or FICON).

MVC Definitions

You define MVCs as described in the following sections:

- "Define and Select Nearline Volumes" on page 22
- "Define Available MVCs with CONFIG" on page 23
- "Define the MVC Pool" on page 23
- "Protect MVCs and Nearline Volumes" on page 24
- "VTCS Considerations to Correctly Specify MVC Media" on page 25

Define and Select Nearline Volumes

First, to define and select Nearline volumes for MVCs, use these guidelines:

- MVCs require VOLATTR statements to ensure that VTCS will select the correct RTD device type for each MVC. Select volumes for MVCs that are compatible with your system's RTD transport types.
- For mixed–media VSM systems, select volumes that include at least one media type compatible with each of your system's RTD transport types. See TABLE 1-4 on page 7 for information about the RTD transport types and media that VSM supports.

  Note that VSM selects media for migration processing and reclaim processing according to the media types of volumes in your system's MVC pool.
- If you define new Nearline volumes as MVCs, you must create MVS volsers for these volumes and initialize STANDARD, ECART, and ZCART volumes as 36-track format standard label volumes.
- As described in "Protect MVCs and Nearline Volumes" on page 24, if possible, create a new and separate volser range for MVCs. Ensure that if you define new volumes, you do not overlap existing TMS ranges.

Define Available MVCs with CONFIG

Second, use VTCS CONFIG to define all MVCs *available* to VTCS. CONFIG reserves space for these volumes in the HSC CDS. The MVCPool statements define the *MVC pool*, which contains the MVCs that VTCS actually *uses*.

For an initial CONFIG definition, consider defining only enough MVCs for reasonable growth of your MVC pool. This method allows you to expand your MVC pool without rerunning CONFIG (you only have to change your MVCPool statements) but does not reserve unnecessary space in the CDS, which can impact MVC processing performance. Note that if the CDS does not contain sufficient space to run VTCS CONFIG, you will also have to run HSC RECONFIG. For more information about sizing the CDS for VSM, see "HSC CDS DASD Space" on page 49.

For example, if you currently need 300 MVCs but will need to add 150 more MVCs within the next 6 months, define an MVC range of 450 volsers with CONFIG, but only apply MVCPool statements to the first 300 "in use" MVCs. As your MVC space requirements increase, update and reapply your MVCPool statements to add the second 150 MVCs.

If your MVC space requirements expand beyond the second 150 MVCs, then rerun CONFIG to define additional MVC ranges and update and reapply your MVCPool statements.

Note –

- You can only add new MVC ranges. A range can consist of a single volume. You cannot delete or modify existing ranges.
- A VSM audit of all MVCs will audit all MVCs defined with CONFIG including those that are *not* specified in the MVCPool statements.

Define the MVC Pool

Third, create MVCPool statements, which specify the pool of MVCs available for migration and consolidation requests, using the following guidelines:

- Because MVCPool statements specify the "in use" MVCs, MVCPool statements can (and typically do) define a subset of the available MVCs you defined via CONFIG. MVCPool statements, however, can only specify MVCs you already defined with CONFIG. For more information about defining an initial MVC pool, see "Defining MVCs to HSC" on page 115.
- StorageTek recommends that you use identical MVCPool statements on all hosts. A host can automigrate any VTV on any VTSS to which the host is connected, including VTVs created by another host. If your VSM configuration consists of hosts cross-connected to multiple VTSSs, therefore, separate MVC pools do not guarantee that a host automigrates only VTVs it creates to only its MVC pool. To most effectively segregate VTVs on groups of MVCs, see *Beyond the Basics: VTCS Leading Edge Techniques*.
- Ensure that your MVC pool consists of volumes that physically reside in ACS that contains your system's RTDs.
- To redefine your MVC pool, change your MVCPool statements and reload them via the VT MVCDEF command.

Protect MVCs and Nearline Volumes

Fourth, protect MVCs and Nearline volumes that are *not* MVCs from accidental overwrites as follows:

- If possible, create a new and separate volser range for MVCs to prevent HSC from writing to MVCs and to prevent VSM from writing to conventional Nearline volumes.
- VTCS, not MVS, controls access to MVCs. The tape management system does not control VSM access to an MVC volume and does not record its usage. If you choose to define MVCs to the tape management system, to ensure that the tape management system does not accidentally access MVCs, follow the guidelines in "Updating the Tape Management System" on page 156.
- Use your security system to restrict access to MVCs as described in "Defining VSM Security" on page 157.
- HSC automatically marks newly entered MVC volumes as non-scratch. If you define existing Nearline volumes as MVCs, ensure that these volumes do not contain data you need, then run the HSC UNSCratch Utility to unscratch them. For more information, see *HSC System Programmer's Guide for MVS*.

VTCS Considerations to Correctly Specify MVC Media

TABLE 1-8 describes the values required to specify the desired media and recording technique on the HSC VOLATTR statement and HSC STORCLAS statement to correctly specify the desired MVC media.

TABLE 1-8 RTD Model/MVC Media Values

Transport Model	TAPEREQ/ VOLATTR MEDIA	RECTECH	STORCLAS MEDIA	Cartridge Type - Specified by STORCLAS MEDIA	Density	Encrypted?
4490	STANDARD	STANDARD	STANDARD	standard length 3480 cartridge	single	N/A
9490, 9490EE	ECART	ECART	ECART	3490E cartridge	single	N/A
9490EE	ZCART	ZCART	ZCART	3490EE cartridge	single	N/A
9840	STK1R	STK1RA	STK1RAB	T9840A or T9840B cartridge	single	N/A
T9840B		STK1RB	STK1RAB	T9840A or T9840B cartridge	single	N/A
T9840C		STK1RC	STK1RC	T9840C cartridge	double	N/A
T9840D - T9840D Non-Encrypting Transport		STK1RD	STK1RD	T9840D cartridge	triple	no
T9840DE - T9840D Encrypting Transport		STK1RDE	STK1RDE	T9840D cartridge for encryption	triple	yes
T9940A	STK2P	STK2PA	STK2PA	T9940A cartridge	single	N/A
T9940B		STK2PB	STK2PB	T9940B cartridge	double	N/A

TABLE 1-8 RTD Model/MVC Media Values

Transport Model	TAPEREQ/ VOLATTR MEDIA	RECTECH	STORCLAS MEDIA	Cartridge Type - Specified by STORCLAS MEDIA	Density	Encrypted?
T1A34 - T10000A Non-Encrypting Transport	T10000T1	T1A34	T1A000T1	T10000 full capacity cartridge	single	no
T1AE34 - T10000A Encrypting Transport		T1AE34	T1A000E1	T10000 full capacity cartridge for encryption	single	yes
T1A34 - T10000A Non-Encrypting Transport	T10000TS	T1A34	T1A000TS	T10000 sport cartridge	single	no
T1AE34 - T10000A Encrypting Transport		T1AE34	T1A000ES	T10000 sport cartridge for encryption	single	yes
T1B34 - T10000B Non-Encrypting Transport	T10000T1	T1B34	T1B000T1	T10000 full capacity cartridge	double	no
T1BE34 - T10000B Encrypting Transport		T1BE34	T1B000E1	T10000 full capacity cartridge for encryption	double	yes
T1B34 - T10000B Non-Encrypting Transport	T10000TS	T1B34	T1B000TS	T10000 sport cartridge	double	no
T1BE34 - T10000B Encrypting Transport		T1BE34	T1B000ES	T10000 sport cartridge for encryption	double	yes

TABLE 1-8 RTD Model/MVC Media Values

Transport Model	TAPEREQ/ VOLATTR MEDIA	RECTECH	STORCLAS MEDIA	Cartridge Type - Specified by STORCLAS MEDIA	Density	Encrypted?
T1C34 - T10000C Non-Encrypting Transport	T10000T2	T1C34	T1C000T2	T10000C full capacity cartridge	triple	no
T1CE34 - T10000C Encrypting Transport		T1CE34	T1C000E2	T10000C full capacity cartridge for encryption	triple	yes
T1C34 - T10000C Non-Encrypting Transport	T10000TT	T1C34	T1C000TT	T10000C sport cartridge	triple	no
T1CE34 - T10000C Encrypting Transport		T1CE34	T1C000ET	T10000C sport cartridge for encryption	triple	yes

Use TABLE 1-8 as a guideline to:

- Create VOLATTR statements that segregate single/double density media or encrypted/non-encrypted media.
- Specify the correct STORCLAS MEDIA values to assign the desired cartridge type and recording technique to MVCs.
- Determine which transport models can write to/read from which media. A higher capability transport (double density vs. single, or encryption vs. non-encryption) can read from media written by a lower capability transport, but can only write to that media from the beginning of the tape. A lower capability transport, however, cannot read from media written by a higher capability transport but can write to that media from the beginning of the tape.

Examples

- If you are adding T1000C encrypting transports and new media to encrypt, create VOLATTRs for the new media and STORCLAS statements to allow VTCS to select this media. For example:

```
VOLATTR VOLSER(MVC500-MVC599) MEDIA(TI0000T2) RECTECH(TICE34)
STORCLAS NAME(CENCRYPT) MEDIA(T1C000E2)
```

- If you are adding T10000C encrypting transports and want to convert existing media to encryption media, change existing VOLATTRs to specify encryption and change existing STORCLAS statements to request encryption. For example:

```
VOLATTR VOLSER(MVC200-MVC299) MEDIA(TI0000T2) RECTECH(TICE34)
STORCLAS NAME(T10K) MEDIA(T1C000E2)
```

  Here's how it works: If I have MVCs that already contain data, I cannot add "encrypted" VTVs to these MVCs. I can, however, encrypt data on initialized MVCs that do not contain data. To make this strategy work, therefore, ensure that you have sufficient free T10000 MVCs and also consider doing demand drains on MVCs that do contain data to free them up.

Using the STORclas MEDIA Parameter for MVC Media Preferencing

By default, in mixed-media VSM systems, VTV automatic and demand migrations (and consolidations) attempt to go to MVCs by media type in this order:

1. Standard length 3480 cartridge
2. 3490E cartridge
3. 3490EE cartridge
4. T9840A/B cartridge
5. T9840C cartridge
6. T9940A cartridge
7. T9840D cartridge
8. T10000A/B sport cartridge
9. T9940B cartridge
10. T10000A/B full capacity cartridge
11. T10000C sport cartridge
12. T10000C full capacity cartridge

By default, for automatic and demand space reclamations, VSM attempts to write VTVs to output MVCs by media type in this order:

1. T10000C full capacity cartridge
2. T10000C sport cartridge
3. T10000A/B full capacity cartridge
4. T9940B cartridge
5. T10000A/B sport cartridge
6. T9840D cartridge
7. T9940A cartridge
8. T9840C cartridge
9. T9840A/B cartridge
10. 3490EE cartridge
11. 3490E cartridge
12. Standard length 3480 cartridge

The MEDIA parameter of the STORclas statement specifies a preference list of MVC media types. This list supersedes the default media selection list. **Note that** for reclamation, VTCS attempts to write VTVs back to MVCs in the **reverse** of the order specified on the MEDIA parameter.

For example, if you specify the following on the MEDIA parameter of the STORclas statement...

```
MEDIA(STK1RAB,STK1RC,STK2PB)
```

- ...to select an MVC for migration to this Storage Class, VTCS searches for a usable MVC in the order STK1RAB, STK1RC, STK2PB.
- ...to select an MVC for the output of reclaim to this Storage Class, VTCS searches for a usable MVC in the order STK2PB, STK1RC, STK1RAB.

You can specify the media and ACS preferencing via the Storage Class(es) specified on the MIGpol parameter of the MGMTclas control statement.

To optimize recall processing in mixed-media systems, ensure that your MVC pool has at least one media type compatible with each RTD type.

CDS VTCS Level

You can determine what level CDS you currently have with the HSC D CDS command as shown in the example in FIGURE 1-2.

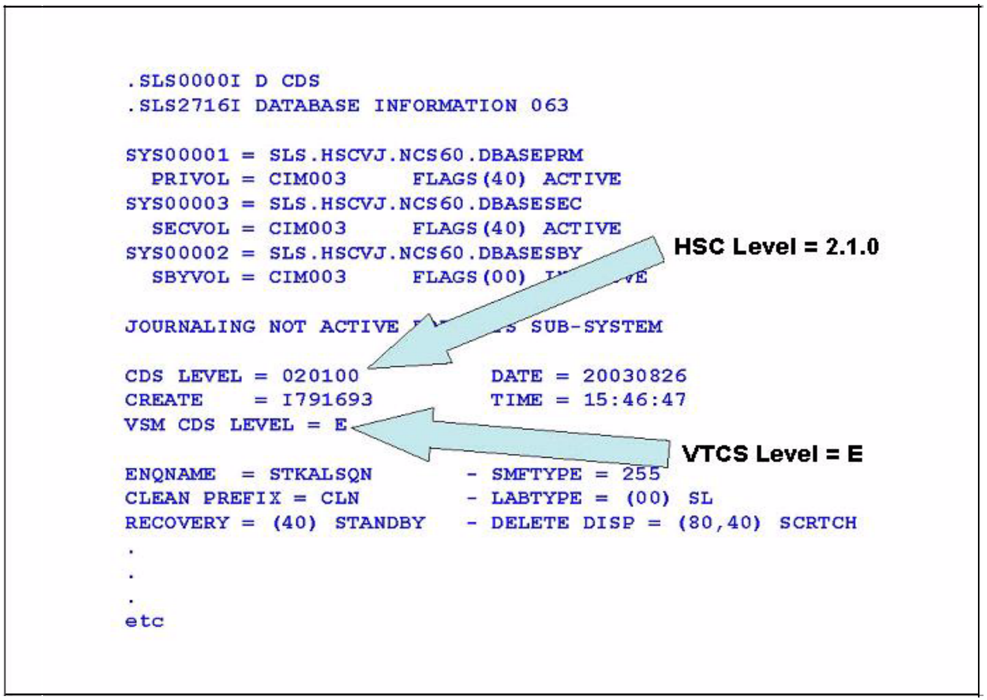


FIGURE 1-2 Example Output from the HSC D CDS Command

The HSC CDS manages HSC and VTCS. There is only one CDS, but internally it has two elements – the HSC portion and the VTCS portion. In FIGURE 1-2, note that a CDS consists of an **HSC Level** (2.1.0 in this example) and a **VTCS Level** (E, in this example).

As described in TABLE 1-9 on page 33, each supported VTCS version supports **only** a subset of these VTCS Levels. If you are, therefore, running with a mixed set of VTCS versions against a CDS it is important to ensure that the CDS is set at a Level that is supported by **all** the versions being run. **Also note that** certain VTCS functions are **only** available by running with the CDS at a certain Level.

TABLE 1-9 CDS Levels for Supported VTCS Versions

This VTCS CDS Level...	...is valid for these VTCS/NCS versions...	...and this VTSS hardware...	...and provides these enhancements
E	6.0, 6.1, 6.2	VSM2 and VSM3 VSM4 with up to 256 VTDs per VTSS and/or up to 16 RTDs per VTSS. RTD sharing **except** for paired RTDs (a paired RTD shares a CIP with another Nearlink connection, either an RTD or a CLINK).	▪ 4 MVC copies ▪ 800 Mb VTVs
F	6.1, 6.2		▪ Near Continuous Operations (NCO) ▪ Bi-directional clustering ▪ Improved CDS I/O performances - reduces the I/O required to manage virtual scratch sub-pools
G	6.2		▪ 400Mb/800Mb/2Gb/4Gb VTVs ▪ Standard/Large VTV Pages ▪ 65000 VTVs per MVC

Guidelines for Changing CDS VTCS Levels

Note the following:

- TABLE 1-10 describes the supported CDS levels for VTCS 6.2 and the corresponding CONFIG CDSLEVEL values.

TABLE 1-10 Valid CONFIG CDSLEVEL Values for VTCS 6.2

CDS VTCS Level	CDSLEVEL Value
E	V6ABOVE
F	V61ABOVE
G	V62ABOVE

- E, F, and G formats are considered "VSM Extended Format CDSs." The VSM Extended Format CDS **is required for VTCS 6.2. Also note that** after you convert the CDS to VSM Extended Format, you **cannot** run VTCS 4.0 or lower against the converted CDS. VTCS 4.0 and below **is incompatible with** and **will not initialize with** the VSM Extended Format CDS. If you are a **new** VTCS 6.2 customer, VSM Extended Format is the default, so no conversion is required.
- "Building a Simple CONFIG Deck" on page 119 tells how to specify the CDS Level. **Note that** VTCS **will not start** with an unsupported CDS Level. In addition, the 'D' Level CDS does not support VSM5s and cannot be created with VTCS 7.0. D level, therefore, is supported but not recommended.

Note –

- Regressing from an E, F, or G Level CDS can cause unpredictable results if the 4 VTV copy feature has been used. Any copies above the maximum of two allowed on a 'B' to 'D' Level CDS will be dropped!
- To regress from a G Level CDS:
  - No MVCs can contain greater than 32000 migrated VTVs. This is not the same as the count reported on the MVC displays and reports. Any MVCs greater than 32000 migrated VTVs must then be drained.
  - Any VTVs written in large page format and/or any VTVs written in either 2 Gb or 4Gb sizes must be deleted, which can be done by changing the VTV's Management Class to a Management Class with DELSCR(YES) and then scratching the VTV.

  Therefore, you may want to install E, F, G in a test system only, or run it in production without using an E, F, G feature until you are sure that the environment is stable.
- The upgrade to a G is non-disruptive (can be done without bringing HSC/VTCS down), but all hosts **must** be at 6.2 down), but all hosts **must** be at 6.2 or above.
- Upgrades to/downgrades from G level typically take less time than upgrades to/downgrades from previous levels.

Storing VTCS Locks in a Coupling Facility (Optional)

Before VTCS updates a CDS record (for example, a VTV record) it locks the record to avoid contention from concurrent updates from multiple hosts. VTCS releases the CDS lock record once the CDS record has been updated.

Customers with large VTCS configurations experience high CDS I/O rates, part of which is due to the need to access CDS lock records. As configurations grow in size, (for example, by adding more hosts), CDS performance becomes a bottleneck.

An MVS Coupling Facility is a suitable alternative medium for VTCS Lock data because:

- A Coupling Facility provides very fast data transfer speeds, so the new I/O to the Coupling Facility with Lock records is less than the corresponding I/O to a CDS with Lock records.
- MVS provides a technique that allows data stored in a failing Coupling Facility to be re-built in another Coupling Facility (if one exists) without terminating the application.

Note – If VTCS locks are held in a coupling facility structure, VTCS uses the Structure (rather than the HSC mechanism) for sending/receiving Host-to-Host messages.

When to implement VTCS locks in a Coupling Facility Structure

Storing VTCS lock data in a Coupling Facility Structure is a solution to the **specific** problem of VTCS causing high I/O demand to the CDS in some configurations.

Note – Using a Coupling Facility Structure is **not** a solution to all CDS performance issues. For this reason, StorageTek recommends that if you believe you have a CDS performance problem, contact StorageTek Software support to have the problem analyzed **before** considering implementing VTCS lock data in a Coupling Facility.

Requirements

To store VTCS Locks in a Coupling Facility:

- All hosts must have access to the same Coupling Facility. Similarly, if you have an alternate Coupling Facility to rebuild the VTCS Lock Structure, all hosts must have access to that alternate Coupling Facility. All hosts must also be in a Sysplex.

  The Coupling Facility implementation currently assumes all hosts run HSC/VTCS. VTCS will not start successfully on a host running HSC/VTCS if a host running HSC only is already active.

- The Coupling Facility Structure must be predefined to MVS before VTCS can use it to store CDS Lock Records. VTCS uses the list form of a Coupling Facility Structure. Display LOCKS shows one of the following VTCS Coupling Facility lock types:

  Host Footprint

  : used to serialize access to the host footprint list.

  Host to Host

  : used to serialize access to a given host to host list.

  Lock data

  : used to serialize access to the VTCS lock data.

  Formatting

  : used to serialize the initial formatting of the structure; also used when rebuilding data.

  System

  : lock is held, but is not a lock used by VTCS; assume it is used by MVS.

Sizing the Coupling Facility Structure

A Structure size of 768K should be sufficient for configurations up to 100 VTSSs.

If the Structure is sized too small, HSC/VTCS will be unable to connect to the Structure or will be able to connect but will be unable to format all of its data. In both cases, VTCS will terminate.

Defining the Coupling Facility Structure to MVS

FIGURE 1-3 on page 37 shows an example of an IXCMIAPU job to define a VTCS Lock Structure within a Coupling Facility Resource Manager (CRFM). In this example, note that:

- There are two Coupling Facilities, FACIL01 and FACIL02.
- There is a 768K Structure called STK_VTCS_LOCKS to store VTCS Lock Records.
- Structure STK_VTCS_LOCKS can exist in either Coupling Facility, but FACIL01 is preferred over FACIL02. If VTCS starts to store lock data in FACIL01 and FACIL01 then becomes unavailable, VTCS attempts to build the STK_VTCS_LOCKS in FACIL02 to ensure continuous operations.

Note – If you define only one Coupling Facility, and it becomes unavailable, VTCS terminates on all hosts but HSC will still be running. If this occurs, do one of the following:

- Fix the Coupling Facility error, then recycle HSC/VTCS on all hosts; you can resume without changing the configuration.

- Use the procedure in Reverting to Storing VTCS Lock Data in the CDS.

Reverting to Storing VTCS Lock Data in the CDS

To revert to storing VTCS Lock Data in the CDS:

1. **Stop HSC on all hosts.**
2. **Run CONFIG RESET *without* the LOCKSTR parameter so VTCS can store VTCS Locks in the CDS.**
3. **Restart HSC/VTCS on all hosts.**

   For more information, see *VTCS Command and Utility Reference*.

```
//SYSPRINT DD SYSOUT=*
//SYSIN        DD *
  DATA TYPE(CFRM) REPORT(YES)
  DEFINE POLICY NAME(POLICY1) REPLACE(YES)
           CF   NAME(FACIL01)
                  TYPE(123456)
                  MFG(IBM)
                  PLANT(02)
                  SEQUENCE(123456789012)
                  PARTITION(1)
                  CPCID(00)
                  SIDE(0)
                  DUMPSPACE(2000)
          CF   NAME(FACIL02)
                  TYPE(123456)
                  MFG(IBM)
                  PLANT(02)
                  SEQUENCE(123456789012)
                  PARTITION(2)
                  CPCID(00)
                  SIDE(1)
                  DUMPSPACE(2000)
        STRUCTURE  NAME(STK_VTCS_LOCKS)
                 SIZE(768)
                    PREFLIST(FACIL01,FACIL02)
```

FIGURE 1-3 Example IXCMIAPU Job to Define a Coupling Facility Structure

Managing Failures/Unavailability of the VTCS lock structure

VTCS supports Structure Rebuild to allow for failures/unavailability of the Structure or the Coupling Facility containing the Structure.

Structure rebuild can be initiated by:

- Operator command (SETXCF START,REBUILD,*xxx*) for a planned outage of the Structure or the Coupling Facility, and/or
- MVS or VTCS detecting an error in, or failure of, the Structure or the Coupling Facility.

Note that VTCS does **not** support System Managed Duplexing.

If the Structure used by VTCS can only be allocated in one Coupling Facility, VTCS will terminate on all Hosts if the Structure (or the Coupling Facility containing the Structure) fails or becomes unavailable.

If the Structure can be allocated in more than one Coupling Facility, VTCS's Structure Rebuild code will attempt to rebuild the data in an alternate Coupling Facility Structure. VTCS will only terminate if the rebuild fails.

Virtual ACS IDs

To connect to VSM, clients use a decimal virtual ACS ID that maps to a VTSS name that you determined in "VTSS Names" on page 12. You use the LibraryStation VIRTACS statement to do this mapping. To avoid conflicts with real ACS IDs, StorageTek recommends that you select virtual ACS IDs by starting with the highest possible value (126) and working backwards.

For example, to define virtual ACS 126 and map it to VTSS VTSS02, you would create the following VIRTACS statement:

```
VIRTACS ID(126) VTSSNAME(VTSS02)
```

Virtual ACS Location for VSM2s and VSM3s

To LibraryStation, a VSM2/3 virtual ACS consists of four virtual LSMs (0-3), each containing four drive panels (1-4) with four VTDs each (0-3) for a total of 64 VTDs. LibraryStation, therefore, references VTDs by their ACS location in *ascid,lsmid,panelnum,devicenum* format. For example, LibraryStation references the 27th VTD in virtual ACS 126, above, by ACS location 126,1,3,2.

TABLE 1-11 cross-references VTD numbers to their LibraryStation virtual ACS locations. See "Connecting MVS/CSC Clients to VSM" on page 148 for procedures for defining these library locations to MVS/CSC clients and mapping them to MVS device numbers.

TABLE 1-11 LibraryStation VSM2/3 Virtual ACS Locations for VTDs

VTD	Virtual ACS Location
1	*acsid*,0,1,0
2	*acsid*,0,1,1
3	*acsid*,0,1,2
4	*acsid*,0,1,3
5	*acsid*,0,2,0
6	*acsid*,0,2,1
7	*acsid*,0,2,2
8	*acsid*,0,2,3
9	*acsid*,0,3,0
10	*acsid*,0,3,1
11	*acsid*,0,3,2
12	*acsid*,0,3,3
13	*acsid*,0,4,0
14	*acsid*,0,4,1
15	*acsid*,0,4,2
16	*acsid*,0,4,3
17	*acsid*,1,1,0
18	*acsid*,1,1,1
19	*acsid*,1,1,2
20	*acsid*,1,1,3

TABLE 1-11 LibraryStation VSM2/3 Virtual ACS Locations for VTDs

21	*acsid*,1,2,0
22	*acsid*,1,2,1
23	*acsid*,1,2,2
24	*acsid*,1,2,3
25	*acsid*,1,3,0
26	*acsid*,1,3,1
27	*acsid*,1,3,2
28	*acsid*,1,3,3
29	*acsid*,1,4,0
30	*acsid*,1,4,1
31	*acsid*,1,4,2
32	*acsid*,1,4,3
33	*acsid*,2,1,0
34	*acsid*,2,1,1
35	*acsid*,2,1,2
36	*acsid*,2,1,3
37	*acsid*,2,2,0
38	*acsid*,2,2,1
39	*acsid*,2,2,2
40	*acsid*,2,2,3
41	*acsid*,2,3,0
42	*acsid*,2,3,1
43	*acsid*,2,3,2
44	*acsid*,2,3,3
45	*acsid*,2,4,0
46	*acsid*,2,4,1
47	*acsid*,2,4,2
48	*acsid*,2,4,3
49	*acsid*,3,1,0
50	*acsid*,3,1,1
51	*acsid*,3,1,2
52	*acsid*,3,1,3
53	*acsid*,3,2,0
54	*acsid*,3,2,1
55	*acsid*,3,2,2
56	*acsid*,3,2,3
57	*acsid*,3,3,0
58	*acsid*,3,3,1

TABLE 1-11 LibraryStation VSM2/3 Virtual ACS Locations for VTDs

59	*acsid*,3,3,2
60	*acsid*,3,3,3
61	*acsid*,3,4,0
62	*acsid*,3,4,1
63	*acsid*,3,4,2
64	*acsid*,3,4,3

Virtual ACS Location for VSM4s

To LibraryStation, a VSM4 virtual ACS consists of sixteen virtual LSMs (0-15), each containing four drive panels (1-4) with four VTDs each (0-3) for a total of 256 VTDs. LibraryStation, therefore, references VTDs by their ACS location in *ascid,lsmid,panelnum,devicenum* format. For example, LibraryStation references the 27th VTD in virtual ACS 126, above, by ACS location 126,1,3,2.

TABLE 1-12 cross-references VTD numbers to their LibraryStation virtual ACS locations. See "Connecting MVS/CSC Clients to VSM" on page 148 for procedures for defining these library locations to MVS/CSC clients and mapping them to MVS device numbers.

TABLE 1-12 LibraryStation VSM4 Virtual ACS Locations for VTDs

VTD	Virtual ACS Location
1	*acsid*,0,1,0
2	*acsid*,0,1,1
3	*acsid*,0,1,2
4	*acsid*,0,1,3
5	*acsid*,0,2,0
6	*acsid*,0,2,1
7	*acsid*,0,2,2
8	*acsid*,0,2,3
9	*acsid*,0,3,0
10	*acsid*,0,3,1
11	*acsid*,0,3,2
12	*acsid*,0,3,3
13	*acsid*,0,4,0
14	*acsid*,0,4,1
15	*acsid*,0,4,2
16	*acsid*,0,4,3
17	*acsid*,1,1,0
18	*acsid*,1,1,1
19	*acsid*,1,1,2
20	*acsid*,1,1,3
21	*acsid*,1,2,0
22	*acsid*,1,2,1
23	*acsid*,1,2,2
24	*acsid*,1,2,3
25	*acsid*,1,3,0
26	*acsid*,1,3,1
27	*acsid*,1,3,2
28	*acsid*,1,3,3
29	*acsid*,1,4,0

TABLE 1-12 LibraryStation VSM4 Virtual ACS Locations for VTDs

30	*acsid*,1,4,1
31	*acsid*,1,4,2
32	*acsid*,1,4,3
33	*acsid*,2,1,0
34	*acsid*,2,1,1
35	*acsid*,2,1,2
36	*acsid*,2,1,3
37	*acsid*,2,2,0
38	*acsid*,2,2,1
39	*acsid*,2,2,2
40	*acsid*,2,2,3
41	*acsid*,2,3,0
42	*acsid*,2,3,1
43	*acsid*,2,3,2
44	*acsid*,2,3,3
45	*acsid*,2,4,0
46	*acsid*,2,4,1
47	*acsid*,2,4,2
48	*acsid*,2,4,3
49	*acsid*,3,1,0
50	*acsid*,3,1,1
51	*acsid*,3,1,2
52	*acsid*,3,1,3
53	*acsid*,3,2,0
54	*acsid*,3,2,1
55	*acsid*,3,2,2
56	*acsid*,3,2,3
57	*acsid*,3,3,0
58	*acsid*,3,3,1
59	*acsid*,3,3,2
60	*acsid*,3,3,3
61	*acsid*,3,4,0
62	*acsid*,3,4,1
63	*acsid*,3,4,2
64	*acsid*,3,4,3
65	*acsid*,4,1,0
66	*acsid*,4,1,1
67	*acsid*,4,1,2

TABLE 1-12 LibraryStation VSM4 Virtual ACS Locations for VTDs

68	*acsid*,4,1,3
69	*acsid*,4,2,0
70	*acsid*,4,2,1
71	*acsid*,4,2,2
72	*acsid*,4,2,3
73	*acsid*,4,3,0
74	*acsid*,4,3,1
75	*acsid*,4,3,2
76	*acsid*,4,3,3
77	*acsid*,4,4,0
78	*acsid*,4,4,1
79	*acsid*,4,4,2
80	*acsid*,4,4,3
81	*acsid*,5,1,0
82	*acsid*,5,1,1
83	*acsid*,5,1,2
84	*acsid*,5,1,3
85	*acsid*,5,2,0
86	*acsid*,5,2,1
87	*acsid*,5,2,2
88	*acsid*,5,2,3
89	*acsid*,5,3,0
90	*acsid*,5,3,1
91	*acsid*,5,3,2
92	*acsid*,5,3,3
93	*acsid*,5,4,0
94	*acsid*,5,4,1
95	*acsid*,5,4,2
96	*acsid*,5,4,3
97	*acsid*,6,1,0
98	*acsid*,6,1,1
99	*acsid*,6,1,2
100	*acsid*,6,1,3
101	*acsid*,6,2,0
102	*acsid*,6,2,1
103	*acsid*,6,2,2
104	*acsid*,6,2,3
105	*acsid*,6,3,0

TABLE 1-12 LibraryStation VSM4 Virtual ACS Locations for VTDs

106	*acsid*,6,3,1
107	*acsid*,6,3,2
108	*acsid*,6,3,3
109	*acsid*,6,4,0
110	*acsid*,6,4,1
111	*acsid*,6,4,2
112	*acsid*,6,4,3
113	*acsid*,7,1,0
114	*acsid*,7,1,1
115	*acsid*,7,1,2
116	*acsid*,7,1,3
117	*acsid*,7,2,0
118	*acsid*,7,2,1
119	*acsid*,7,2,2
120	*acsid*,7,2,3
121	*acsid*,7,3,0
122	*acsid*,7,3,1
123	*acsid*,7,3,2
124	*acsid*,7,3,3
125	*acsid*,7,4,0
126	*acsid*,7,4,1
127	*acsid*,7,4,2
128	*acsid*,7,4,3
129	*acsid*,8,1,0
130	*acsid*,8,1,1
131	*acsid*,8,1,2
132	*acsid*,8,1,3
133	*acsid*,8,2,0
134	*acsid*,8,2,1
135	*acsid*,8,2,2
136	*acsid*,8,2,3
137	*acsid*,8,3,0
138	*acsid*,8,3,1
139	*acsid*,8,3,2
140	*acsid*,8,3,3
141	*acsid*,8,4,0
142	*acsid*,8,4,1
143	*acsid*,8,4,2

TABLE 1-12 LibraryStation VSM4 Virtual ACS Locations for VTDs

144	*acsid*,8,4,3
145	*acsid*,9,1,0
146	*acsid*,9,1,1
147	*acsid*,9,1,2
148	*acsid*,9,1,3
149	*acsid*,9,2,0
150	*acsid*,9,2,1
151	*acsid*,9,2,2
152	*acsid*,9,2,3
153	*acsid*,9,3,0
154	*acsid*,9,3,1
155	*acsid*,9,3,2
156	*acsid*,9,3,3
157	*acsid*,9,4,0
158	*acsid*,9,4,1
159	*acsid*,9,4,2
160	*acsid*,9,4,3
161	*acsid*,10,1,0
162	*acsid*,10,1,1
163	*acsid*,10,1,2
164	*acsid*,10,1,3
165	*acsid*,10,2,0
166	*acsid*,10,2,1
167	*acsid*,10,2,2
168	*acsid*,10,2,3
169	*acsid*,10,3,0
170	*acsid*,10,3,1
171	*acsid*,10,3,2
172	*acsid*,10,3,3
173	*acsid*,10,4,0
174	*acsid*,10,4,1
175	*acsid*,10,4,2
176	*acsid*,10,4,3
177	*acsid*,11,1,0
178	*acsid*,11,1,1
179	*acsid*,11,1,2
180	*acsid*,11,1,3
181	*acsid*,11,2,0

TABLE 1-12 LibraryStation VSM4 Virtual ACS Locations for VTDs

182	*acsid*,11,2,1
183	*acsid*,11,2,2
184	*acsid*,11,2,3
185	*acsid*,11,3,0
186	*acsid*,11,3,1
187	*acsid*,11,3,2
188	*acsid*,11,3,3
189	*acsid*,11,4,0
190	*acsid*,11,4,1
191	*acsid*,11,4,2
192	*acsid*,11,4,3
193	*acsid*,*12*,1,0
194	*acsid*,12,1,1
195	*acsid*,12,1,2
196	*acsid*,12,1,3
197	*acsid*,12,2,0
198	*acsid*,12,2,1
199	*acsid*,12,2,2
200	*acsid*,12,2,3
201	*acsid*,12,3,0
202	*acsid*,12,3,1
203	*acsid*,12,3,2
204	*acsid*,12,3,3
205	*acsid*,12,4,0
206	*acsid*,12,4,1
207	*acsid*,12,4,2
208	*acsid*,12,4,3
209	*acsid*,13,1,0
210	*acsid*,13,1,1
211	*acsid*,13,1,2
212	*acsid*,13,1,3
213	*acsid*,13,2,0
214	*acsid*,13,2,1
215	*acsid*,13,2,2
216	*acsid*,13,2,3
217	*acsid*,13,3,0
218	*acsid*,13,3,1
219	*acsid*,13,3,2

TABLE 1-12 LibraryStation VSM4 Virtual ACS Locations for VTDs

220	*acsid*,13,3,3
221	*acsid*,13,4,0
222	*acsid*,13,4,1
223	*acsid*,13,4,2
224	*acsid*,13,4,3
225	*acsid*,14,1,0
226	*acsid*,14,1,1
227	*acsid*,14,1,2
228	*acsid*,14,1,3
229	*acsid*,14,2,0
230	*acsid*,14,2,1
231	*acsid*,14,2,2
232	*acsid*,14,2,3
233	*acsid*,14,3,0
234	*acsid*,14,3,1
235	*acsid*,14,3,2
236	*acsid*,14,3,3
237	*acsid*,14,4,0
238	*acsid*,14,4,1
239	*acsid*,14,4,2
240	*acsid*,14,4,3
241	*acsid*,15,1,0
242	*acsid*,15,1,1
243	*acsid*,15,1,2
244	*acsid*,15,1,3
245	*acsid*,15,2,0
246	*acsid*,15,2,1
247	*acsid*,15,2,2
248	*acsid*,15,2,3
249	*acsid*,15,3,0
250	*acsid*,15,3,1
251	*acsid*,15,3,2
252	*acsid*,15,3,3
253	*acsid*,15,4,0
254	*acsid*,15,4,1
255	*acsid*,15,4,2
256	*acsid*,15,4,3

HSC CDS DASD Space

Before installing VTCS, you must calculate the DASD space required for the HSC control data set (CDS). The DASD space for the CDS must be increased to accommodate your VSM system's resource definitions. The additional number of 4k blocks required in the CDS for VTCS can be expressed as:

- **For B format CDSs**:

  (number of VTVs /58) + (number of MVCs/71) + 17(number of VTSS) + number of configured MVC ranges + number of configured VTV ranges + 13

- **For C, D, and E format CDSs**:

  (number of VTVs /23) + (number of MVCs/37) + 17(number of VTSS) + number of configured MVC ranges + number of configured VTV ranges + 13

- **For F and G format CDSs**:

  (# VTV ranges) + (# VTV ranges)/862 + (# VTVs defined)/23 + (# VTVs defined)/19826 + (# MVC ranges) + (# MVCs defined)/37 + 18*(# of VTSSs) + 14

Tape Management System DASD Space

To accommodate your VSM system's VTVs, you may need to increase the DASD space for your tape management system. After you determine the number and range of VTVs your VSM system requires, see your tape management system documentation for specific information on calculating the DASD space requirements.

VSM Candidate Data Sets

Your StorageTek representative will run the VSM pre-sales planning tool to identify VSM candidate data sets. You choose a method to route these data sets to VSM as described in "Routing Data Sets to VSM" on page 160.

HSC COMMPath METHod Value

To optimize performance, StorageTek recommends that you set the HSC COMMPath METHod parameter to either LMU or VTAM, *not* to CDS to allow even sharing of resources in a multi-host configuration as shown in the example in "Updating the HSC PARMLIB Member (SLSSYSxx)" on page 146.

Data Chaining a VTD Read Forward or Write Command

Note that when data chaining a Read Forward or Write command, the VTSS requires the minimum data chained update count.

CHAPTER 2

Planning VTCS Operating Policies

We're going to take one more pause and talk about VTCS Operating Polices now, even though you don't actually implement them until you get to Chapter 7, "Configuring VTCS". That's because here you're doing your final planning tasks, and starting with "Preparing for Installation" on page 85, you're just doing stuff and checking the boxes.

Operating policies...one way to approach operating policies is to take the defaults, run for a while, and see what's working/not working and adjust accordingly[1]. Or you can take a quick glance at this chapter, which is designed as a sort of Cliff's Notes on VTCS policies. The idea of a quick once-over is to see how the policies can affect your site's operations...and either adjust as needed or take the defaults and make a mental note to come back and check up on things in a week or two.

The following sections describe VTCS operating policies:

- "VTSS Policies" on page 52
- "VTV Policies" on page 70
- "MVC Policies" on page 72

Note that these are **rough** groupings only. All the pieces are parts of VSM interact with each other constantly, so it's impossible to talk VTVs without thinking about MVCs, and vice versa. These policies are all set globally on the CONFIG statement[2], some of which you can override on a per job basis.

1.Exceptions: you must set MINMIG and MAXMIG.

2.There is one exception, and that's VTV size (MAXVtvsz on MGMTclas).

VTSS Policies

The following sections describe these VTSS policies:

- "AMT Settings" on page 53. AMT settings are a biggie because they're the triggers for starting and stopping automatic migration and all its attendant functions. On this one, you have a global setting that you can override with the SET MIGOPT command.
- "Deleting Scratched VTVs" on page 55, which can free buffer space, but has some fine print associated, so please read the usage notes before enabling this feature.
- "VTV Residency Interval before Automatic Migration Candidacy" on page 56. A method to influence VTV availability.
- "Immediately Migrate VTVs On Dismount" on page 58. The opposite of the above bullet, a method to quickly safeguard your data.
- "MVC Retain Interval" on page 59. A method to reduce MVC mounts.
- "VTV Page Size" on page 59 and "Maximum VTV Size" on page 60. These two are linked, and even though they are VTV policies, they are more about optimizing VTSS usage, so they're discussed here. Large VTV page sizes, when matched with the appropriate VTSS model, can optimize performance within the VTSS. VTV page sizes are about the page data size within the VTSS, not the actual VTV size. VTV page sizes, however, are actively linked to VTV sizes, which can improve performance in high-capacity VTSSs. In fact, selecting VTV sizes of 2 or 4 GB **enforces** a Large VTV Page.
- "Maximum RTDs per VTSS" on page 62. Your choices are 16 or 32 maximum, and it's a CONFIG GLOBAL setting.
- "Maximum and Minimum Concurrent Migration Tasks" on page 67. Something to pay attention to because a thoughtful setting can help optimize your VSM resources. Again, you have a global setting that you can override with the SET MIGOPT command.
- "Stacked Migrates" on page 68. Enabling stacked migrates can improve migration performance, and it's also a CONFIG GLOBAL setting.
- "Controlling Migration Workloads" on page 69 which you do with MIGRSEL SCHLIMIT.

AMT Settings

TABLE 2-1 AMT Settings

This policy does the following...	Valid values are...	The default is...	To set the policy, use...
Controls the automatic space management/migration cycle. This cycle begins when DBU (Disk Buffer Utilization) exceeds the high AMT (HAMT) or as described in TABLE 2-2. AMTs are a percentage of DBU.	▪ LAMT - 5 - 95, must be one less than HAMT. ▪ HAMT - 5 - 95, must be one greater than LAMT.	▪ LAMT - 70 ▪ HAMT - 80	▪ CONFIG VTSS LOW and HIGH parameters ▪ SET MIGOPT HIGHthld and LOWthld parameters.

TABLE 2-2 VTSS Models Maximum VTVs and Automigration Thresholds

VSM Model	Maximum VTVs	Autmigration Starts Above...
VSM2/VSM3	100,000	97,000
VSM4/VSM5	300,000	291,000
VSMc Models: ▪ VSM5-45TB-IFF3 ▪ VSM5-68TB-IFF3 ▪ VSM5-90TB-IFF3	500,000	485,000

Usage Notes

- With CONFIG, AMT settings take effect when you start HSC and apply to the specified VTSS.
- With SET MIGOPT:
  - AMT settings take effect immediately and apply to the specified VTSS or if no VTSS is specified, to all VTSSs. If you try to set global values (no VTSS specified) and the values are not valid for one VTSS (for example, MAXMIG(5) and one VTSS only has 4 RTDs connected), VTCS will not set values for any VTSSs
  - You can set the LAMT, the HAMT, or both.
- The following are general guidelines for changing the defaults:
  - The *difference* between the high and low AMTs affects the duration of the space management/migration cycle.
  - *Lowering* the HAMT tends to trigger *more frequent* space management/migration cycles.
  - *Raising* the HAMT tends to trigger *less frequent* space management/migration cycles.
  - *Lowering* the LAMT tends to free more VTSS space *and* migrate *more* VTVs.
  - *Raising* the LAMT tends to keep more VTVs resident in VTSS space *and* migrate *fewer* VTVs.

Tip – You can use Display VTSS to display the DBU, HAMT, and LAMT for each VTSS in your system. You can also use Display MIGrate to display migration status.

Deleting Scratched VTVs

TABLE 2-3 Delete Scratched VTVs Settings

This policy does the following...	Valid values are...	The default is...	To set the policy, use...
Specifies whether VSM deletes scratched VTVs.	NO, YES	NO	MGMTCLAS DELSCR

Usage Notes

- Specifying DELSCR YES causes VSM to delete scratched VTVs, which frees VTSS buffer space.

Caution –

- When you scratch a VTV with DELSCR YES attribute, VSM erases the VTV data at scratch synchronization time, which eliminates the ability to “unscratch” a VTV to recover data!
- Also note that when using previous releases of HSC SLUCONDB to perform scratch synchronization, SLUCONDB attempted to scratch everything that was marked scratch in the TMS database. For HSC 6.0 and above, however, SLUCONDB has been updated to scratch only those volumes that are not in scratch status in the HSC CDS. Therefore, for HSC 6.0 and above, the only possibilities of inadvertently scratching a VTV resulting in data loss at scratch synchronization time are as follows:
  - If you are running the HSC SLUADMIN Scratch Update Utility at the same time that SLUCONDB is running.
  - If you do not specify the current TMS database and/or the current HSC CDS when using SLUCONDB.

  For more information about HSC scratch synchronization with the Scratch Conversion Utility (SLUCONDB), see HSC System Programmer’s Guide for MVS.

  Also note that for HSC and MVS/CSC, the DELDISP parameter has two values that affect how HSC manages the scratch status of VTVs and real volumes that were mounted scratch and the delete disposition on the dismount message is delete ('D').
- For more information about ExLM scratch synchronization with the SYNCVTV function, see the *ExLM User’s Guide*.

VTV Residency Interval before Automatic Migration Candidacy

TABLE 2-4 VTV Residency Interval

This policy does the following...	Valid values are...	The default is...	To set the policy, use...
By default, VSM selects VTVs for migration. You can, however, specify how long (in hours) VTCS attempts to keep a VTV as VTSS-resident before it becomes an automatic migration candidate.	1 - 9999	none	MGMTCLAS RESTIME

Usage Notes

- The RESTIME value in a VTV's Management Class sets the recommended interval that the VTV remains VTSS-resident from the time that instance of the VTV is created. A new instance of the VTV is created whenever the VTV is updated. At automigration time, the creation date and time of the VTV instance plus the RESTIME value is compared to the TOD clock to determine if the VTV is an automatic migration candidate.
- Note the following:
  - A VTV's Management Class (and attributes, such as RESTIME) is set after a scratch mount or optionally after a specific mount if VTVattr = ALLmount.
  - The RESTIME value is only a recommendation. VTCS can migrate a VTV before its residency interval expires if the DBU has not reached the LAMT or the specified migrate-to-threshold value and no VTVs have expired their residency intervals.
  - You can do a demand migrate of a VTV and delete it from the VTSS even if its residency interval has not expired.
  - The RESTIME and IMMEDmig(DELETE) parameters are mutually exclusive.

The following example shows how the RESTIME parameter works:

1. You create Management Class with a RESTIME of 10 hours.
2. A job requests a scratch mount for the Management Class you created in Step 1 VTCS selects and mounts a scratch VTV. The VTV is updated, so at dismount time, its RESTIME value is set to 10 hours (which began when VTCS mounted the VTV).
3. VTCS migrates the VTV after 3 hours, then recalls the VTV 2 hours later for a read. The RESTIME value is not reset, and there are now 5 hours of residency remaining.
4. 2 hours later, a job updates the VTV, which was 7 hours old. The update creates a new instance of the VTV and the residency interval will restart from the time the VTV was mounted for update.
5. 24 hours later, VTCS migrates the VTV, then recalls it 2 days later for a read. VTCS does not create a new instance of the VTV because it is not updated. The residency interval has expired and the VTV is therefore an automatic migration candidate based only on least-recently-used/size criteria.
6. A week later, the VTV is scratched. VTCS eventually selects and mounts the VTV to satisfy a scratch mount request. If the VTV is updated, its residency interval is set to the RESTIME value of the Management Class being used.

Immediately Migrate VTVs On Dismount

TABLE 2-5 Immediately Migrate VTVs on Dismount

This policy does the following...	Valid values are...	The default is...	To set the policy, use...
Specifies whether VSM will immediately schedule a VTV for migration after dismounting it. When the migration actually occurs depends on RTD availability, Storage Classes for immediate migration, and the total number of immediate migrates scheduled.	▪ NO - do not immediately migrate ▪ KEEP - immediately migrate and keep a VTSS resident copy of the VTV ▪ DELETE - immediately migrate and delete VTVs from the VTSS after migration.	NO	MGMTCLAS IMMEDMIG

Usage Notes

The following are guidelines for setting IMMEDmig:

- Specify NO (the default) if you do not want immediately migration and you do want other migration policies to determine your migration strategy.
- Specify KEEP if you want immediate migration and want to keep copies of the migrated VTVs resident on the VTSS until they become eligible for deletion.

Caution – IMMEDmig KEEP ensures that VTVs are immediately migrated and kept VTSS-resident; however, it does not free up VTSS space, may increase I/O to the RTDs, uses up MVC space more quickly, and may also increase the need for MVC space reclamation.

- Specify DELETE if you want immediate migration and want to delete VTVs from the VTSS after migration.

Caution – IMMEDmig DELETE ensures that VTVs are immediately migrated and frees VTSS space; however, it preferences migration processing, may increase I/O to the RTDs, uses up MVC space more quickly, and may also increase the need for MVC space reclamation and VTV recalls.

- The RESTIME and IMMEDmig(DELETE) parameters are mutually exclusive; for more information, see “VTV Policies” on page 70.
- IMMEDmig KEEP and IMMEDmig DELETE are mutually exclusive with CONFIG HOST NOMIGRAT. If you specify both, the IMMEDmig value overrides NOMIGRAT (for only those VTVs with the IMMEDmig value), and VTCS does not issue a message about this override.
- If VTCS stops with pending immediate migrations, these migrations will resume when VTCS restarts.

MVC Retain Interval

TABLE 2-6 MVC Retain Interval

This policy does the following...	Valid values are...	The default is...	To set the policy, use...
Specifies how long (in minutes) VTCS will retain an MVC on an RTD in idle mode after a migration. Retaining the MVC can reduce MVC mounts. **Note**: When VTCS shuts down, VTCS dismounts all MVCs regardless of the MVC retain interval.	1 - 60	10	CONFIG VTSS RETAIN

VTV Page Size

TABLE 2-7 VTV Page Size

This policy does the following...	Valid values are...	The default is...	To set the policy, use...
Specifies the default page size used to store VTV data in the VTSS and on the MVCs for 400 and 800 MB VTVs only. For 2 and 4 GB VTVs (MAXVtvsz 2000 or 4000) a VTPAGE setting of LARGE is always used.	STANDARD or LARGE	STANDARD	CONFIG GLOBAL VTVPAGE MGMTclas VTVPAGE

Usage Notes

- Large page size (which requires that the CDS is at a G level or above) can provide improved performance within the VTSS and for migrates and recalls. For more information about CDS levels, see "CDS VTCS Level" on page 32.

Caution –

- The page size of a VTV can **only** be changed by a VTV scratch mount. Additional restrictions may also apply for scratch VTVs that were previously resident in a VTSS.
- VTVPAGE **does not** apply to VSM2s. The VTSS microcode requirements are as follows:
  - For VSM3s: microcode level N01.00.77.00 or higher.
  - For VSM4s/VSM5s: microcode level D02.02.00.00 or higher.
- If you specify LARGE and the CDS level and/or VTSS microcode **do not** support LARGE, VTCS issues warning messages and VTVPAGE defaults to STANDARD.
- Creating VTVs with large pages makes these VTVs **unreadable** in configurations that do not support large VTV pages.

Maximum VTV Size

TABLE 2-8 Maximum VTV Size

This policy does the following...	Valid values are...	The default is...	To set the policy, use...
Specifies the maximum size of a VTV.	400 - 400 Mb. 800 - 800 Mb. The CDS must be at a D level or above. 2000 - 2 Gb. The CDS must be at a G level or above. 4000 - 4 Gb. The CDS must be at a G level or above.	400	CONFIG GLOBAL MAXVTVSZ, MGMTCLAS MAXVTVSZ

Usage Notes

- The size of a VTV changes *only* after it goes through a scratch cycle. Therefore, if you change the Management Class and DISP=MOD, then it will still retain the original size.
- If you specify a VTV size that is not supported by the configuration, VTCS issues warning messages and MAXVtvsz defaults to the largest VTV size supported by the configuration.
- MAXVtvsz **does not** apply to VSM2s. The VSM3/VSM4 microcode requirements are as follows:
  - For VSM3s and 800MB support: microcode level N01.00.69.04 or microcode level N01.00.71.00 and above.
  - For VSM4s and 800MB support: microcode level D01.00.04.03 or microcode level D01.00.06.03 and above.
  - For VSM3s and 2/4GB support: microcode level N01.00.77.00 or higher.
  - For VSM4s/VSM5s and 2/4GB support: microcode level D02.02.00.00 or higher.
- The `CONFIG GLOBAL` and `MGMTCLAS MAXVTVSZ` parameters interact as follows:
  - If `MAXVTVSZ` **is** specified on `MGMTCLAS`, this value overrides the `CONFIG GLOBAL MAXVTVSZ` value.
  - If `MAXVTVSZ` is **not** specified on `MGMTCLAS`, the `CONFIG GLOBAL MAXVTVSZ` value, if specified, is used. Otherwise, `MAXVTVSZ` defaults to 400MB.
  - If `MAXVTVSZ` is **not** specified on `MGMTCLAS` **or** on `CONFIG GLOBAL, MAXVTVSZ` defaults to 400MB.

Caution – Specifying 2GB or 4GB VTVs:

- Increases MVC usage.
- May degrade space management of smaller-capacity VTSSs, which have relatively small cache and buffer sizes.
- Increases delays to jobs waiting recalls. Although the actual data transfer time from an MVC is the same and there may be fewer interruptions, each interruption lasts longer, which can cause job time-outs.

Maximum RTDs per VTSS

You can connect up to 32 RTDs per VTSS.

TABLE 2-9 Maximum RTDS per VTSS - 16 or 32

This policy does the following...	Valid values are...	The default is...	To set the policy, use...
Specifies the maximum RTDs per VTSS.	16, 32	16	CONFIG GLOBAL MAXRTDS

Usage Notes

- For maximum 32 RTDs support, you must fulfill the requirements described in TABLE 2-10.

TABLE 2-10 32 RTDs Support Requirements for ELS 7.0

32 RTDs Support requires...	..the following VSM4/VSM5 microcode...	...and the following VTCS/NCS 6.2 PTFs...	...and CDS level...
FICON ports for the CLINKs	D02.05.00.00 or higher	L1H13ZF (SOS6200) L1H13ZG (SWS6200	"F" or higher

- The VTCS addressing scheme for maximum 32 RTDs is different than that for maximum 16 RTDs. For more information, see "RTD/CLINK Addresses - Maximum 32 RTDs" on page 41.

RTD/CLINK Addresses - Maximum 32 RTDs

VSM5 is available **only** with 8 VCF (FICON) cards in the configuration for a maximum of 32 RTDs shown in FIGURE 2-1.

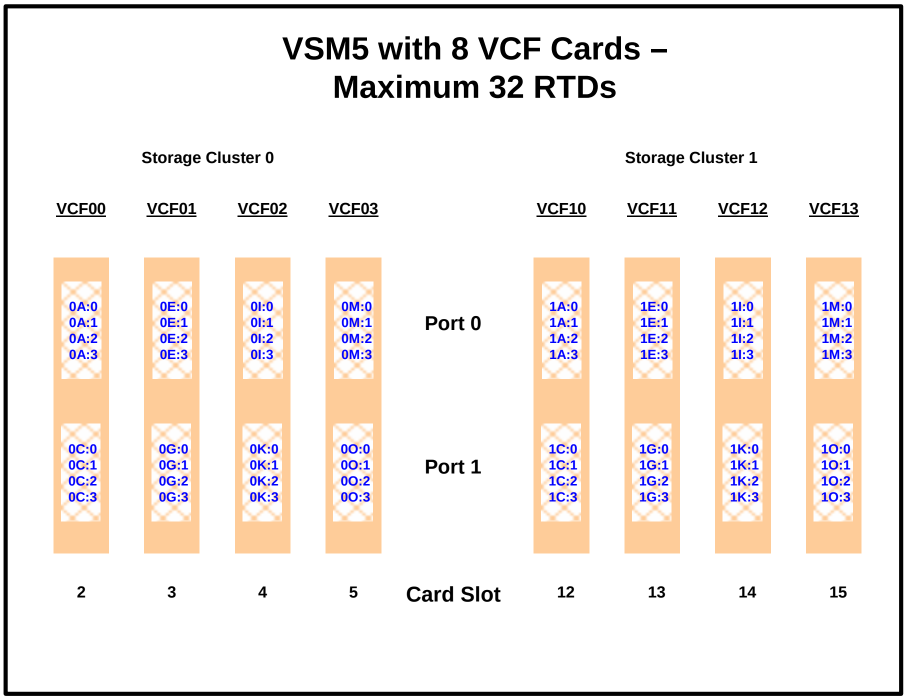


FIGURE 2-1 VSM5 with 8 VCF cards - Max 32 RTDs

As FIGURE 2-1 shows, each FICON interface supports 4 devices attached via FICON directors. **Note that** the addressing scheme used by VTCS **is different** from that used for support of 2 devices per interface. The device addresses are now in the format *CI*:*R*, where:

- *C* is the cluster number (0 or 1)
- *I* is the interface number (A, C, E, G, I, K, M, or O)
- *R* is the device number on the interface (0, 1, 2, or 3).

TABLE 2-11 on page 64 shows the "old" address (maximum 16 RTDs) and its corresponding "new" address (maximum 32 RTDs).

Note – If you upgrade an existing configuration from maximum 16 to maximum 32 RTDs, you have to change the addresses in your CONFIG deck on your RTD statements, your CLINK statements, or both.

TABLE 2-11 RTD/CLINK Addresses - Maximum 32 RTDs

Cluster Number	Interface	RTD/CLINK	Old Address (Maximum 16 RTDs)	New Address (Maximum 32 RTDs)
0	A	0	0A	0A:0
0	A	1	0B	0A:1
0	A	2	-	0A:2
0	A	3	-	0A:3
0	C	0	0C	0C:0
0	C	1	0D	0C:1
0	C	2	-	0C:2
0	C	3	-	0C:3
0	E	0	0E	0E:0
0	E	1	0F	0E:1
0	E	2	-	0E:2
0	E	3	-	0E:3
0	G	0	0G	0G:0
0	G	1	0H	0G:1
0	G	2	-	0G:2
0	G	3	-	0G:3
0	I	0	0I	0I:0
0	I	1	0J	0I:1
0	I	2	-	0I:2
0	I	3	-	0I:3
0	K	0	0K	0K:0
0	K	1	0L	0K:1
0	K	2	-	0K:2
0	K	3	-	0K:3
0	M	0	0M	0M:0
0	M	1	0N	0M:1
0	M	2	-	0M:2
0	M	3	-	0M:3
0	O	0	0O	0O:0
0	O	1	0P	0O:1
0	O	2	-	0O:2
0	O	3	-	0O:3
1	A	0	1A	1A:0
1	A	1	1B	1A:1
1	A	2	-	1A:2
1	A	3	-	1A:3

TABLE 2-11 RTD/CLINK Addresses - Maximum 32 RTDs

1	C	0	1C	1C:0
1	C	1	1D	1C:1
1	C	2	-	1C:2
1	C	3	-	1C:3
1	E	0	1E	1E:0
1	E	1	1F	1E:1
1	E	2	-	1E:2
1	E	3	-	1E:3
1	G	0	1G	1G:0
1	G	1	1H	1G:1
1	G	2	-	1G:2
1	G	3	-	1G:3
1	I	0	1I	1I:0
1	I	1	1J	1I:1
1	I	2	-	1I:2
1	I	3	-	1I:3
1	K	0	1K	1K:0
1	K	1	1L	1K:1
1	K	2	-	1K:2
1	K	3	-	1K:3
1	M	0	1M	1M:0
1	M	1	1N	1M:1
1	M	2	-	1M:2
1	M	3	-	1M:3
1	O	0	1O	1O:0
1	O	1	1P	1O:1
1	O	2	-	1O:2
1	O	3	-	1O:3

Maximum and Minimum Concurrent Migration Tasks

TABLE 2-12 Maximum and Minimum Concurrent Migration Tasks

This policy does the following...	Valid values are...	The default is...	To set the policy, use...
Specifies the maximum and minimum number of concurrent automatic migration, immediate migration, and migrate-to-threshold tasks for each VTSS.	MAXMIG - 1 to the number of RTDs connected to the VTSS. See "Maximum RTDs per VTSS" on page 62 for more information.	Half the number of VTDs connected to the VTSS.	CONFIG VTSS MAXMIG or SET MIGOPT
	MINMIG - 1 to the MAXMIG setting.	1	CONFIG VTSS MINMIG or SET MIGOPT

Usage Notes

- Use these parameters to balance migration tasks with other tasks (such as recall and reclaim) for the RTDs you have defined for each VTSS.
- In some situations, VTCS may not be able to activate *all* the migration tasks specified by the MAXMIG parameter. For example:
  - The VSM-wide RTD configuration consists of four 9840 and four 9490 transports.
  - No Storage Classes with STK1R as the primary media have been defined.
  - There is sufficient MVC media for the 9490 transports.

  In this configuration, because only the 9490 media is used, only a maximum of four migration tasks are activated using the 9490 RTDs.
- Similarly, there are circumstances when VTCS will start *less* than the number of migration tasks specified by the MINMIG parameter. For example:
  - The configuration consists of a single VTSS with 4 RTDs in ACS 0 and 4 RTDs in ACS 1. All RTD device types are identical.
  - MINMIG and MAXMIG are both set to 8.
  - Two Storage Classes are defined, which point, respectively, to ACS 0 and ACS 1.

  In this configuration, if there are migrates queued for both Storage Classes, then VTCS will start 8 requests. If however, there are only migrations queued for one Storage Class, then VTCS *will not* start 8 requests because the workload can only be serviced by one Storage Class and this class can only run on 4 RTDs.
- Finally, also note that when you reset the MINMIG and/or MAXMIG values, the actual number of migration tasks may not be immediately affected because of the way that VTCS manages migration tasks.

Tip – You can use Display MIGrate to display migration status.

Stacked Migrates

With the prerequisites described in TABLE P-8 on page viii, you can enabled stacked migrates.

TABLE 2-13 Stacked Migrates Policy

This policy does the following...	Valid values are...	The default is...	To set the policy, use...
Specifies whether stacked migrates is enabled	YES, NO	NO	CONFIG GLOBAL FASTMIGR

Usage Notes

Enabling the stacked migrates feature can improve migration performance by allowing multiple migrations concurrently to an RTD.

Controlling Migration Workloads

TABLE 2-14 Controlling Migration Workloads

This policy does the following...	Valid values are...	The default is...	To set the policy, use...
Controls migration workloads.	0-99	99 (no limit up to the VTSS MAXMIG value)	MIGRSEL SCHLIMIT

Usage Notes

MIGRSEL SCHLIMIT provides finer control over how Storage Classes are treated for migration, but in the opposite direction. That is, MIGRSEL SCHLIMIT, applies limits on migration resources (RTDs/MVCs) per Storage Class, and you can do so on a VTSS and/or host basis. Use MIGRSEL SCHLIMIT, as follows:

- Use MIGRSEL SCHLIMIT to depreference migration per Storage Class. Lower values depreference migration, and you can specify automatic, immediate, demand, and reclaim migrates. Lower values can do the following:
  - Optimize MVC usage. You can use this strategy where you want to limit the number MVCs used. Example: a workload migrated to a Storage Class that is subsequently used for offsite vaulting.
  - Preference migration to other Storage Classes.
  - Restricts migration concurrency to better use MVCs.
  - Limits migration to keep RTDs available for auto recalls.
  - Reduce MVC swapping when workloads change.

- MIGRSEL SCHLIMIT allows you to specify the VTSS and host to which the preferencing applies.

- For auto and immediate migration processing, MIGRSEL SCHLIMIT depreferences migration for the VTSS to Storage Class relationship. This comparison is not global and only effects requests driven by the individual VTCS host

- For demand migration requests, MIGRSEL SCHLIMIT it will cause the request to be held if the scheduling of it would cause the number of globally active migration requests on the VTSS that satisfy the same FUNCTION and STORCLAS selection criteria to be exceeded. The migration requests will be released and a MVC picked once the constraint subsides.

VTV Policies

The following sections describe these VTV policies, roughly in order of importance:

- Maximum VTVs per MVC. Take the default, unless...well, read the fine print about high-capacity media.
- “Hosts Disabled from Migration, Consolidation and Export by VTV or Management Class” on page 71. I don’t really have a “for instance” for this one. If, for whatever reason, you don’t want a host to do these things you can explicitly turn them off. The default is not to disable any hosts, so if this isn’t an issue, don’t worry about it.
- “Recall VTVs with Read Data Checks” on page 71. The default is to recall VTVs with Read Data Checks...might as well try, right? If this doesn’t suit you, you can turn it off either globally or per specific operation.

Maximum VTVs per MVC

TABLE 2-15 Maximum VTVs per MVC

This policy does the following...	Valid values are...	The default is...	To set the policy, use...
Specifies the maximum number of VTVs that can be migrated to a single MVC.	• 4 to 32000 for a D, E or F level CDS. • 4 to 65000 for a G level CDS.	• 32000 for a D, E or F level CDS. • 65000 for a G level CDS.	CONFIG GLOBAL MAXVTV parameter

Usage Notes

- This policy applies to all MVCs and, at the time you set the policy, applies only to future migrations. That is, it will not lower the number of VTVs already migrated to an MVC. If the policy is not specified, the default is as shown in TABLE 2-15 unless the available MVC space is less than any remaining current VTSS resident VTV.
- Generally, use the default to allow VSM to automatically manage VTV stacking. However, with high-capacity media (for example, in a VSM system where all MVCs are type STK2P), specifying a value lower than the maximum may improve recall performance in some situations. Note, however, that a very low value can reduce that percentage of usable MVC space. If the maximum VTVs per MVC is exceeded, then usable space is reported as 0%.
- For more information about CDS levels, see “CDS VTCS Level” on page 32.

Hosts Disabled from Migration, Consolidation and Export by VTV or Management Class

TABLE 2-16 Hosts Disabled from Migration, Consolidation, or Export by VTV or MGMTclas

This policy does the following...	Valid values are...	The default is...	To set the policy, use...
Specifies that a host cannot initiate automatic and demand migration and consolidation processing, or export by VTV or Management Class.	NOMIGRAT, if not specified, migration, etc., is not disabled	do not disable	CONFIG HOST

Usage Notes

- Specifying NOMIGRAT also causes NORECLAM to be set; for more information, see “Hosts Disabled from Reclamation” on page 78.
- MGMTclas IMMEDmig KEEP and IMMEDmig DELETE are mutually exclusive with CONFIG HOST NOMIGRAT. If you specify both, the IMMEDmig value overrides NOMIGRAT (for only those VTVs with the IMMEDmig value), and VTCS does not issue a message about this override.

Recall VTVs with Read Data Checks

TABLE 2-17 Recall VTVs with Read Data Checks

This policy does the following...	Valid values are...	The default is...	To set the policy, use...
Specifies whether VTCS recalls VTVs with read data checks.	YES - Recall VTVs with Read Data Checks. NO - Do not recall VTVs with Read Data Checks.	YES	CONFIG GLOBAL RECALWER CONsolid GLOBAL RECALWER EXPORT GLOBAL RECALWER MVCDRAIN GLOBAL RECALWER RECALLG GLOBAL RECALWER

Note – During MVC reclaims, VTCS will never recall VTVs with read data checks, regardless of the RECALWER setting on the CONFIG GLOBAL statement.

MVC Policies

The following sections describe these MVC policies...and some of these take some Deep Thought, so take your time, and remember, it's usually a good idea to supplement automatic reclamation with demand reclamation as described in ...

- "MVC Fragmented Space Threshold - Determines MVC Eligibility for Reclamation" on page 73. It makes sense that you only want to go for the low-hanging fruit when you do reclamation, because reclamation costs you resources that you could be otherwise using for migrates and recalls. This is the dial you can use to specify how fragmented an MVC must be to make it a reclaim candidate.
- "VMVC Fragmented Space Threshold - Determines VMVC Eligibility for Reclamation" on page 74, as above, only this dial is for VMVCs on VLE appliances.

  Note that these values only dictate when an MVC or VMVC becomes a reclaim candidate. It does not **start** automatic reclamation; see the next two bullets.
- "Free MVCs Threshold - Starts Automatic Space Reclamation" on page 74 and "Eligible/Total MVCs Threshold - Starts Automatic Space Reclamation" on page 76.

  Yes, folks...there are **two** different triggers that start automatic reclamation. More fine tuning, more granularity. Read the fine print, and you'll see that these triggers supplement rather than compete with each other. If either you run low on Free MVCs or your ratio of eligible to total MVCs gets to a bad number, that's a dangerous situation, and it starts automatic reclamation.
- "Maximum MVCs Processed Per Reclaim" on page 77. Here's where you have to put on your thinking cap. VTCS reclaims MVCs one at a time...serially, that is. You can, however, specify the maximum number you want to reclaim in one run, either automatic or demand. More fine tuning...you can say that when reclamation kicks in, run it to reclaim this number of MVCs, because that's the magic number that will get me out of the ditch.
- "Maximum MVCs Concurrently Processed for Reclamation and Drain" on page 78. Okay, this is a test...what's the difference between this and maximum MVCs processed per reclaim, especially since you can only reclaim one MVC at a time? It turns out that reclaim and drain are separate but related processes that can be processing multiple MVCs, and you get to say how many.
- "Hosts Disabled from Reclamation" on page 78. Again, I can't think of a real world instance where you'd want to do this. If, for whatever reason, you don't want a host to initiate reclamation, you can explicitly turn it off. The default is not to disable any hosts, so if this isn't an issue, don't worry about it.
- "MVC Retain Interval" on page 79. Is this a VTSS policy or an MVC policy? I'm going to call it an MVC policy just because of the title. The idea is pretty simple: if you think you're going to write more data to an MVC once it's mounted and initially written to, you can specify a value that'll make it hang around for a while...
- "MVC Initialization on First Mount" on page 80. If you want to automate new MVC initialization...

Note – Reclamation turns fragmented MVC space (space that contains non-current VTVs) into usable space (writable MVC space). MVC reports and Display MVCpool show the percentages of MVC space that is fragmented, used (space that contains current VTVs), available, and usable. **Note that** usable space may be zero even if there is still space physically available. For example, if the maximum VTVs per MVC is exceeded, then usable space is reported as 0%. You set maximum VTVs per MVC as described in "Maximum VTVs per MVC" on page 70. Similarly, if a data check error has been reported against an MVC, VTCS will not use this MVC for output and usable space is reported as 0%.

MVC Fragmented Space Threshold - Determines MVC Eligibility for Reclamation

TABLE 2-18 MVC Fragmented Space Threshold

This policy does the following...	Valid values are...	The default is...	To set the policy, use...
Specifies the fragmented space threshold (as a percentage) that determines when an MVC is eligible for demand or automatic reclamation.	4 - 98	40	CONFIG RECLAIM THRESHLD RECLAIM THRESHLD

Usage Notes

- If fragmented space on an MVC exceeds the value specified on THRESHLD, VTCS makes the MVC eligible for reclamation. Regardless of the percentage of fragmented space on an MVC versus this value, however, VTCS also considers where fragmented space occurs. For example, if the first fragmented space is near the end of the MVC, VTCS may process the MVC before an MVC with more total fragmented space.
- You can use Display MVCpool to display the MVCs eligible for reclamation in your MVC pool, as well as information about MVC status and space.

VMVC Fragmented Space Threshold - Determines VMVC Eligibility for Reclamation

TABLE 2-19 MVC Fragmented Space Threshold

This policy does the following...	Valid values are...	The default is...	To set the policy, use...
Specifies the fragmented space threshold (as a percentage) that determines when a VMVC on a VLE appliance is eligible for demand or automatic reclamation.	4 - 98	30	CONFIG RECLAIM VLTHRES

Usage Notes

- Reclaim on a VMVC consists of simply deleting the expired VTV images from the VMVC. That is, no recall and re-migrate of the VTV is required. VMVC reclaim is therefore much faster than MVC reclaim, and you can set `VLTHRES` lower (more aggressive) than `THRESHLD`.
- You can use Display MVCpool to display the VMVCs eligible for reclamation in your VMVC pools, as well as information about VMVC status and space.

Free MVCs Threshold - Starts Automatic Space Reclamation

TABLE 2-20 Free MVCs Threshold

This policy does the following...	Valid values are...	The default is...	To set the policy, use...
Specifies the minimum number of free MVCs. If the actual number drops below this value, automatic reclamation starts.	0 - 255	40	CONFIG GLOBAL MVCFREE

Usage Notes

- A free MVC has 100% usable space and does not contain any migrated VTVs.
- VTCS checks the MVCFREE value for each ACS. VTCS issues message SLS6616I and starts an automatic space reclamation if *both* of the following occurs:
  - Free MVCs is equal to or less than the value specified on CONFIG MVCFREE.
  - There is at least one eligible MVC as defined by the CONFIG RECLAIM THRESHLD parameter; for more information, see "Eligible/Total MVCs Threshold - Starts Automatic Space Reclamation" on page 76.
- If you set MVCFREE=0, VTCS actually uses the default value (40).

- StorageTek **strongly recommends** that you ensure that your MVC pool always has *at least* one eligible MVC for each MVC media type.

  Otherwise, you may need to change the CONFIG GLOBAL MVCFREE value, add more MVCs to the pool, or both. You can use Display to display the number of free MVCs in your MVC pool.

Eligible/Total MVCs Threshold - Starts Automatic Space Reclamation

TABLE 2-21 Eligible/Total MVCs Threshold

This policy does the following...	Valid values are...	The default is...	To set the policy, use...
Specifies a percentage value that represents the ratio of reclaim candidates to total MVCs, which triggers automatic space reclamation as described in "Usage Notes" on page 76	1 - 98	35	CONFIG RECLAIM START

Usage Notes

- CONFIG RECLAIM START specifies a percentage value, which is equal to:

  (*Reclaim Candidates*/*Reclaim Candidates* + *Free MVCs*) * 100

  Where:

  Reclaim Candidates

  is the number of Reclaim Candidates determined by the CONFIG RECLAIM THRESHLD parameter. For more information, see "Eligible/Total MVCs Threshold - Starts Automatic Space Reclamation" on page 76.

  Reclaim Candidates + Free MVCs

  equals the number of Reclaim Candidates *plus* the number of free MVCs. A free MVC:

  - Has 100% usable space and does not contain any migrated VTVs.
  - Is defined as described in "Define Available MVCs with CONFIG" on page 23 and "Define the MVC Pool" on page 23.
  - Is writeable.
  - Is resident in the ACS.
- For each ACS (not globally for all ACSs), VTCS issues message SLS6616I and starts an automatic space reclamation if *both* of the following occurs:
  - The actual value of (*Reclaim Candidates*/*Reclaim Candidates* + *Free MVCs*) * 100 exceeds the value specified on CONFIG RECLAIM START parameter.
  - The number of eligible MVCs exceeds the value specified on the MAXMVC parameter; for more information, see "Maximum MVCs Processed Per Reclaim" on page 77.

Note – The only exception to the above two conditions occurs if an SLS6699 message indicates a critical shortage of free MVCs, in which case automatic reclamation will start anyway.

- The following are general guidelines for specifying values for the START parameter:
  - A *low* value (for example, 5%), starts automatic space reclamation when there are *few* eligible MVCs compared to free MVCs *unless* you set the MAXMVC value high compared to the number of eligible MVCs.
  - A *high* value (for example, 95%), starts automatic space reclamation when there are *many* eligible MVCs compared to free MVCs unless you set the MAXMVC value *very* high and your MVC pool is *very* small.
- You can use Display MVCPOOL to display eligible and free MVCs.

Maximum MVCs Processed Per Reclaim

TABLE 2-22 Maximum MVCs Processed Per Reclaim

This policy does the following...	Valid values are...	The default is...	To set the policy, use...
Specifies the maximum number of MVCs that will be processed in a single space reclamation run.	1 - 98	40	CONFIG RECLAIM MAXMVC RECLAIM MAXMVC

Usage Notes

- Automatic and demand space reclamation processes one MVC at a time. You can, however, use MAXMVC to control the **maximum** number of MVCs that will be processed in a single space reclamation run (automatic or demand).
- For automatic space reclamation to start via the CONFIG RECLAIM START parameter setting, the number of eligible MVCs (determined by the CONFIG RECLAIM THRESHLD parameter) must also exceed the MAXMVC value. For more information, see "Eligible/Total MVCs Threshold - Starts Automatic Space Reclamation" on page 76.
- The following are general guidelines for specifying values for the MAXMVC parameter:
  - A *low* value reclaims *fewer* MVCs in a single run, but may have *negligible* effect on migrations and recalls and may start automatic space reclamation *more* frequently; for more information, see "Eligible/Total MVCs Threshold - Starts Automatic Space Reclamation" on page 76.
  - A *high* value reclaims *more* MVCs in a single run, but may have *considerable* effect on migrations and recalls and may start automatic space reclamation *less* frequently; for more information, see "Eligible/Total MVCs Threshold - Starts Automatic Space Reclamation" on page 76.
- You can use Display MVCpool to display eligible and free MVCs.

Maximum MVCs Concurrently Processed for Reclamation and Drain

TABLE 2-23 Maximum MVCs Concurrently Processed for Reclamation and Drain

This policy does the following...	Valid values are...	The default is...	To set the policy, use...
Specifies the maximum number of MVCs concurrently processed for reclamation and drain.	1 - 99	1	■ CONFIG RECLAIM CONMVC ■ MVCDRAIN CONMVC ■ RECLAIM CONMVC

Hosts Disabled from Reclamation

TABLE 2-24 Hosts Disabled from Reclamation

This policy does the following...	Valid values are...	The default is...	To set the policy, use...
Specifies that a host cannot initiate automatic and demand reclamation.	NOMIRECLAM, if not specified, reclamation is not disabled	do not disable	CONFIG HOST NORECLAM

Usage Notes

- Specifying NOMIGRAT also causes NORECLAM to be set; for more information, see "Hosts Disabled from Migration, Consolidation and Export by VTV or Management Class" on page 71.
- Disabled hosts can still do demand MVC drains via MVCDRain.

MVC Retain Interval

TABLE 2-25 MVC Retain Interval

This policy does the following...	Valid values are...	The default is...	To set the policy, use...
Specifies how long VTCS will retain an MVC on an RTD in idle mode after a migration.	1 - 60 minutes	10	CONFIG VTSS RETAIN parameter

Usage Notes

- Retaining the MVC can reduce MVC mounts.
- When VTCS shuts down, VTCS dismounts all MVCs regardless of the MVC retain interval.

MVC Initialization on First Mount

TABLE 2-26 MVC Initialization on First Mount

This policy does the following...	Valid values are...	The default is...	To set the policy, use...
Specifies whether uninitialized MVCs are initialized when first mounted.	NO, YES	NO for CONFIG GLOBAL, none for MVCPOOL	CONFIG GLOBAL INITMVC parameter, MVCPOOL NAME INITMVC parameter

---

Note –

- MVCPOOL INITMVC overrides GLOBAL INITMVC. There is no default for MVCPOOL INITMVC; if not specified for a named MVC Pool the CONFIG GLOBAL value (or default) is used.
- Initialization of MVCs in the DEFAULTPOOL is controlled by the GLOBAL INITMVC specification (or default).
- MVC Initialization applies only to VSM4/5 and requires microcode level D02.05.00.00 or higher. If this level of microcode is not installed on all VTSSs in the configuration, MVC initialization will be limited to the VTSSs that have it installed.

VTV Replication Policies

VTSS CLINKs are capable of replication, which is copying VTVs from one VTSS to another. The following sections describe VTSS replication policies...

- “VTCS Replication - Always or Changed VTV Only” on page 82. Here, the concept is simply, but read the fine print so that you understand what “changed” really means.
- “VTCS Replication - Synchronous or Asynchronous” on page 83. You have another choice with replication (assuming that you have satisfied all the requirements), which is synchronous or asynchronous processing.

VTCS Replication - Always or Changed VTV Only

TABLE 2-27 VTV Replication - Always or Changed VTV Only

This policy does the following...	Valid values are...	The default is...	To set the policy, use...
Specifies whether a VTV is always replicated or only when changed.	ALWAYS, CHANGED	ALWAYS	CONFIG GLOBAL REPlicat

Usage Notes

- ALWAYS means the replicate request is added to the VTCS replication queue every time the VTV is dismounted, regardless of whether the VTV was changed while it was mounted (the default).
- CHANGED means the replicate request is added to the VTCS replication queue if the VTV:

  Was changed while it was mounted **or**

  Was only read while mounted but less than the expected number of MVC copies of the VTV exist.

Note – Regardless of the CONFIG GLOBAL REPlicat setting, replication **also** requires that:

- The VTV must be dismounted in a VTSS that supports replication **and** there cannot be an identical copy of the VTV in the other VTSS in the Cluster.
- In addition to the CONFIG GLOBAL REPlicat value, you **must** specify REPlicat(YES) on a VTV's Management Class for replication to occur.

VTCS Replication - Synchronous or Asynchronous

TABLE 2-28 VTV Replication - Synchronous or Asynchronous

This policy does the following...	Valid values are...	The default is...	To set the policy, use...
Specifies whether synchronous replication is enabled.	YES, NO	NO	CONFIG GLOBAL SYNCHREP

Usage Notes

YES means that synchronous replication is enabled...but make sure you have all the requirements lined up as described in TABLE P-10 on page ix.

Note –

- **Regardless** of the CONFIG GLOBAL SYNCHREP setting, replication **also** requires that you **must** specify REPlicat(YES) on a VTV's Management Class for replication to occur.
- Replication **does incur** a performance penalty, as follows:

Caution – With synchronous replication the time required to replicate a virtual volume will delay the completion of any job creating data that has a synchronous replication policy.

CHAPTER 3

Preparing for Installation

Almost there...in the next chapter, we actually start installing software! But first, there are several items you need to get done in MVS to ensure a successful installation and configuration. So **before** doing the tasks described in Chapter 4 “Installing NCS and VTCS”, complete the preparation tasks described in the following sections:

- “Defining A Security System User ID for HSC, SMC, and VTCS” on page 86
- “Configuring MVS Device Numbers and Esoterics” on page 87
- “Setting the MVS Missing Interrupt Handler (MIH) Value” on page 88
- “Specifying the Region Size” on page 88

Note – Most of the tasks in this Chapter require you to specify VSM system values that you determined on page 10 and recorded in TABLE A-1 on page 165.

Defining A Security System User ID for HSC, SMC, and VTCS

VSM software (HSC, SMC, and VTCS) uses the MVS System Authorization Facility (SAF) to control the usage of MVCs. Before mounting an MVC and before issuing writes to an MVC, VSM issues a SAF query to verify that the HSC user has UPDATE authority for the MVC.

You must define a security system user ID to be used by the SAF queries. All VSM SAF queries are issued on behalf of HSC, so the security system user ID must be associated with the HSC started task. Refer to your security system documentation for details on how to associate a security system user ID with the HSC started task.

You must also set up TAPEVOL profiles to ensure that VSM is authorized to mount MVCs, to ensure that VSM is authorized to update MVCs, and to guard against accidental overwrites of MVCs; for more information, see "Defining MVC Pool Volser Authority" on page 157.

Caution – Depending on the default settings of your security system, VSM may not be able to mount and to write to MVCs until you have defined a security system user ID for HSC and TAPEVOL profiles for the MVCs.

Configuring MVS Device Numbers and Esoterics

The following sections tell how to use the HCD facility to do the following:

- Assign MVS device numbers to VTDs and shared RTDs.
- Associate VTD MVS device numbers and esoterics.

You determined these values in "Determining VSM Configuration Values" on page 11 and recorded them in TABLE A-1 on page 165. See your IBM documentation for more information on the HCD facility.

Assigning MVS Device Numbers to VTDs

Use the HCD facility to assign MVS 3490E device numbers to your VSM system's VTDs. You determined these device numbers in "VTD Unit Addresses" on page 13. For more information about assigning these device numbers, see *Virtual Storage Manager Planning, Implementation, and Usage Guide*.

Associating VTD MVS Device Numbers and Esoterics

If you use esoteric substitution to allocate VTDs as described in "VSM Esoterics and Esoteric Substitution" on page 14, use the HCD facility to associate each esoteric name with the MVS device numbers for the VTDs that you have chosen for that esoteric.

Assigning MVS Device Numbers to RTDs

Use the HCD facility to assign MVS device numbers to these RTDs.

Tip – You must use the same unit addresses you determined for these transports for LIBGEN updates as described in "RTD Definitions" on page 21.

Setting the MVS Missing Interrupt Handler (MIH) Value

The VTSS's internal error recovery procedures requires the MVS missing-interrupt handler (MIH) value to be 20 minutes. You set this value by modifying the MIH parameter in SYS1.PARMLIB member IECIOS*xx*.

Note – Adjust applications running on your system that detect missing interrupts and that are independent of the system MIH setting to allow a five-minute MIH value.

Specifying the Region Size

StorageTek recommends that you run HSC/VTCS with a region size of **at least** 6 MB except if you are running utilities or commands that manipulate manifest files, in which case you need the maximum region size your system will allow.

CHAPTER 4

Installing VTCS Base

As you read through this chapter, you'll see that VTCS base installation is a series of standard SMP/E jobs, and therefore no big deal. There **are** some intricacies involved with VTCS installation, however, that you need to pay attention to, chiefly in the area of getting HSC/SMC in place and talking to VTCS. The 6.2 release now offers installation from CD-ROM as well as from tape, so if you want to use this media, it has its own subsection for getting the files off the CD and to a place where you can work with them. And one other item to consider...

Tip – Optionally, you can do all the tasks in "Reconfiguring HSC" on page 111 **before** you install VTCS, so if you want to arrange your work flow like that, go ahead and reconfigure HSC, then come back to this chapter and install the software.

And, as usual, there are a couple of "stop and think items," so **before** you install the software, complete the pre-installation tasks described in the following sections:

- "Ensuring that HSC and SMC Are Installed" on page 90
- "Reviewing Coexistence Requirements" on page 91
- "Verifying Base Installation Materials" on page 92

Next, install HSC, HSC maintenance, and VTCS 6.2.0 as described in the following sections:

- "Installing VTCS Base from CD-ROM" on page 96
- "Unloading the SMP/E JCL Library" on page 95
- "Allocating VTCS Target and Distribution Library Data Sets and Required DDDEF Entries" on page 94
- "Applying the VTCS 6.2.0 FMID" on page 99
- "Accepting the VTCS 6.2.0 FMID" on page 100
- "Adding SWSLINK to the Authorized Program List" on page 101
- "Modifying the HSC Startup Procedure to include the VTCS 6.2.0 LINKLIB" on page 102

Ensuring that HSC and SMC Are Installed

Install HSC, SMC and all HSC/SMC maintenance before you install VTCS. For more information, see *NCS Installation Guide.*

StorageTek **strongly recommends** upgrading HSC and the LibraryStation and MVS/CSC components (if you intend to connect MVS/CSC clients to VSM) to the current maintenance level before installing VTCS. Also note that StorageTek **requires** you to install SMC, which is required to perform allocation influencing and message interception on MVS.

Note –

- If you are running at an SMP/E level lower than that supplied with OS/390 Version 2.5, you *must* accept the HSC FMID (SOS6200 for HSC 6.2.0) and all HSC maintenance before installing VTCS 6.2.0.
- Installation from CD-ROM requires SMP/E version 3 Release 1 or higher to install VTCS from the USS platform. Users performing the VTCS installation must have access to USS with “make” and “write” permissions.

SMC Installation Considerations

Caution – If you are running in a JES3 environment, ensure that you create and install the SMC JES3 IATUX09 User Exit as described in *NCS Installation Guide.* This exit modification enables deferred mount processing, without which VTV mounts may fail.

Reviewing Coexistence Requirements

For more information, see "VTCS System Software Requirements" on page 5.

Note –

- The VSM Extended Format CDS **is required for VTCS 6.2.0. Also note that** after you convert the CDS to VSM Extended Format, you **cannot** run VTCS 4.0.0 or lower against the converted CDS. For more information, see "CDS VTCS Level" on page 32.
- Ensure that you install the Toleration PTFS (L1H1355 for VTCS 6.0 or L1H1356 for VTCS 6.1). These PTFs prevent a Manifest File created on a G Level CDS from being processed on a system with a lower level CDS.

Verifying Base Installation Materials

The installation materials consist of VTCS 6.2 Installation Base Media (tape or CD-ROM). The tape media is a single standard label tape with a volume serial number of SWS6200.

Note –

- Contact StorageTek Software Support for information about additional PTFs that might be required before installing the NCS product components. See the *Requesting Help from Software Support* guide for information about contacting StorageTek for technical support and for requesting changes to software products.
- If you are using HSC or MVS/CSC, the SMC software **must** be installed.
- StorageTek recommends that you use the MVS Program Binder when installing NCS products and maintenance. Failure to do so may result in link-editing errors.

VTCS Base Installation Media Contents

Base Media Contents

The Base Media contains the VTCS 6.2.0 FMID. TABLE 4-1 lists the files included on the VTCS 6.2.0 Base Tape.

TABLE 4-1 VTCS 6.2.0 Base Tape Contents

File	Data Set Name	Description
1	SMPMCS	SMP/E control statements
2	SWS6200.F1	SWS6200 JCLIN
3	SWS6200.F2	SWS6200 SAMPLIB members (automatically installed in the HSC SAMPLIB)
4	SWS6200.F3	SWS6200 MACLIB members (automatically installed in the HSC MACLIB)
5	SWS6200.F4	SWS6200 object modules

TABLE 4-2 lists the files included on the VTCS 6.2.0 Base CD-ROM.

TABLE 4-2 VTCS Release 6.2 Base CD Contents

File	Data Set Name	Description
1	Documents	The installation documentation for installing VTCS from a CD
2	VTCS62.gimzip	Compressed file containing the complete VTCS product minus sample JCL.
3	VTCS62.pax	Compressed file containing the VTCS product including sample JCL.
4	Samples	Unix version of VTCS samples (without CR/LF)
5	Samples.win	Windows version of VTCS samples (with CR/LF)
6	Start Here.xml	XML starting point to the documents folder

Note – The VTCS 6.2.0 installation automatically installs the following VTCS members in the HSC SAMPLIB:

SWSJCRDB

Sample CONFIG utility JCL

SWSJMVCR

Sample MVCRPT utility JCL

SWSJVTVR

Sample VTVRPT utility JCL

VTCS FMIDs

The VTCS 6.2.0 software is packaged in standard SMP/E format. The VTCS 6.2.0 installation tape includes the following VTCS FMID:

SWS6200

The SWS6200 function contains the VTCS load modules for VTCS 6.2.0 running with HSC 6.2.0.

Note – The SWS6200 FMID is a subsidiary HSC 6.2.0 FMID (SOS6200 for HSC 6.2.0) and you must apply the SWS6200 FMID to the same SMP/E zone as HSC. Products from other vendors should **not** be installed in this SMP/E CSI.

Allocating VTCS Target and Distribution Library Data Sets and Required DDDEF Entries

If you **did not** previously allocate these data sets during NCS installation, you **must** allocate VTCS target and distribution data sets and define the appropriate DDDEF entries in the SMP/E CSI, using the sample batch job provided in member NCSDDEF of your SMP/E JCL library.

Note – At this point, depending on the Base Media:

- If installing from tape, go to "Unloading the SMP/E JCL Library" on page 95.
- If installing from CD-ROM, go to "Installing VTCS Base from CD-ROM" on page 96

Installing the VTCS Base from Tape

t Unloading the SMP/E JCL Library

To unload the files from tape, use the SWS6200.F1 JCLIN file described in TABLE 4-1 on page 93 (JCL shown in FIGURE 4-1), then go to “Receiving the VTCS 6.2.0 FMID from Tape”.

```
//jobname JOB your jobcard parameters
//UNLOAD  EXEC PGM=IEBCOPY
//INDD   DD DSN=SWS6200.F1,DISP=SHR,
//       UNIT=tape-unit,VOL=SER=SWS6200,LABEL=(2,SL)
//OUTDD  DD DSN=your.smpe.jcllib,DISP=(NEW,CATLG),
//       UNIT=SYSALLDA,
//       SPACE=(TRK,(5,1,4)),
//       DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120)
//SYSPRINT DD SYSOUT=*
//SYSIN  DD *
 C I=INDD,O=OUTDD
 E M=SWS6200
/*
```

FIGURE 4-1 JCL to Unload the VTCS SMP/E JCL Library from Tape

t Receiving the VTCS 6.2.0 FMID from Tape

Modify SAMPLIB member NCSRECV per the instructions in the prologue and run it to receive the VTCS 6.2.0 FMID. The receive must run with return code zero; otherwise, contact StorageTek Software Support.

Installing VTCS Base from CD-ROM

t Unloading the VTCS Base Installation CD

The following sections tell how to unload the VTCS base functions from the Base CD-ROM.

Note – After unloading the VTCS base functions, to receive the VTCS 6.2 FMID go to "Receiving the VTCS 6.2.0 FMID from CD-ROM" on page 74

---

t FTPing VTCS62.pax to USS

The VTCS62.pax file contains the VTCS Base and sample JCL.

To FTP VTCS62.pax to USS:

1. **Create a USS directory to receive the VTCS62.pax file:**

   a. Establish network connectivity between your desktop PC and a MVS host and USS.

   b. Logon to USS by entering **OMVS** from the TSO READY prompt or enter **TSO OMVS** from an ISPF command line.

   c. Use the **mkdir** command to create a new directory. For example, the **mkdir SMPNTS** command creates a new directory named **SMPNTS**. Use the **pwd** command to show the complete USS path to **SMPNT**S.

   Note this path information, which you need in the SMP/E RECEIVE batch job NTSVTCS.

2. **Insert the VTCS 6.2 installation CD into the CD-ROM drive.**

3. **Open a DOS window and cd to your CD-ROM.**

4. **Transfer the VTCS62.pax file, in binary, from the CD to the USS directory using the following FTP commands:**

   ftp *mvshost*
   user
   password
   cd /uss/*userid/SMPNTS*
   binary
   put VTCS62.pax
   quit

5. **Watch for FTP messages to ensure the successful transfer of VTCS62.pax to the new directory.**

   You can also logon on to USS, change directory (**cd**) to the SMPNTS directory and then enter the **ls** command from within the SMPNTS directory. The VTCS62.pax file should be listed.

t Unpacking the VTCS62.pax File

This section tells how to unpack the VTCS62.pax file.

To unpack the VTCS62.pax file:

1. **Log on to USS by entering** TSO OMVS **from an ISPF command line or** OMVS **from the TSO READY prompt.**

   If you are still logged on to USS and within the SMPNTS directory, go to Step 3. Otherwise, continue with Step 2.

2. **Change directory (**cd**) to the SMPNTS directory.**

3. **To unpack the VTCS62.pax file:**

   pax -rv <VTCS62.pax

   This file will unpack in place and will create the files and directories shown in TABLE 4-3 and

TABLE 4-3 Contents of SMPNTS after unpacking the VTCS62.pax file

File	Data Set Name	Description
1	LOADSAMP.xmit	NCS Samples formatted for input into the TSO Receive
2	VTCS62.gimzip	Directory Input to the SMP/E Receive process
3	VTCS62.pax	Compressed file FTPed from the install CD
4	Samples	CD installation JCL

TABLE 4-4 Contents of the VTCS62.gimzip directory

File	Data Set Name	Description
1	GIMPAF.XML	Product attribute file
2	GIMPAF.XSL	Style sheet
3	SMPHOLD	Directory containing SMP/E HOLD data
4	SMPPTFIN	Directory containing SMP/E PTFIN file for each product
5	SMPRELF	Directory containing SMP/E REL files for each product

   TABLE 4-4.

4. **If desired, you can now delete the VTCS62.pax file to reclaim space:**

   rm VTCS62.pax

t TSO RECEIVEing the LOADSAMP.xmit file

After you unpack the VTCS62.pax file, copy the LOADSAMP.xmit file to an MVS PDS data set using the following JCL:

```
//Your jobcard
//****************************************************************************
//* Turn caps off for this member. Enter CAPS OFF on the command line.       *
//*                                                                          *
//* Use this JCL member to unload VTCS sample JCL members from USS.          *
//*                                                                          *
//*** You must first FTP the VTCS62.pax file to USS and unpack the file   ****
//*                                                                          *
//****************************************************************************
//ALLOCJCL EXEC PGM=IEFBR14,REGION=4800K
//DD1   DD DSN=hlq.VTCS62.JCLSAMP,
//         DISP=(NEW,CATLG),UNIT=XXXX,VOL=SER=XXXXXX,
//         DCB=(LRECL=80,BLKSIZE=27920,RECFM=FB),
//         SPACE=(TRK,(9,4,10))
/*
//UNLOAD   EXEC PGM=IKJEFT01,COND=(0,NE)
//SYSPRINT DD SYSOUT=*
//SYSTSPRT DD SYSOUT=*
//SYSTSIN  DD DATA
 ALLOCATE DD(LOADDD) PATHOPTS(ORDONLY) FILEDATA(BINARY) +
              PATH('/uss/userid/SMPNTS/LOADSAMP.xmit')

 RECEIVE INDD(LOADDD)
          DSNAME('hlq.VTCS62.JCLSAMP') SHR
```

FIGURE 4-2 JCL to Copy LOADSAMP.xmit to a PDS

t SMP/E Receiving the VTCS Functions from the VTCS Installation CD

Use the NTSVTCS sample member to SMP/E RECEIVE the VTCS functions from Unix Systems Services (USS).

Follow the instructions in the prologue of the NTSVTCS sample member and submit the batch job to receive the VTCS FMID.

The return code **must** be zero (0) for all steps executed in this job. If you receive a different return code, contact StorageTek Software Support.

Applying the VTCS 6.2.0 FMID

Modify SAMPLIB member NCSAPPLY per the instructions in the prologue and run it to apply the VTCS 6.2.0 FMID. The apply must run with return code less than or equal to 4; otherwise, contact StorageTek Software Support.

Note – Because the VTCS FMID is a subsidiary HSC FMID, SMPE will also apply any VTCS PTFs called out by conditional COREQS in the HSC PTFs already applied to your system. If any of these PTFs have HOLDDATA, you will receive a GIM35965I message for each, and the APPLY will fail. This is not an error condition. If you encounter this condition, please review the individual PTF cover letters, note any additional action(s) to be taken, and repeat the APPLY step with the following parameters:

```
APPLY S(SWS6200) GROUPEXTEND
                                BYPASS(HOLDSYSTEM) .
```

Accepting the VTCS 6.2.0 FMID

Modify SAMPLIB member NCSACCPT per the instructions in the prologue and run it to accept the VTCS 6.2.0 FMID. The accept must run with return code less than or equal to 4; otherwise, contact StorageTek Software Support.

Note – If you are running at an SMP/E level lower than that supplied with OS/390 Version 2.5, you *must* accept the HSC FMID and all HSC maintenance before installing VTCS 6.2.0. For more information, see "Ensuring that HSC and SMC Are Installed" on page 90.

Also note that because the VTCS FMID is a subsidiary HSC FMID, SMPE will also accept any VTCS PTFs called out by conditional COREQS in the HSC PTFs already accepted on your system. If any of these PTFs have HOLDDATA, you will receive a GIM35965I message for each, and the ACCEPT will fail. This is not an error condition. If you encounter this condition, please review the individual PTF cover letters, note any additional action(s) to be taken, and repeat the ACCEPT step with the following parameters:

```
ACCEPT S(SWS6200) GROUPEXTEND
            BYPASS(HOLDSYSTEM) .
```

Adding SWSLINK to the Authorized Program List

VTCS must run as an authorized program, which you do by adding the VTCS Link Library (SWSLINK) to the authorized program list on your system in one of two ways:

- Dynamically
- "Using IEAAPFxx to APF Authorize the SWSLINK"
- "Using PROGxx to APF Authorize SWSLINK"

Using IEAAPF*xx* to APF Authorize the SWSLINK

To use the IEAAPF*xx* member of SYS1.PARMLIB to authorize the SWSLINK, add the following entry to that list with your HLQ and volser:

```
your.SWSLINK volser
```

Note – If SWSLINK resides on an SMS-managed volume, you do not need to specify a volume in the IEAAPF*xx* member as follows:

```
your.SWSLINK
```

Using PROG*xx* to APF Authorize SWSLINK

To use the PROG*xx* member of SYS1.PARMLIB to authorize SWSLINK, add the following entries to that list with your HLQ and volser:

```
APF ADD
              DSNAME(your.SLSLINK)
              VOLUME(volser)
```

Note – If SWSLINK resides on an SMS-managed volume, you do not need to specify a volume in the PROG*xx* member as follows:

```
APF ADD
              DSNAME(your.SLSLINK)
              VOLUME SMS
```

Modifying the HSC Startup Procedure to include the VTCS 6.2.0 LINKLIB

Use the example JCL in FIGURE 4-1 as an example of how to modify the HSC startup procedure to start VTCS. Include the VTCS 6.2.0 LINKLIB (SWSLINK) in the STEPLIB **before** the HSC LINKLIB (SLSLINK).

```
//SLSO      PROC PROG=SLSBINIT
//IEFPROC   EXEC PGM=&PROG,TIME=1440,DPRTY=(7,5),
//          PARM='SSYS(SLSO) E(E086) F(23) M(00)'
//STEPLIB   DD DSN=hlq.SWSLINK,DISP=SHR
//          DD DSN=hlq.SLSLINK,DISP=SHR
```

FIGURE 4-1 JCL Example: Modifying the HSC started task to include the SWSLINK library

Installing VTCS 6.2 Maintenance

After installing VTCS 6.2 Base, you **must** install the current maintenance from the VTCS 6.2 Service Tape. For more information, see “Installing VTCS Maintenance” on page 105.

CHAPTER 5

Installing VTCS Maintenance

After installing VTCS 6.2 Base as described in "Installing VTCS Base" on page 89, you **must** install the current maintenance from the VTCS 6.2 Service Tape as described in the following sections:

- "Verifying Service Media" on page 105
- "Unloading the VTCS Service Tape" on page 107
- "Unloading the VTCS Service CD-ROM" on page 107
- "Receiving the VTCS 6.2 Maintenance" on page 108
- "Applying the VTCS 6.2 Maintenance" on page 109
- "Accepting the VTCS 6.2 Maintenance" on page 109

Verifying Service Media

The VTCS 6.2 Service Media (tape or CD-ROM) contains VTCS PTFs since the Base Media was created.

Note –

- Contact StorageTek Software Support as described in "Customer Support" on page xiv for information about additional PTFs that might be required before installing VTCS maintenance.
- StorageTek recommends that you use the MVS Program Binder when installing VTCS products and maintenance. Failure to do so may result in link-editing errors.

VTCS Service Media Contents

TABLE 5-1 lists the files included on the VTCS 6.2.0 service tape. The rest of the tables list files included on the service CD-ROM.

TABLE 5-1 VTCS 6.2.0 Service Tape Contents

File	Data Set Name	Description
1	SYSMODS	SYSMODS
2	CVR	PTF cover letters and JCL samples
3	SMM	Summary data
4	HOLDDATA	SMP/E HOLDDATA

TABLE 5-2 lists the files included on the VTCS 6.2.0 service CD-ROM.

TABLE 5-2 VTCS 6.2.0 Service CD Contents

File	Data Set Name	Description
1	code	Cumulative service
2	READMEs	Documentation

TABLE 5-3 lists the files included in the README directory of the VTCS 6.2.0 service CD-ROM.

TABLE 5-3 VTCS 6.2.0 Service CD Contents - README Directory

File	Data Set Name	Description
1	badge	HTML element
2	CD_ROM. html	Documentation
3	logoab8.gif	HTML element
4	Corrective Service Installation Guide.pdf	Corrective service documentation in PDF format

TABLE 5-4 lists the files included in the code directory of the VTCS 6.2.0 service CD-ROM.

TABLE 5-4 VTCS 6.2.0 Service CD Contents - code Directory

File	Data Set Name	Description
1	vtcs62.cvr	Cover letters
2	vtcs62.hdd	HOLDDATA
3	vtcs62.ptf	Cumulative Service PTFs
4	vtcs62.smm	Summary

Note – At this point, depending on the Service Media:

- If installing from tape, go to “Unloading the VTCS Service Tape” on page 107.
- If installing from CD-ROM, go to “Unloading the VTCS Service Tape” on page 107

Unloading the VTCS Service Tape

Sample JCL members for unloading VTCS maintenance from tape were unloaded from the VTCS base tape or CD-ROM during the VTCS installation process. See "VTCS Base Installation Media Contents" on page 93 for more information.

Unloading the VTCS Service CD-ROM

To unload COVER letters, HOLDDATA, PTFs, and SUMMARY data from your CD-ROM drive to your MVS host:

1. Pre-allocate the following FTP receiving data sets. Change "*hlq*" to your high-level-qualifier; change "*vr*" to 6.2.

   Data Set Name . . . :*hlq*.VTCS*vr*.HDD - HOLDDATA

   // RECFM=FB,LRECL=80,SPACE=(27920,(30,30))

   Data Set Name . . . :*hlq*.VTCS*vr*.PTF - PTFs

   // RECFM=FB,LRECL=80,SPACE=(27920,(13000,300))

   Data Set Name . . . :*hlq*.VTCS*vr*.CVR - Cover Letters

   // RECFM=FB,LRECL=80,SPACE=(27920,(30,30))

   Data Set Name . . . :*hlq*.VTCS*vr*.SMM - Summary

   // RECFM=FB,LRECL=80,SPACE=(27920,(30,30))

2. Insert the Service CD into your CD-ROM drive.
3. Open a DOS window and cd to your CD-ROM drive.
4. Enter the following commands:

```
FTP mvshost
User
Password
binary
mput vtcs62.cvr
mput vtcs62.hdd
mput vtcs62.ptf
mput vtcs62.smm
quit
```

Running this sequence of mput commands copies the following CD files:

- vtcs62.cvr
- vtcs62.hdd
- vtcs62.ptf
- vtcs62.smm

...to the following data sets on your MVS system:

- *hlq*.VTCS62.CVR
- *hlq*.VTCS62.HDD
- *hlq*.VTCS62.PTF
- *hlq*.VTCS62.SMM

Receiving the VTCS 6.2 Maintenance

To receive the VTCS 6.2 maintenance, modify and run one of the following SAMPLIB members per the instructions in the prologue:

- Use MAINTRCF to SMP/E RECEIVE maintenance by specific FMID.
- Use MAINTRCS to SMP/E RECEIVE maintenance by specific SYSMOD.

The receive must run with return code zero; otherwise, contact StorageTek Software Support.

Note –

- If you are performing the SMP/E RECEIVE from CD-ROM, modify the MAINTRCF and MAINTRCS members as follows:
  - Modify the SMPPTFIN DD statement to point to the *hlq*.VTCS62.PTF data set.
  - Modify the SMPHOLD DD statement to point to the *hlq*.VTCS62.HDD data set.
- When installing VTCS, you **must** SMP/E receive all maintenance from the corrective service tape or CD before performing an SMP/E APPLY and ACCEPT for the HSC and VTCS base functions because of the ifreq and coreq relationship between HSC and VTCS.

Applying the VTCS 6.2 Maintenance

To apply the VTCS 6.2 maintenance, modify and run one of the following SAMPLIB members per the instructions in the prologue:

- Use MAINTAPF to SMP/E APPLY maintenance by specific FMID.
- Use MAINTAPS to SMP/E APPLY maintenance by specific SYSMOD.

The apply must run with return code zero; otherwise, contact StorageTek Software Support.

Note – You can modify the sample members to do an APPLY CHECK, then modify again to do an APPLY.

Accepting the VTCS 6.2 Maintenance

To accept the VTCS 6.2 maintenance, modify and run one of the following SAMPLIB members per the instructions in the prologue:

- Use MAINTACF to SMP/E ACCEPT maintenance by specific FMID.
- Use MAINTACS to SMP/E ACCEPT maintenance by specific SYSMOD.

The accept must run with return code zero; otherwise, contact StorageTek Software Support.

Note – You can modify the sample members to do an ACCEPT CHECK, then modify again to an ACCEPT.

CHAPTER 6

Reconfiguring HSC

Okay, the NCS/VTCS software is installed, and now things get interesting. Because, as you've probably guessed, **before** you configure VSM, you must do some or all of the HSC reconfiguration tasks described in the following sections.

The "some or all" you already determined by reading through the notes in Table 2. on page 1. **That is,** if you are upgrading from a previous release of VTCS, you may not need to do all the tasks in this chapter. For example, if you are not adding RTDs to your configuration, you do not need to update the HSC LIBGEN.

In addition, as noted in Chapter 4, "Installing VTCS Base"...

Tip – ...you can do all the tasks in "Reconfiguring HSC" **before** you install VTCS. Most of the tasks in this Chapter require you to specify VSM system values that you recorded in TABLE A-1 on page 165.

Your task list for reconfiguring NCS is as follows:

- "Creating or Updating the HSC LIBGEN" on page 112
- "Verifying the LIBGEN" on page 113
- "Formatting the New CDS" on page 114
- "Creating VOLATTR Statements for VTVs" on page 115
- "Defining MVCs to HSC" on page 115

Creating or Updating the HSC LIBGEN

If your system's RTDs are new transports, you must update the HSC LIBGEN by adding a SLIDRIVS macro to define the device addresses you determined in "RTD Definitions" on page 21. Similarly, if you have made other hardware changes (for example, adding or removing LSMs), you must update the related LIBGEN macros as described in Step 2, below. If you are converting the CDS to VSM Extended Format as described in "CDS VTCS Level" on page 32, you must create a new CDS. For more information, see *HSC System Programmer's Guide for MVS*.

To update the HSC LIBGEN to define new transports that are RTDs:

1. **Run the HSC Database Decompile (LIBGEN) Utility to create LIBGEN macro statements from your existing CDS.**

   Do *not* edit the original LIBGEN, because if the SET Utility was used to change the library configuration stored in the CDS, the original LIBGEN no longer matches the CDS. For more information about the Database Decompile Utility, see *HSC System Programmer's Guide for MVS*.

2. **After you run the HSC Database Decompile Utility, add a SLIDRIVS macro to define the RTD device addresses.**

   You may also need to update related LIBGEN macros, such as the SLIACS, SLILSM and SLIDLIST macros. For more information about the LIBGEN macros, see *HSC Configuration Guide for MVS.*

Note – You can specify that Nearline transports can only be used as RTDs on the SLIACS macro as shown below:

```
SLIACS ACSDRV=(esoteric0,...,esoteric15),LSM=(....)
```

As shown in this example, in the HSC SLIACS macro:

- The ACSDRV parameter specifies the esoteric name of each host that refers to the transports attached to this ACS. A comma is a placeholder for any esoteric name not specified.

Also note that you can use the HSC SET ACS utility to you specify that Nearline transports can only be used as RTDs as follows:

- The ACSDRV parameter specifies the esoteric name of the host that refers to the transports attached to this ACS. A () specifies that the esoteric for the specified host and ACS is set to blank.

Finally, also note that if you want to have MVS/CSC clients connected to an ACS whose RTDs are dedicated to VSM:

- In the LIBDEV startup parameter, for an ACS that is only attached to VSM, the position ACS esoteric name must be blank.
- Omit the LIBUNIT startup parameter for an ACS that is only attached to VSM.
- Omit the UNITMAP startup parameter for an ACS that is only attached to VSM if *both* of following are true:
  - Clients use the same MVS device numbers defined for this ACS's RTDs.

- Your configuration has no cartridge tape UCB defined with an MVS device number that matches a device number defined for this ACS's RTDs.

If either or both of these statements **is not** true, then you must create both a LIBUNIT and a UNITMAP statement to map the drives to the client addresses.

Tip – As an alternative to MVS/CSC and LibraryStation, you can use the SMC 6.2 client/server capabilities. For more information, see *SMC Configuration and Administration Guide*.

---

3. **After you update the LIBGEN macros, reassemble and link–edit the LIBGEN file.**

   For more information, see "LIBGEN Process Verification" in *HSC Configuration Guide for MVS.*

Verifying the LIBGEN

After you assemble and link edit the LIBGEN file, run the SLIVERFY program to verify the LIBGEN. For more information, see "Verifying the Library Generation" *HSC Configuration Guide.*

Caution – Before you run SLIVERFY, if your system's RTDs are new transports that you will share with MVS, you must install them and define their MVS unit addresses via the HCD facility as described in "Configuring MVS Device Numbers and Esoterics" on page 87.

Formatting the New CDS

In "HSC CDS DASD Space" on page 49, you determined the size of the CDS to support your VSM system. If you require a larger CDS, you must format a new CDS to this size by using the HSC SLICREAT macro. For more information on the HSC SLICREAT macro, see *HSC Configuration Guide.*

Note that:

- Before converting the CDS to VSM Extended Format as described in "Configuring VTCS" on page 117, you need to allocate a new data set for the VSM Extended Format CDS.
- If you change the CDS data set name, make sure to update the name in the HSC started task and in any other started tasks or batch jobs (such as ExPR and ExLM) that reference this data set.
- As described in "CDS Locations" on page 6VSM **does not** support copies of the CDS at multiple sites.
- For more information on using SLICREAT to format the new CDS, see *HSC System Programmer's Guide*. Typically, you run SLICREAT with the highest level of HSC that you will run after converting the CDS format.

Creating VOLATTR Statements for VTVs

As described in "VTV Definitions" on page 19, to define VTVs, you must create:

- CONFIG VTVVOL statements as shown in Step 5 on page 121.
- HSC VOLATTR statements to define VTVs to HSC, for example:

```
VOLATTR SERIAL (905000-999999) MEDIA(VIRTUAL)
VOLATTR SERIAL (C00000-C25000) MEDIA(VIRTUAL)
VOLATTR SERIAL (RMM000-RMM020) MEDIA(VIRTUAL)
```

Defining MVCs to HSC

As described in "MVC Definitions" on page 22, to define MVCs, you must create:

- CONFIG MVCVOL statements as shown in Step 5 on page 121.
- HSC VOLATTR statements to define MVCs to HSC. For example, to define MVCs **N25980-N25989** as T9840A/T9840B volumes and MVCs **N3500-N3599** as T9840C volumes, create the following VOLATTR statements:

```
VOLATTR SERIAL(N25980-N25989) MEDIA(STK1R) RECTECH(STK1RAB)
VOLATTR SERIAL(N3500-N3599) MEDIA(STK1R) RECTECH(STK1RC)
```

- HSC MVCPool statements, which specify the pool of MVCs available for migration and consolidation requests. For example, to define a single MVC Pool for the above volsers:

1. **Create the following MVCPool statements:**

```
MVCPOOL VOLSER(N25980-N25989)
MVCPOOL VOLSER(N3500-N3599)
```

2. **Run the VT MVCDEF command to activate the updated data set, for example:**

```
VT MVCDEF DSN(VSM.MVCPOOL)
```

CHAPTER 7

Configuring VTCS

All the warming up is done, and it's now time for the main part of the show, which is configuring VTCS, which you do using the VTCS CONFIG utility...so now would be a good time to open up your *ELS Command, Control Statement, and Utility Reference*.

Much of the CONFIG deck is devoted to what you would expect...defining to VTCS the essential virtual/hardware parts of the VSM solution...VTSSs, VTVs, VTDs, RTDs, and MVCs. What you **also** have the option of doing is setting non-default operating values, which we discussed in Chapter 2, "Planning VTCS Operating Policies"...and if you did your homework, you have the values recorded in TABLE A-1 on page 165.

This chapter has two major sections:

- "Building a Simple CONFIG Deck" on page 119 is basically a procedure where we build an example plain-vanilla CONFIG deck step by step. To lead you through this, we'll do it in ordered fashion, building the CONFIG deck as we go, as described in TABLE 7-1.
- "Special Cases: A Gallery of Advanced Uses of CONFIG" on page 123 is...well, what *he* said. There are variations of the CONFIG deck beyond the plain vanilla model, such as the one where you disable a host from initiating migrations, and this is the place to go if you're looking for that kind of information.

TABLE 7-1 VTCS Configuration Tasks

Step	Notes	Planning Information
Step 1 on page 119	Are you doing an upgrade install? Then you want to run DECOM to get a true picture of your current configuration. You can then update the DECOM listing and resubmit it to CONFIG to update your configuration. Typically, you run DECOM with the highest level of the VTCS (6.0 or 6.1) that supports the CDS 'from' level. First time install? Skip on down to Step 2 on page 119.	
Step 2 on page 119	The actual CDS level specification is pretty easy, but deciding which level takes some Deep Thought, so please study carefully the planning section before you begin coding...	..."CDS VTCS Level" on page 32
Step 3 on page 120	This is where you specify global operating policies, such as the number of free MVCs...if you want to. Remember, you can always take the defaults, and tune things up later...	- "Storing VTCS Locks in a Coupling Facility (Optional)" on page 35 - "Planning VTCS Operating Policies" on page 51

TABLE 7-1 VTCS Configuration Tasks

Step	Notes	Planning Information
Step 4 on page 120	Moving right along to the MVC reclamation policies...	"MVC Policies" on page 72
Step 5 on page 121	...followed by VTV and MVC volsers...	■ "VTV Definitions" on page 19 ■ "MVC Definitions" on page 22
Step 6 on page 122	...and, to complete the hardware definitions, CONFIG statements for the system's VTSSs, the RTDs attached to the VTSSs, and the VTDs in each VTSS.	■ "VTSS Names" on page 12 ■ "RTD Definitions" on page 21 ■ "VTD Unit Addresses" on page 13 ■ "VSM Esoterics and Esoteric Substitution" on page 14

Building a Simple CONFIG Deck

t To build a simple CONFIG deck:

1. **Run the VTCS DECOM Utility...**

   ...if you're doing an upgrade install. Otherwise, skip to Step 2.

   FIGURE 7-1 shows example JCL to run the DECOM utility with output to flat file CFG22202.

```
//DECOM EXEC PGM=SWSADMIN,PARM='MIXED'
//STEPLIB DD DSN=hlq.SLSLINK,DISP=SHR
//SLSCNTL DD DSN=FEDB.VSMLMULT.DBASEPRM,DISP=SHR
//SLSCNTL2 DD DSN=FEDB.VSMLMULT.DBASESEC,DISP=SHR
//SLSSTBY DD DSN=FEDB.VSMLMULT.DBASETBY,DISP=SHR
//CFG22202 DD DSN=FEDB.VSMLMULT.CFG22202,DISP=SHR
//SLSPRINT DD   SYSOUT=*
//SLSIN DD   *
 DECOM FLATDD(CFG22202)
```

FIGURE 7-1 Example JCL for the DECOM utility

2. **Start the CONFIG deck by specifying the CDS Level.**

   FIGURE 7-2 shows CONFIG JCL example to specify CDS level G.

```
//CREATECF EXEC PGM=SWSADMIN,PARM='MIXED'
//STEPLIB DD DSN=hlq.SLSLINK,DISP=SHR
//SLSCNTL DD DSN=FEDB.VSMLMULT.DBASEPRM,DISP=SHR
//SLSCNTL2 DD DSN=FEDB.VSMLMULT.DBASESEC,DISP=SHR
//SLSSTBY DD DSN=FEDB.VSMLMULT.DBASETBY,DISP=SHR
//CFG22202 DD DSN=FEDB.VSMLMULT.CFG22202,DISP=SHR
//SLSPRINT DD   SYSOUT=*
//SLSIN DD   *
 CONFIG CDSLEVEL(V62ABOVE)
```

FIGURE 7-2 CONFIG Example: CDSLEVEL(V62ABOVE) to Specify Level G

Note – In FIGURE 7-2, per VTCS NCO support, we do not have to specify RESET. RESET is **not** required when going from F to G level, but all hosts accessing the CDS must be at VTCS/NCS 6.2 during the CDS level change. See *ELS Command, Control Statement, and Utility Reference* for details about when RESET is required.

---

3. **Specify global values.**

FIGURE 7-3 shows our example CONFIG deck all spruced up with global values. **Note that** we've specified a Coupling Facility Structure to hold VTCS Lock Data:

- The Structure must be predefined as shown in "Defining the Coupling Facility Structure to MVS" on page 36.
- We now have to specify RESET and all hosts must be down to implement or remove a VTCS Lock Structure within a Coupling Facility.

```
//CREATECF EXEC PGM=SWSADMIN,PARM='MIXED'
//STEPLIB DD DSN=hlq.SLSLINK,DISP=SHR
//SLSCNTL DD DSN=FEDB.VSMLMULT.DBASEPRM,DISP=SHR
//SLSCNTL2 DD DSN=FEDB.VSMLMULT.DBASESEC,DISP=SHR
//SLSSTBY DD DSN=FEDB.VSMLMULT.DBASETBY,DISP=SHR
//CFG22202 DD DSN=FEDB.VSMLMULT.CFG22202,DISP=SHR
//SLSPRINT DD   SYSOUT=*
//SLSINDD   *
 CONFIG RESET CDSLEVEL(V62ABOVE)
 GLOBAL MAXVTV=65000 MVCFREE=60 VTVATTR=SCRATCH RECALWER=YES
 LOCKSTR=STK_VTCS_LOCKS VTVPAGE=LARGE
```

FIGURE 7-3 CONFIG example: Specifying Global Values

Tip – If a Lock Structure is already defined to VTCS, you can use DECOM, Display CONFIG, and Display LOCKS to display information about the Lock Structure.

---

4. **Specify reclamation policy values.**

FIGURE 7-4 shows our example CONFIG deck, now sporting reclamation policy values.

```
//CREATECF EXEC PGM=SWSADMIN,PARM='MIXED'
//STEPLIB DD DSN=hlq.SLSLINK,DISP=SHR
//SLSCNTL DD DSN=FEDB.VSMLMULT.DBASEPRM,DISP=SHR
//SLSCNTL2 DD DSN=FEDB.VSMLMULT.DBASESEC,DISP=SHR
//SLSSTBY DD DSN=FEDB.VSMLMULT.DBASETBY,DISP=SHR
//CFG22202 DD DSN=FEDB.VSMLMULT.CFG22202,DISP=SHR
//SLSPRINT DD   SYSOUT=*
//SLSIN DD   *
 CONFIG RESET CDSLEVEL(V62ABOVE)
 GLOBAL MAXVTV=65000 MVCFREE=60 VTVATTR=SCRATCH RECALWER=YES
 LOCKSTR=STK_VTCS_LOCKS VTVPAGE=LARGE
 RECLAIM THRESHLD=70 MAXMVC=30 START=40 CONMVC=5
```

FIGURE 7-4 CONFIG example: Specifying Reclamation Values

5. **Specify VTV and MVC volsers.**

FIGURE 7-5 shows our example CONFIG deck with the addition of VTV and MVC volsers. Note that the VTVs are defined in scratch status so we won't have to explicitly scratch them with HSC SLUADMIN.

```
//CREATECF EXEC PGM=SWSADMIN,PARM='MIXED'
//STEPLIB DD DSN=hlq.SLSLINK,DISP=SHR
//SLSCNTL DD DSN=FEDB.VSMLMULT.DBASEPRM,DISP=SHR
//SLSCNTL2 DD DSN=FEDB.VSMLMULT.DBASESEC,DISP=SHR
//SLSSTBY DD DSN=FEDB.VSMLMULT.DBASETBY,DISP=SHR
//CFG22202 DD DSN=FEDB.VSMLMULT.CFG22202,DISP=SHR
//SLSPRINT DD   SYSOUT=*
//SLSIN DD   *
 CONFIG RESET CDSLEVEL(V62ABOVE)
 GLOBAL MAXVTV=65000 MVCFREE=60 VTVATTR=SCRATCH RECALWER=YES
 LOCKSTR=STK_VTCS_LOCKS VTVPAGE=LARGE
 RECLAIM THRESHLD=70 MAXMVC=30 START=40 CONMVC=5
 VTVVOL LOW=905000 HIGH=999999 SCRATCH
 VTVVOL LOW=C00000 HIGH=C25000 SCRATCH
 VTVVOL LOW=RMM000 HIGH=RMM020 SCRATCH
 MVCVOL LOW=N25980 HIGH=N25989
 MVCVOL LOW=N35000 HIGH=N3599
```

FIGURE 7-5 CONFIG example: Specifying VTV and MVC Volsers

6. **Define VTSSs, RTDs, and VTDs.**

FIGURE 7-6 shows our example CONFIG deck that next defines our VTSSs, the RTDs attached to each VTSS (both VSM4s), and the VTDs in each VTSS. **Note that** the RTD and VTD definitions immediately follow the VTSS statement.

```
//CREATECF EXEC PGM=SWSADMIN,PARM='MIXED'
//STEPLIB DD DSN=hlq.SLSLINK,DISP=SHR
//SLSCNTL DD DSN=FEDB.VSMLMULT.DBASEPRM,DISP=SHR
//SLSCNTL2 DD DSN=FEDB.VSMLMULT.DBASESEC,DISP=SHR
//SLSSTBY DD DSN=FEDB.VSMLMULT.DBASETBY,DISP=SHR
//CFG22202 DD DSN=FEDB.VSMLMULT.CFG22202,DISP=SHR
//SLSPRINT DD   SYSOUT=*
//SLSIN DD   *
 CONFIG RESET CDSLEVEL(V62ABOVE)
 GLOBAL MAXVTV=65000 MVCFREE=60 VTVATTR=SCRATCH RECALWER=YES
 LOCKSTR=STK_VTCS_LOCKS VTVPAGE=LARGE
 RECLAIM THRESHLD=70 MAXMVC=30 START=40 CONMVC=5
 VTVVOL LOW=905000 HIGH=999999 SCRATCH
 VTVVOL LOW=C00000 HIGH=C25000 SCRATCH
 VTVVOL LOW=RMM000 HIGH=RMM020 SCRATCH
 MVCVOL LOW=N25980 HIGH=N25989
 MVCVOL LOW=N35000 HIGH=N3599
 VTSS NAME=VSM41 LOW=70 HIGH=80 MAXMIG=8 MINMIG=4 RETAIN=5
 RTD  NAME=PR11A00 DEVNO=1A00 CHANIF=0C
 RTD  NAME=PR11A01 DEVNO=1A01 CHANIF=0D
 RTD  NAME=PR11A02 DEVNO=1A02 CHANIF=0K
 RTD  NAME=PR11A03 DEVNO=1A03 CHANIF=0L
 RTD  NAME=PR12A08 DEVNO=2A08 CHANIF=1C
 RTD  NAME=PR12A09 DEVNO=2A09 CHANIF=1D
 RTD  NAME=PR12A0A DEVNO=2A0A CHANIF=1K
 RTD  NAME=PR12A0B DEVNO=2A0B CHANIF=1L
 VTD LOW=9900 HIGH=99FF
 VTSS NAME=VSM42 LOW=70 HIGH=80 MAXMIG=8 MINMIG=4 RETAIN=5
 RTD  NAME=PR23A00 DEVNO=3A00 CHANIF=0C
 RTD  NAME=PR23A01 DEVNO=3A01 CHANIF=0D
 RTD  NAME=PR23A02 DEVNO=3A02 CHANIF=0K
 RTD  NAME=PR23A03 DEVNO=3A03 CHANIF=0L
 RTD  NAME=PR24A08 DEVNO=4A08 CHANIF=1C
 RTD  NAME=PR24A09 DEVNO=4A09 CHANIF=1D
 RTD  NAME=PR24A0A DEVNO=4A0A CHANIF=1K
 RTD  NAME=PR24A0B DEVNO=4A0B CHANIF=1L
 VTD LOW=9900 HIGH=99FF
```

FIGURE 7-6 CONFIG example: Defining VTSSs

Special Cases: A Gallery of Advanced Uses of CONFIG

Welcome to Star Wars, the movie...in this section, you’re going to learn everything you ever wanted to know about wringing every last bit of functionality out of the CONFIG utility. Note that in many of these examples, you’re changing the configuration, which is something you typically do when you’re doing an upgrade install: upgrade the hardware configuration, run DECOM, then update the CONFIG deck to match the hardware changes at a release boundary.

Note that when you’re doing something like adding RTDs, there’s more to it than just plugging in the hardware and updating the CONFIG deck...you need to do all the good stuff we discussed back in “RTD Definitions” on page 21.

CONFIG Example: All Hosts Access VTDs in One VTSS, Only Selected Hosts Access VTDs in Second VTSS

FIGURE 7-7 shows example CONFIG JCL to define a VSM configuration as follows:

- The VTD statement specifies default VTD addresses 8900 - 893F for VTSS1. All hosts have access to these VTDs by their default addresses.
- No default VTD addresses are specified for VTSS2. The VTD statements that immediately follow the HOST statements for MVS1 and MVS2 specify that only these hosts can access the VTDs in VTSS2 by the addresses 9900 - 993F. HOST statement MVS3 is a placeholder; this host cannot access the VTDs in VTSS2, and is disabled from initiating migrates and reclaims.

```
//UPDATECFGEXEC PGM=SWSADMIN,PARM='MIXED'
//STEPLIB DD DSN=hlq.SLSLINK,DISP=SHR
//SLSCNTL DD DSN=FEDB.VSMLMULT.DBASEPRM,DISP=SHR
//SLSCNTL2 DD DSN=FEDB.VSMLMULT.DBASESEC,DISP=SHR
//SLSSTBY DD DSN=FEDB.VSMLMULT.DBASETBY,DISP=SHR
//CFG22202 DD DSN=FEDB.VSMLMULT.CFG22202,DISP=SHR
//SLSPRINT DD   SYSOUT=*
//SLSIN DD   *
 CONFIG CDSLEVEL(V62ABOVE)
 GLOBAL MAXVTV=65000 MVCFREE=60 VTVATTR=SCRATCH RECALWER=YES
 LOCKSTR=STK_VTCS_LOCKS VTVPAGE=LARGE
 RECLAIM THRESHLD=70 MAXMVC=30 START=40 CONMVC=5
 VTVVOL LOW=905000 HIGH=999999 SCRATCH
 VTVVOL LOW=C00000 HIGH=C25000 SCRATCH
 VTVVOL LOW=RMM000 HIGH=RMM020 SCRATCH
 MVCVOL LOW=N25980 HIGH=N25989
 MVCVOL LOW=N35000 HIGH=N35999
 VTSSNAME=VTSS1  LOW=70 HIGH=80  MAXMIG=3  RETAIN=5
 RTDNAME=VT128800 DEVNO=8800 CHANIF=0A
 RTDNAME=VTS18801 DEVNO=8801 CHANIF=0I
 RTDNAME=VTS18802 DEVNO=8802 CHANIF=1A
 RTDNAME=VTS18803 DEVNO=8803 CHANIF=1I
 VTDLOW=8900 HIGH=893F
 VTSSNAME=VTSS2  LOW=70 HIGH=80  MAXMIG=3  RETAIN=5
 RTDNAME=VTS28804 DEVNO=8804 CHANIF=0A
 RTDNAME=VTS28805 DEVNO=8805 CHANIF=0I
 RTDNAME=VTS28806 DEVNO=8806 CHANIF=1A
 RTDNAME=VTS28807 DEVNO=8807 CHANIF=1I
 HOST   NAME=MVS1
 VTD    LOW=9900 HIGH=993F
 HOST   NAME=MVS2
 VTD    LOW=9900 HIGH=993F
 HOST   NAME=MVS3 NOMIGRAT NORECLAM
```

FIGURE 7-7 CONFIG example: All hosts access VTDs in one VTSS, selected hosts access VTDs in second VTSS

CONFIG Example: Update Configuration to Add RTDs

FIGURE 7-8 shows example JCL to run CONFIG to add RTDs VTS18811 and VTS18813 (connected to VTSS1) to the configuration shown in Figure 19 on page 98. Because of NCO, you can add the RTDs dynamically (no RESET required), by simply adding new definitions for RTDs VTS18811 and VTS18813.

```
//UPDATECFGEXEC PGM=SWSADMIN,PARM='MIXED'
//STEPLIB DD DSN=hlq.SLSLINK,DISP=SHR
//SLSCNTL DD DSN=FEDB.VSMLMULT.DBASEPRM,DISP=SHR
//SLSCNTL2 DD DSN=FEDB.VSMLMULT.DBASESEC,DISP=SHR
//SLSSTBY DD DSN=FEDB.VSMLMULT.DBASETBY,DISP=SHR
//CFG22202 DD DSN=FEDB.VSMLMULT.CFG22202,DISP=SHR
//SLSPRINT DD   SYSOUT=*
//SLSIN DD   *
 CONFIG CDSLEVEL(V62ABOVE)
 GLOBAL MAXVTV=65000 MVCFREE=60 VTVATTR=SCRATCH RECALWER=YES
 LOCKSTR=STK_VTCS_LOCKS VTVPAGE=LARGE
 RECLAIM THRESHLD=70 MAXMVC=30 START=40 CONMVC=5
 VTVVOL LOW=905000 HIGH=999999 SCRATCH
 VTVVOL LOW=C00000 HIGH=C25000 SCRATCH
 VTVVOL LOW=RMM000 HIGH=RMM020 SCRATCH
 MVCVOL LOW=N25980 HIGH=N25989
 MVCVOL LOW=N35000 HIGH=N35999
 VTSSNAME=VTSS1  LOW=70 HIGH=80  MAXMIG=3  RETAIN=5
 RTDNAME=VTS18800 DEVNO=8800 CHANIF=0A
 RTDNAME=VTS18801 DEVNO=8801 CHANIF=0I
 RTDNAME=VTS18802 DEVNO=8802 CHANIF=1A
 RTDNAME=VTS18803 DEVNO=8803 CHANIF=1I
 RTDNAME=VTS18811 DEVNO=8811 CHANIF=0E
 RTDNAME=VTS18813 DEVNO=8813 CHANIF=1E
 VTDLOW=8900 HIGH=893F
 VTSSNAME=VTSS2  LOW=70 HIGH=80  MAXMIG=3  RETAIN=5
 RTDNAME=VTS28804 DEVNO=8804 CHANIF=0A
 RTDNAME=VTS28805 DEVNO=8805 CHANIF=0I
 RTDNAME=VTS28806 DEVNO=8806 CHANIF=1A
 RTDNAME=VTS28807 DEVNO=8807 CHANIF=1I
 HOST   NAME=MVS1
 VTD    LOW=9900 HIGH=993F
 HOST   NAME=MVS2
 VTD    LOW=9900 HIGH=993F
 HOST   NAME=MVS3
```

FIGURE 7-8 CONFIG example: updating configuration to add RTDs

CONFIG Example: Update Configuration to Add MVCs and VTVs and Change AMTs

FIGURE 7-9 shows example JCL to run CONFIG to modify the configuration shown in FIGURE 7-8 on page 125 by:

- Adding VTVs C25001 to C50000 as scratch.
- Adding MVCs N45000 to N45999.
- Changing the LAMT to 50 and the HAMT to 85 on both VTSS1 and VTSS2.

```
//UPDATECFGEXEC PGM=SWSADMIN,PARM='MIXED'
//STEPLIB DD DSN=hlq.SLSLINK,DISP=SHR
//SLSCNTL DD DSN=FEDB.VSMLMULT.DBASEPRM,DISP=SHR
//SLSCNTL2 DD DSN=FEDB.VSMLMULT.DBASESEC,DISP=SHR
//SLSSTBY DD DSN=FEDB.VSMLMULT.DBASETBY,DISP=SHR
//CFG22202 DD DSN=FEDB.VSMLMULT.CFG22202,DISP=SHR
//SLSPRINT DD   SYSOUT=*
//SLSIN DD   *
 CONFIG CDSLEVEL(V62ABOVE)
 GLOBAL MAXVTV=65000 MVCFREE=60 VTVATTR=SCRATCH RECALWER=YES
 LOCKSTR=STK_VTCS_LOCKS VTVPAGE=LARGE
 RECLAIM THRESHLD=70 MAXMVC=30 START=40 CONMVC=5
 VTVVOL LOW=905000 HIGH=999999 SCRATCH
 VTVVOL LOW=C00000 HIGH=C25000 SCRATCH
 VTVVOL LOW=RMM000 HIGH=RMM020 SCRATCH
 VTVVOL LOW=C25001 HIGH=C50000 SCRATCH
 MVCVOL LOW=N25980 HIGH=N25989
 MVCVOL LOW=N35000 HIGH=N35999
 MVCVOL LOW=N45000 HIGH=N45999
 VTSSNAME=VTSS1  LOW=50 HIGH=85  MAXMIG=3  RETAIN=5
 RTDNAME=VTS18800 DEVNO=8800 CHANIF=0A
 RTDNAME=VTS18801 DEVNO=8801 CHANIF=0I
 RTDNAME=VTS18802 DEVNO=8802 CHANIF=1A
 RTDNAME=VTS18803 DEVNO=8803 CHANIF=1I
 RTDNAME=VTS18811 DEVNO=8811 CHANIF=0E
 RTDNAME=VTS18813 DEVNO=8813 CHANIF=1E
 VTDLOW=8900 HIGH=893F
 VTSSNAME=VTSS2  LOW=50 HIGH=85  MAXMIG=3  RETAIN=5
 RTDNAME=VTS28804 DEVNO=8804 CHANIF=0A
 RTDNAME=VTS28805 DEVNO=8805 CHANIF=0I
 RTDNAME=VTS28806 DEVNO=8806 CHANIF=1A
 RTDNAME=VTS28807 DEVNO=8807 CHANIF=1I
 HOST   NAME=MVS1
 VTD    LOW=9900 HIGH=993F
 HOST   NAME=MVS2
 VTD    LOW=9900 HIGH=993F
 HOST   NAME=MVS3
```

FIGURE 7-9 CONFIG example: updating configuration to add MVCs and VTVs and change AMTs

CONFIG Example: Denying Host Access to a Physically Removed VTSS

Here's an interesting one. You take a VTSS out of the mix...how do you let the hosts know it's not there any more. FIGURE 7-10 shows example JCL to run CONFIG to deny host access to VTSS2 that you physically removed from your configuration. In this example, you simply respecify the VTSS statement for VTSS2 with no parameters to deny host access to this VTSS.

```
//UPDATECFGEXEC PGM=SWSADMIN,PARM='MIXED'
//STEPLIB DD DSN=hlq.SLSLINK,DISP=SHR
//SLSCNTL DD DSN=FEDB.VSMLMULT.DBASEPRM,DISP=SHR
//SLSCNTL2 DD DSN=FEDB.VSMLMULT.DBASESEC,DISP=SHR
//SLSSTBY DD DSN=FEDB.VSMLMULT.DBASETBY,DISP=SHR
//CFG22202 DD DSN=FEDB.VSMLMULT.CFG22202,DISP=SHR
//SLSPRINT DD   SYSOUT=*
//SLSIN DD   *
 CONFIG CDSLEVEL(V62ABOVE)
 GLOBAL MAXVTV=65000 MVCFREE=60 VTVATTR=SCRATCH RECALWER=YES
 LOCKSTR=STK_VTCS_LOCKS VTVPAGE=LARGE
 RECLAIM THRESHLD=70 MAXMVC=30 START=40 CONMVC=5
 VTSSNAME=VTSS1  LOW=70 HIGH=80  MAXMIG=3  RETAIN=5
 RTDNAME=VTS18800 DEVNO=8800 CHANIF=0A
 RTDNAME=VTS18801 DEVNO=8801 CHANIF=0I
 RTDNAME=VTS18802 DEVNO=8802 CHANIF=1A
 RTDNAME=VTS18803 DEVNO=8803 CHANIF=1I
 RTDNAME=VTS18811 DEVNO=8811 CHANIF=0E
 RTDNAME=VTS18813 DEVNO=8813 CHANIF=1E
 VTDLOW=8900 HIGH=893F
 VTSSNAME=VTSS2
```

FIGURE 7-10 CONFIG example: updating configuration to deny host access to a physically removed VTSS

CONFIG Example: Defining Native IP Connections

FIGURE 7-11 and FIGURE 7-12 on page 129 show example CONFIG JCL to define a Bi-Directional Cluster of two VSM5s (VSMPR1 and VSMPR2) with Native IP connections. **Note that**:

- The CLUSTER statement defines the Cluster as consisting of VSMPR1 and VSMPR2.
- There are CLINK statements using the sending Native IP ports of **both VTSSs** to enable the Cluster as Bi-Directional. The sending Native IP ports for outbound CLINK connections are 0O:0 and 1O:0 on both VTSSs.

Note – This CLINK configuration is valid because CLINKs using IP ports, which are on IFF3 cards, each with its own processor and operating system, do not have the "Same Storage Cluster" requirement because they are not controlled through the Shared IUP/VCF card interface.

```
//CREATECF  EXEC PGM=SLUADMIN,PARM='MIXED'
//STEPLIB    DD DSN=hlq.SEALINK,DISP=SHR
//SLSCNTL    DD DSN=FEDB.VSMLMULT.DBASEPRM,DISP=SHR
//SLSCNTL2   DD DSN=FEDB.VSMLMULT.DBASESEC,DISP=SHR
//SLSSTBY    DD DSN=FEDB.VSMLMULT.DBASETBY,DISP=SHR
//SLSPRINT   DD   SYSOUT=*
//SLSIN      DD   *
 CONFIG RESET CDSLEVEL(V61ABOVE)
GLOBAL       MAXVTV=32000 MVCFREE=40 VTVATTR=SCRATCH RECALWER=YES LOCKSTR=
VTCS_LOCKS
REPLICAT=ALWAYS VTVPAGE=LARGE SYNCHREP=YES MAXRTDS=32
RECLAIM     THRESHLD=70 MAXMVC=40  START=35
RECLAIM     THRESHLD=70MAXMVC=40  START=35
VTSS NAME=VSMPR1 LOW=70 HIGH=80 MAXMIG=8 MINMIG=4 RETAIN=5
RTD  NAME=VPR12A00 DEVNO=2A00 CHANIF=0C:0
 RTD  NAME=VPR12A01 DEVNO=2A01 CHANIF=0C:1
 RTD  NAME=VPR12A02 DEVNO=2A02 CHANIF=0C:2
 RTD  NAME=VPR12A03 DEVNO=2A03 CHANIF=0C:3
 RTD  NAME=VPR12A04 DEVNO=2A04 CHANIF=0G:0
 RTD  NAME=VPR12A05 DEVNO=2A05 CHANIF=0G:1
 RTD  NAME=VPR12A06 DEVNO=2A06 CHANIF=0G:2
 RTD  NAME=VPR12A07 DEVNO=2A07 CHANIF=0G:3
 RTD  NAME=VPR12A08 DEVNO=2A08 CHANIF=0K:0
 RTD  NAME=VPR12A09 DEVNO=2A09 CHANIF=0K:1
 RTD  NAME=VPR12A0A DEVNO=2A0A CHANIF=0K:2
 RTD  NAME=VPR12A0B DEVNO=2A0B CHANIF=0K:3
        RTD  NAME=VPR13A00 DEVNO=3A00 CHANIF=1C:0
 RTD  NAME=VPR13A01 DEVNO=3A01 CHANIF=1C:1
RTD  NAME=VPR13A02 DEVNO=3A02 CHANIF=1C:2
 RTD  NAME=VPR13A03 DEVNO=3A03 CHANIF=1C:3
 RTD  NAME=VPR13A04 DEVNO=3A04 CHANIF=1G:0
 RTD  NAME=VPR13A05 DEVNO=3A05 CHANIF=1G:1
 RTD  NAME=VPR13A06 DEVNO=3A06 CHANIF=1G:2
 RTD  NAME=VPR13A07 DEVNO=3A07 CHANIF=1G:3
 RTD  NAME=VPR13A08 DEVNO=3A08 CHANIF=1K:0
 RTD  NAME=VPR13A09 DEVNO=3A09 CHANIF=1K:1
 RTD  NAME=VPR13A0A DEVNO=3A0A CHANIF=1K:2
 RTD  NAME=VPR13A0B DEVNO=3A0B CHANIF=1K:3
VTD LOW=9900 HIGH=99FF
```

FIGURE 7-11 CONFIG example: Dual ACS Bi-Directional Clustered VTSS System, Native IP (Part 1)

```
VTSS NAME=VSMPR2 LOW=70 HIGH=80 MAXMIG=8 MINMIG=4 RETAIN=5
RTD  NAME=VPR22B00 DEVNO=2B00 CHANIF=0C:0
 RTD  NAME=VPR22B01 DEVNO=2B01 CHANIF=0C:1
 RTD  NAME=VPR22B02 DEVNO=2B02 CHANIF=0C:2
 RTD  NAME=VPR22B03 DEVNO=2B03 CHANIF=0C:3
 RTD  NAME=VPR22B04 DEVNO=2B04 CHANIF=0G:0
 RTD  NAME=VPR22B05 DEVNO=2B05 CHANIF=0G:1
 RTD  NAME=VPR22B06 DEVNO=2B06 CHANIF=0G:2
 RTD  NAME=VPR22B07 DEVNO=2B07 CHANIF=0G:3
 RTD  NAME=VPR22B08 DEVNO=2B08 CHANIF=0K:0
 RTD  NAME=VPR22B09 DEVNO=2B09 CHANIF=0K:1
 RTD  NAME=VPR22B0A DEVNO=2B0A CHANIF=0K:2
 RTD  NAME=VPR22B0B DEVNO=2B0B CHANIF=0K:3
RTD  NAME=VPR23B00 DEVNO=3B00 CHANIF=1C:0
 RTD  NAME=VPR23B01 DEVNO=3B01 CHANIF=1C:1
RTD  NAME=VPR23B02 DEVNO=3B02 CHANIF=1C:2
 RTD  NAME=VPR23B03 DEVNO=3B03 CHANIF=1C:3
 RTD  NAME=VPR23B04 DEVNO=3B04 CHANIF=1G:0
 RTD  NAME=VPR23B05 DEVNO=3B05 CHANIF=1G:1
 RTD  NAME=VPR23B06 DEVNO=3B06 CHANIF=1G:2
 RTD  NAME=VPR23B07 DEVNO=3B07 CHANIF=1G:3
 RTD  NAME=VPR23B08 DEVNO=3B08 CHANIF=1K:0
 RTD  NAME=VPR23B09 DEVNO=3B09 CHANIF=1K:1
 RTD  NAME=VPR23B0A DEVNO=3B0A CHANIF=1K:2
 RTD  NAME=VPR23B0B DEVNO=3B0B CHANIF=1K:3
VTD LOW=9900 HIGH=99FF
 CLUSTER NAME=CLUSTER1 VTSSs(VSMPR1,VSMPR2)
 CLINK VTSS=VSMPR1 IPIF=0A:0
 CLINK VTSS=VSMPR1 IPIF=1A:0
CLINK VTSS=VSMPR2 IPIF=0I:0
 CLINK VTSS=VSMPR2 IPIF=1I:0
```

FIGURE 7-12 CONFIG example: Dual ACS Bi-Directional Clustered VTSS System, Native IP (Part 2)

CHAPTER 8

Configuring the Host Software for VLE 1.0

This chapter provides the software configuration for VLE appliances as described in the following sections:

- "Key Configuration Values" on page 132
- "Common Configuration Tasks (All Examples)" on page 133
- "Example 1: One VTSS Connected to One VLE Appliance, Direct Connect (No Switch)" on page 134
- "Example 2: Four VTSSs Connected to One VLE Appliance, Direct Connect (No Switch)" on page 139

Note – These three examples use the same TCP/IP network and addressing scheme of the corresponding three examples shown in *VLE Planning Guide*.

Key Configuration Values

The following sections describes 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 appliance (set by the VLE GUI) is specified in the following:

- The `STORMNGR` parameter value on the VTCS `CONFIG TAPEPLEX` statement for the TapePlex that connects to the VLE appliance.
- The `STORMNGR` parameter value on the VTCS `CONFIG RTD` statement for the VLE appliance.
- The `NAME` parameter value on the SMC `STORMNGR` command that defines the VLE appliance to SMC.
- The `STORMNGR` parameter value on the SMC `SERVER` command for the VLE appliance.

VLE Data Port and VSM5 IFF3 Card Target IP Addresses

These IP addresses are **initially set** on the VSM5 DOP `IFF IP Configuration Status` panel and at the VLE GUI and the values **must match**. On the DOP panel, they are set as IP addresses with their equivalent *c:ip* addresses shown, which are required for the `CONFIG RTD IPIF` parameter.

IP Addresses of VLE Ports for Host (UUI) Communication

These addresses 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 "Common Configuration Tasks (All Examples)" on page 133.

Common Configuration Tasks (All Examples)

Note –

- VMVC volsers must be defined both to the host software and through the VLE GUI; for more information, see *VLE Planning Guide* and *Installing, Configuring, and Servicing the VLE Appliance*
- The VLE requires SMC to have an OMVS RACF security entry in order to have a TCP/IP connection to the host.
- Migration controls apply to migration to VLE as well as to RTDs, so plan migration policies for VLE as well.

Creating VMVC Volume Pools

1. **Code HSC VOLATTR statements to define the VMVCs to HSC.**

   For example, to define two separate VMVC volser ranges for VLE1 and VLE2:

   ```
   VOLATTR SERIAL(VL0000-VL1200)
   VOLATTR SERIAL(VL2000-VL3200)
   ```

2. **In your VTCS CONFIG JCL, code MVCVOL statements to define the VMVCs to VTCS.**

   For example:

   ```
   MVCVOL LOW=VL0000 HIGH=VL1200
   MVCVOL LOW=VL2000 HIGH=VL3200
   ```

3. **Code HSC MVCPOOL statements to define the VMVC pools.**

   For example:

   ```
   MVCPOOL VOLSER(VL0000-VL1200)
   MVCPOOL VOLSER(VL2000-VL3200)
   ```

4. **Run the VT MVCDEF command to activate the updated data set, for example:**

   ```
   .VT MVCDEF DSN(VSM.VMVCPOOL)
   ```

Example 1: One VTSS Connected to One VLE Appliance, Direct Connect (No Switch)

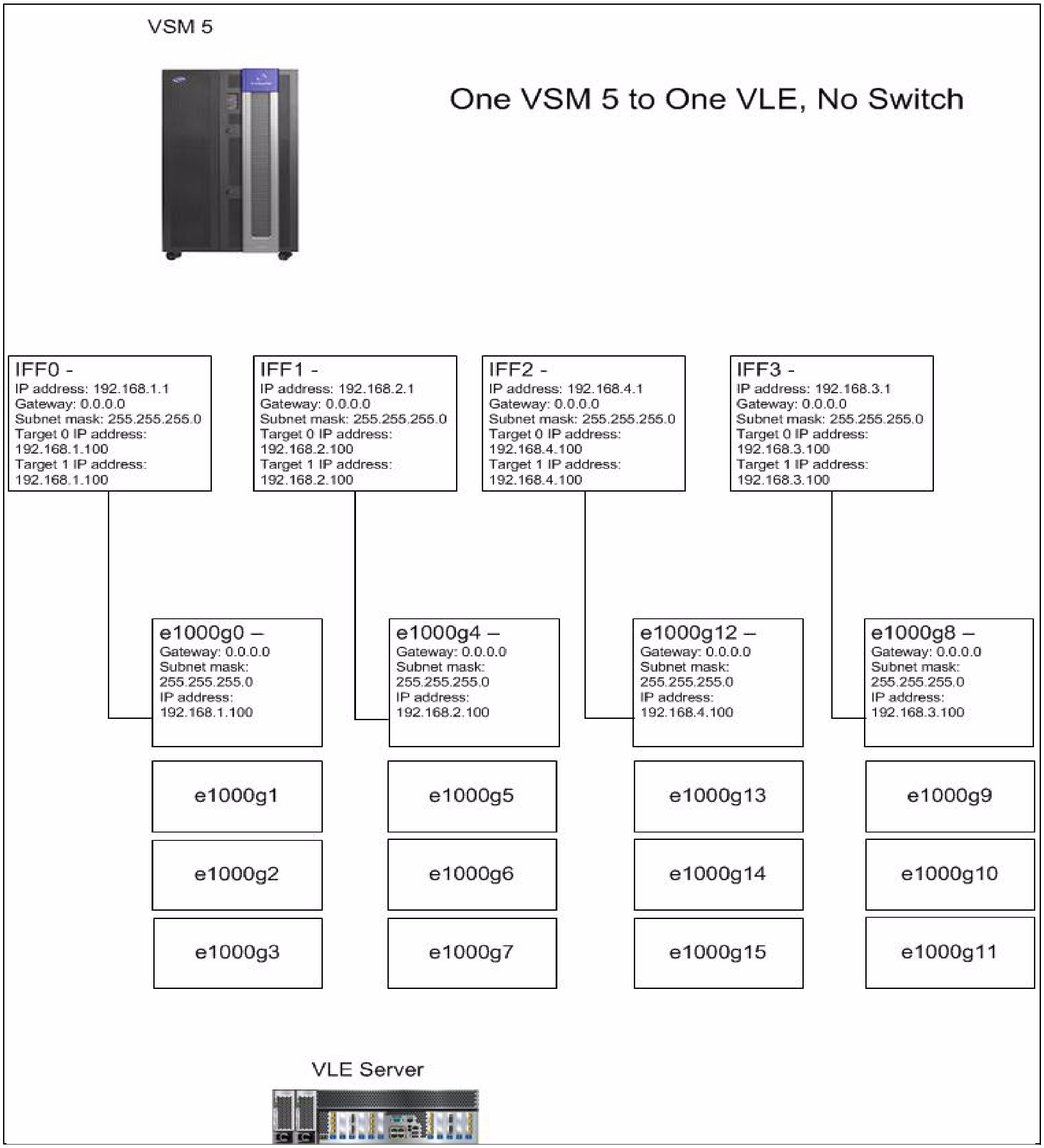


FIGURE 8-1 Example 1: One VTSS Connected to One VLE Appliance, Direct Connect (No Switch)

As FIGURE 8-1 on page 134 and TABLE 8-1 show, in this one VTSS to one VLE appliance (no switch) example, a single target on each IFF card connects to a single port on the VLE appliance, where the IP addresses **must match**. **Note that** the third octet of the IP addresses is unique to each IFF card to VLE port connection, so these connections share a unique subnet.

For this example, assume the following values:

- VLE subsystem name of `VLESERV1`
- VLE server UUI communication IP address of 192.168.1.10
- Single TapePlex, `TMVSA`
- Single VTSS, `VTSS1`

TABLE 8-1 Example 1 Configuration Values

IFF Card and Target	IPIF Value	Connected to VLE Port...	Target IP and VLE Port IP
IFF0 Target 0	0A:0	e1000g0	192.168.1.1
IFF0 Target 1	0A:1	e1000g0	192.168.1.1
IFF1 Target 0	0I:0	e1000g4	192.168.2.1
IFF1 Target 1	0I:1	e1000g4	192.168.2.1
IFF2 Target 0	1A:0	e1000g8	192.168.4.1
IFF2 Target 1	1A:1	e1000g8	192.168.4.1
IFF3 Target 0	1I:0	e1000g12	192.168.3.1
IFF3 Target 1	1I:1	e1000g12	192.168.3.1

Configuring the System

To configure the example system shown in FIGURE 8-1 on page 134, do the following:

1. **Ensure that your system has all prerequisites.**
2. **Code the following SMC commands:**
   - A `STORMNGR` command for VLE appliance `VLESERV1` attached to `TMVSA`.
   - A `SERVER` command that points to the VLE appliance, where:
     - The `STORMNGR` parameter value is `VLESERV1`.
     - The `IP` parameter value is the VLE server IP address of 192.168.1.10.
     - The `PORT` parameter value is 60000; this value is **always** used for the `SERVER PORT` parameter for SMC communication with a VLE appliance.

   For 7.0, you may want to do this in your SMC `CMDS` file. For example:

```
STORMNGR NAME(VLESERV1)
SERVER NAME(TAPE1) STORMNGR(VLESERV1)IP(192.168.1.10)PORT(60000)
```

3. Code CONFIG **JCL for TapePlex TMVSA, as shown in** FIGURE 8-2**.**

In this figure, note:

- The CONFIG TAPEPLEX statement, which defines TMVSA as THISPLEX and also defines its connection to the VLESERV1 appliance.
- The CONFIG RTD statements for VTSS1, which specify:
  - The connection to the VLESERV1 appliance.
  - The IPIF value and an RTD name for each IFF target to VLE connection.
- The VMVC reclaim threshold (RECLAIM VLTHRES) is 30.

```
//CREATCFG    EXEC PGM=SLUADMIN,PARM='MIXED'
//STEPLIB  DD DSN=hlq.SEALINK,DISP=SHR
//SLSCNTL  DD DSN=hlq.TMVSA.DBASEPRM,DISP=SHR
//SLSCNTL2 DD DSN=hlq.TMVSA.DBASESEC,DISP=SHR
//SLSSTBY  DD DSN=hlq.TMVSA.DBASESBY,DISP=SHR
//SLSPRINT DD   SYSOUT=*
//SLSIN    DD   *
 CONFIG RESET CDSLEVEL(V71ABOVE)
 GLOBAL MAXVTV=65000 MVCFREE=60 VTVATTR=SCRATCH RECALWER=YES
 LOCKSTR=STK_VTCS_LOCKS VTVPAGE=LARGE REPLICAT=CHANGED
 RECLAIM THRESHLD=70 VLTHRES=30 MAXMVC=30 START=40 CONMVC=5
TAPEPLEX THISPLEX=TMVSA STORMNGR=VLESERV1
VTSS NAME=VTSS1 LOW=70 HIGH=80 MAXMIG=8 MINMIG=4 RETAIN=5
RTD  NAME=VL1RTD1 STORMNGR=VLESERV1 IPIF=0A:0
RTD  NAME=VL1RTD2 STORMNGR=VLESERV1 IPIF=0A:1
RTD  NAME=VL1RTD3 STORMNGR=VLESERV1 IPIF=0I:0
RTD  NAME=VL1RTD4 STORMNGR=VLESERV1 IPIF=0I:1
RTD  NAME=VL1RTD5 STORMNGR=VLESERV1 IPIF=1A:0
RTD  NAME=VL1RTD6 STORMNGR=VLESERV1 IPIF=1A:1
RTD  NAME=VL1RTD7 STORMNGR=VLESERV1 IPIF=1I:0
RTD  NAME=VL1RTD8 STORMNGR=VLESERV1 IPIF=1I:1
VTD LOW=6900 HIGH=69FF
```

FIGURE 8-2 CONFIG for Example 1

t Defining Policies

To define policies for the example system shown in FIGURE 8-1 on page 134:

1. Create the Storage Class for VLE replication.

```
STOR NAME(VLCPY1) STORMNGR(VLESERV1)
```

FIGURE 8-3 VLE Storage Class

In FIGURE 8-3, the `STORclas` statements define the Storage Class `VLCPY1` for replication to VMVCs on VLESERV1.

2. Create the Management Class that points to the Storage Class in Step 1.

```
MGMT NAME(VLECOPY) IMMED(DELETE) MIGPOL(VLCPY1)
```

FIGURE 8-4 Management Class for VLE

In FIGURE 8-4, the `MGMTclas` statement does the following:

- Migrates a VTV copy to VLESERV1.
- After the migration succeeds, deletes the VTV from the VTSS.

3. Create an SMC Policy that specifies virtual media and assigns the Management Class created in Step 2.

```
POLICY NAME(VLEMIGR) MEDIA(VIRTUAL) MGMT(VLECOPY)
```

FIGURE 8-5 Policy for VLE

4. Create `TAPEREQ` statements to route data to VSM and assign the corresponding Policy to the data.

```
TAPEREQ DSN(*.PAYROLL.**) POLICY(VLEMIGR)
TAPEREQ DSN(*.HR.**) POLICY(VLEMIGR)
```

FIGURE 8-6 TAPEREQ Statement to Route Data, Assign Policy

Note – Also note that although you can use SMC policies to direct your data to a specific esoteric, StorageTek recommends using **only** `MGMTCLAS` so that the SMC/VTCS allocation influencing can use any VTSS that supports the `MGMTCLAS` requirements.

Example 2: Four VTSSs Connected to One VLE Appliance, Direct Connect (No Switch)

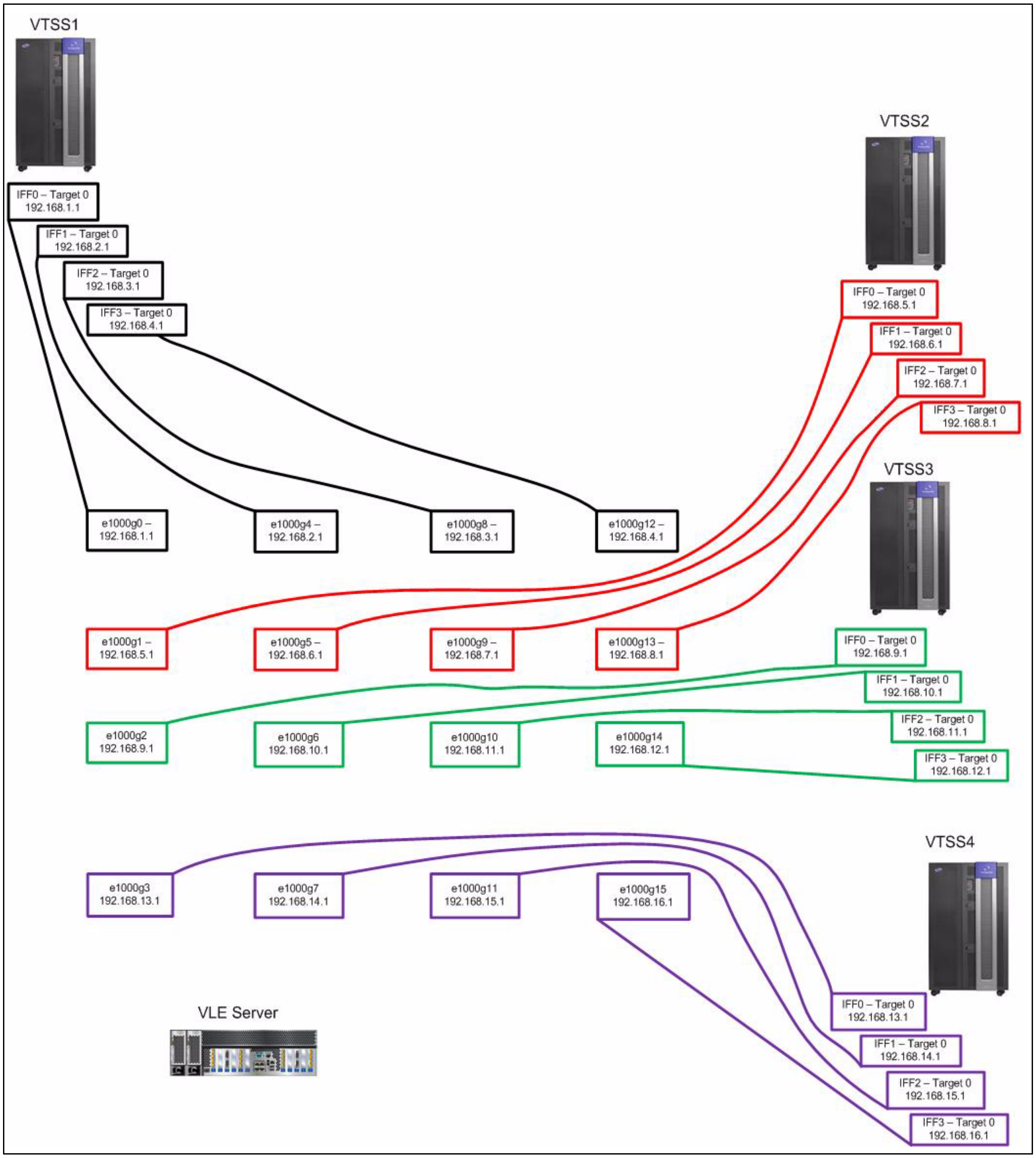


FIGURE 8-7 Example 2: Four VTSSs Connected to One VLE Appliance, Direct Connect (No Switch)

As FIGURE 8-7 and TABLE 8-2 show, in this four VTSS to one VLE appliance (no switch) example, each IFF target to VLE port connection (where the IP addresses **must match**) is on its own unique subnet, as shown by the different colors for each subnet. For this example, assume the following values:

- VLE subsystem name of `VLESERV2`
- VLE server UUI communication IP address of 192.168.1.20
- Single TapePlex, `TMVSB`
- Four VTSSs: VTSS1, VTSS2, VTSS3, VTSS4

TABLE 8-2 Example 2 Configuration Values

VSM5	IFF Card and Target	IPIF Value	Connected to VLE Port...	Target and VLE Port IP
VTSS1	IFF0 Target 0	0A:0	e1000g0	192.168.1.1
	IFF1 Target 0	0I:0	e10004	192.168.2.1
	IFF2 Target 0	1A:0	e1000g8	192.168.3.1
	IFF3 Target 0	1A:1	e1000g12	192.168.4.1
VTSS2	IFF0 Target 0	0A:0	e1000g1	192.168.5.1
	IFF1 Target 0	0I:0	e1000g5	192.168.6.1
	IFF2 Target 0	1A:0	e1000g9	192.168.7.1
	IFF3 Target 0	1A:1	e1000g13	192.168.8.1
VTSS3	IFF0 Target 0	0A:0	e1000g2	192.168.9.1
	IFF1 Target 0	0I:0	e1000g6	192.168.10.1
	IFF2 Target 0	1A:0	e1000g10	192.168.11.1
	IFF3 Target 0	1A:1	e1000g14	192.168.12.1
VTSS4	IFF0 Target 0	0A:0	e1000g3	192.168.13.1
	IFF1 Target 0	0I:0	e1000g7	192.168.14.1
	IFF2 Target 0	1A:0	e1000g11	192.168.15.1
	IFF3 Target 0	1A:1	e1000g15	192.168.16.1

Configuring the System

To configure the example system shown in FIGURE 8-7 on page 139, do the following:

1. **Ensure that your system has all prerequisites.**
2. **Code the following SMC commands:**
   - A `STORMNGR` command for VLE appliance `VLESERV2` attached to `TMVSB`.
   - A `SERVER` command that points to the VLE appliance, where:
     - The `STORMNGR` parameter value is `VLESERV2`.
     - The `IP` parameter value is the VLE server IP address of 192.168.1.20.
     - The `PORT` parameter value is 60000; this value is **always** used for the `SERVER PORT` parameter for SMC communication with a VLE appliance.

   For 7.0, you may want to do this in your SMC `CMDS` file. For example:

```
STORMNGR NAME(VLESERV2)
SERVER NAME(TAPE2) STORMNGR(VLESERV2)IP(192.168.1.20)PORT(60000)
```

3. **Code** `CONFIG` **JCL for TapePlex TMVSB, as shown in** FIGURE 8-8**.**

In this figure, note:

- The `CONFIG TAPEPLEX` statement, which defines `TMVSB` as `THISPLEX` and also defines its connection to the `VLESERV2` appliance.
- The `CONFIG RTD` statements for `VTSS1` through `VTSS4`, which specify:
  - The connection to the `VLESERV2` appliance.
  - The `IPIF` value and an RTD name for each IFF target to VLE connection.
- The VMVC reclaim threshold (`RECLAIM VLTHRES`) is 30.

```
//CREATCFG    EXEC PGM=SLUADMIN,PARM='MIXED'
//STEPLIB  DD DSN=hlq.SEALINK,DISP=SHR
//SLSCNTL  DD DSN=hlq.TMVSB.DBASEPRM,DISP=SHR
//SLSCNTL2 DD DSN=hlq.TMVSB.DBASESEC,DISP=SHR
//SLSSTBY  DD DSN=hlq.TMVSB.DBASESBY,DISP=SHR
//SLSPRINT DD    SYSOUT=*
//SLSIN    DD    *
 CONFIG RESET CDSLEVEL(V71ABOVE)
 GLOBAL MAXVTV=65000 MVCFREE=60 VTVATTR=SCRATCH RECALWER=YES
 LOCKSTR=STK_VTCS_LOCKS VTVPAGE=LARGE REPLICAT=CHANGED
 RECLAIM THRESHLD=70 VLTHRES=30 MAXMVC=30 START=40 CONMVC=5
TAPEPLEX THISPLEX=TMVSB STORMNGR=VLESERV2
VTSS NAME=VTSS1 LOW=70 HIGH=80 MAXMIG=8 MINMIG=4 RETAIN=5
 RTD  NAME=VSM1VRTD1 STORMNGR=VLESERV2 IPIF=0A:0
RTD  NAME=VSM1VRTD2 STORMNGR=VLESERV2 IPIF=0I:0
RTD  NAME=VSM1VRTD3 STORMNGR=VLESERV2 IPIF=1A:0
RTD  NAME=VSM1VRTD4 STORMNGR=VLESERV2 IPIF=1A:1
VTD LOW=6900 HIGH=69FF

VTSS NAME=VTSS2 LOW=70 HIGH=80 MAXMIG=8 MINMIG=4 RETAIN=5
RTD  NAME=VSM2VRTD1 STORMNGR=VLESERV2 IPIF=0A:0
RTD  NAME=VSM2VRTD2 STORMNGR=VLESERV2 IPIF=0I:0
RTD  NAME=VSM2VRTD3 STORMNGR=VLESERV2 IPIF=1A:0
RTD  NAME=VSM2VRTD4 STORMNGR=VLESERV2 IPIF=1A:1
VTD LOW=7900 HIGH=79FF

VTSS NAME=VTSS3 LOW=70 HIGH=80 MAXMIG=8 MINMIG=4 RETAIN=5
RTD  NAME=VSM3VRTD1 STORMNGR=VLESERV2 IPIF=0A:0
RTD  NAME=VSM3VRTD2 STORMNGR=VLESERV2 IPIF=0I:0
RTD  NAME=VSM3VRTD3 STORMNGR=VLESERV2 IPIF=1A:0
RTD  NAME=VSM3VRTD4 STORMNGR=VLESERV2 IPIF=1A:1
VTD LOW=8900 HIGH=89FF

VTSS NAME=VTSS4 LOW=70 HIGH=80 MAXMIG=8 MINMIG=4 RETAIN=5
RTD  NAME=VSM4VRTD1 STORMNGR=VLESERV2 IPIF=0A:0
RTD  NAME=VSM4VRTD2 STORMNGR=VLESERV2 IPIF=0I:0
RTD  NAME=VSM4VRTD3 STORMNGR=VLESERV2 IPIF=1A:0
RTD  NAME=VSM4VRTD4 STORMNGR=VLESERV2 IPIF=1A:1
VTD LOW=9900 HIGH=99FF
```

FIGURE 8-8 CONFIG for Example 2

Defining Policies

To define policies for the example system shown in FIGURE 8-7 on page 139:

1. **Create the Storage Class for VLE replication.**

```
STOR NAME(VLCPY2) STORMNGR(VLESERV2)
```

FIGURE 8-9 VLE Storage Class

In FIGURE 8-9, the `STORclas` statements define the Storage Class `VLCPY2` for replication to VMVCs on VLESERV2.

2. **Create the Management Class that points to the Storage Class in Step 1.**

```
MGMT NAME(VLECOPY2) IMMED(DELETE) MIGPOL(VLCPY2)
```

FIGURE 8-10 Management Class for VLE

In FIGURE 8-10, the `MGMTclas` statement does the following:

- Migrates a VTV copy to VLESERV2.
- After the migration succeeds, deletes the VTV from the VTSS.

3. **Create an SMC Policy that specifies virtual media and assigns the Management Class created in Step 3.**

```
POLICY NAME(VLEMIGR2) MEDIA(VIRTUAL) MGMT(VLECOPY2)
```

FIGURE 8-11 Policy for VLE

4. **Create `TAPEREQ` statements to route data to VSM and assign the corresponding Policy to the data.**

```
TAPEREQ DSN(*.PAYROLL.**) POLICY(VLEMIGR2)
TAPEREQ DSN(*.HR.**) POLICY(VLEMIGR2)
```

FIGURE 8-12 TAPEREQ Statement to Route Data, Assign Policy

Note – Also note that although you can use SMC policies to direct your data to a specific esoteric, StorageTek recommends using **only** `MGMTCLAS` so that the SMC/VTCS allocation influencing can use any VTSS that supports the `MGMTCLAS` requirements.

CHAPTER 9

Completing the VSM Configuration

In the home stretch now...we've configured VTCS, now it's time to complete some other key tasks to round out a working VSM configuration, to whit:

- "Updating the HSC PARMLIB Member (SLSSYSxx)" on page 146
- "Adding SMF Parameters for VTCS to SYS1.PARMLIB" on page 147
- "Connecting MVS/CSC Clients to VSM" on page 148

As an alternative to the MVS/CSC and LibraryStation configuration, you can simply install SMC in your client MVS system, and SMC will route virtual allocation and mount requests to HSC with HTTP server running in a remote server HSC system. For more information, see "Using the SMC Client/Server Feature" on page 150.

Note – In a future release of NCS/VTCS, StorageTek will drop support for MVS/CSC clients communicating with HSC via LibraryStation. StorageTek strongly recommends, therefore, that you implement the SMC client/server feature for the 6.2 release.

- "Connecting Non-MVS/CSC Clients to VSM" on page 155
- "Updating the Tape Management System" on page 156
- "Updating HSM" on page 159
- "Routing Data Sets to VSM" on page 160
- "Restarting NCS/VTCS" on page 164

Tip – Several tasks in this Chapter require you to specify VSM system values that you determined on page 10 and recorded in TABLE A-1 on page 165.

Updating the HSC PARMLIB Member (SLSSYSxx)

You can specify the VT MVCDEF command as a statement in the HSC PARMLIB. FIGURE 9-1 shows an example of TREQDEF, VT MVCDEF, and MGMTDEF commands specified as statements in the HSC PARMLIB member.

```
TREQDEF DSN(SMC.TAPEREQ)
VT MVCDEF DSN(VSM.MVCPOOL)
MGMTDEF DSN(HSC.PARMS)
COMMP METH LMU
FEAT VSM(ADVMGMT)
```

FIGURE 9-1 Example: Updating the HSC PARMLIB Member for VSM

In FIGURE 9-1:

SMC.TAPEREQ

is the data set that contains your system's TAPEREQ statements (including TAPEREQ statements for VTVs).

VSM.MVCPOOL

is the data set that contains your system's MVCPool statements.

HSC.PARMS

is the data set that contains your system's MGMTclas and STORclas statements.

COMMP METH LMU

specifies that LMU is the communications method. StorageTek recommends that you specify either LMU or VTAM, not CDS to allow even sharing of resources in a multi-host environment.

FEAT VSM(ADVMGMT)

enables the Advanced Management Feature.

Adding SMF Parameters for VTCS to SYS1.PARMLIB

HSC can produce SMF record subtypes for VTCS events. To produce these record subtypes, you must add two statements to your SMF parameters in SYS1.PARMLIB member SMFPRMxx to specify the following:

- HSC subsystem for which records are produced
- Recording interval in seconds
- SMF record subtypes. The record subtypes must be specified as a list (*subtype1*, *subtype2*,...*subtypen*), as a range (*subtype1*-*subtypen*), or as a combination *(subtype1*, *subtype2*-*subtypen*). A range must be specified using a dash; a colon is invalid for a range.

Tip – If you use ExPR for VSM reporting, StorageTek recommends that you specify that your system produces the HSC SMF record subtypes 1 through 8 and 10, 11, 13, 14, 15, 16, 17, 18, 19, 20, 21, 25, 26, 27, 28, and 29 as shown in FIGURE 9-2.

FIGURE 9-2 shows example statements that produce record subtypes 1 through 8 and 10, 11, 13, 14, 15, 16, 17, 18, 19, 20, 21, 25, 26, 27, 28, and 29 at 15 minute intervals for HSC subsystem SLS0.

```
SUBSYS(SLS0,INTERVAL(001500),TYPE(255))
SUBPARM(SLS0(SUBTYPE,
(1-8, 10, 11, 13, 14, 15, 16, 17, 18, 19, 20, 21, 25, 26, 27, 28, 29)))
```

FIGURE 9-2 SYS1.PARMLIB member SMFPRMxx example for VTCS SMF records

Connecting MVS/CSC Clients to VSM

The following procedure tells how to update LibraryStation and MVS/CSC to connect MVS/CSC clients to VSM.

t To connect MVS/CSC clients to VSM:

1. **Define a virtual ACS using the LibraryStation VIRTACS statement.**

   For example, to define virtual ACS 126 and map it to VTSS VTSS02, create the following VIRTACS statement:

   ```
   VIRTACS ID(126) VTSSNAME(VTSS02)
   ```

   This concludes this procedure if *both* of following are true:

   - Clients use the same MVS device numbers defined for VSM as described in "VTD Unit Addresses" on page 13.
   - Your configuration has no cartridge tape UCB defined with an MVS device number that matches a device number defined for VSM.

   Otherwise, continue with Step 2 to map the MVS device numbers defined for VSM to client device numbers.

2. **For each VTSS to which MVS/CSC clients connect, define VTD device addresses using the MVS/CSC LIBUNIT statement.**

Caution – Each device in the LIBUNIT statement must be represented by a UCB and may not be used to access any device other than the associated VTD. These device addresses do not, however, have to be online.

   For more information about the LIBUNIT statement, see *MVS/CSC Configuration Guide*.

   For example, for a single VTSS (VSM2 or VSM3), create a LIBUNIT statement such as the following:

   ```
   LIBUNIT (B00,B01,B02,B03,B04,B05,B06,B07,  -
   B08,B09,B0A,B0B,B0C,B0D,B0E,B0F,  -
   .
   .
   B38,B39,B3A,B3B,B3C,B3D,B3E,B3F)
   ```

3. **For each VTSS to which MVS/CSC clients connect, map the VTD device addresses from Step 2 to the VTD virtual ACS locations using the MVS/CSC UNITMAP statement.**

   For more information about the UNITMAP statement, see *MVS/CSC Configuration Guide*.

   See TABLE 1-11 on page 39 and TABLE 1-12 on page 42 for information on VTD locations in virtual ACSs.

   For example, for the device addresses you defined in Step 2 on page 148, create a UNITMAP statement such as the following:

   ```
   UNITMAP (B00,7E:00:1:0,B01,7E:00:1:1,  -
   ```

B02,7E:00:1:2,B03,7E:00:1:3, -

.

B3E,7E:03:4:2,B3F,7E:03:4:3)

Note – The VIRTACS statement specifies virtual ACS IDs in decimal, but the UNITMAP statement specifies these IDS in hexadecimal.

This concludes this procedure if you use the HSC common subpool. Otherwise, continue with Step 4.

4. **Define an HSC subpool that contains VTVs.**

   For more information, see *HSC System Programmer's Guide for MVS.*

5. **Define a LibraryStation subpool that corresponds to the HSC subpool in Step 4 using the LibraryStation SPNUM statement.**

   For example, create the following SPNUM statement to define VTV subpool 7 that corresponds to HSC subpool LSVIRT1.

   ```
   SPNUM NUM(07) SPNAME(LSVIRT1) VIRT(YES)
   ```

Using the SMC Client/Server Feature

SMC provides a client/server feature that lets you run SMC only on the client hosts and HSC/VTCS and the HTTP server on one or more server hosts. Using the SMC client/server feature provides the following benefits:

- **Reduces the number of hosts on which you run HSC/VTCS**. StorageTek recommends that you execute HSC/VTCS on only two hosts (primary and backup). Running HSC/VTCS on fewer hosts reduces CDS contention and eliminates the need to manage multiple MVS syslog files.
- **Eliminates the need for MVS/CSC communicating with LibraryStation**. The SMC client/server feature allows SMC to communicate directly with a remote HSC/VTCS without using either MVS/CSC or LibraryStation.
- **Communicate with multiple HSC/VTCS TapePlex systems** representing physically different hardware configurations.

For example, in FIGURE 9-3 on page 151, client MVSA is running SMC only, but is connected to two servers running HSC/VTCS and HTTP server.

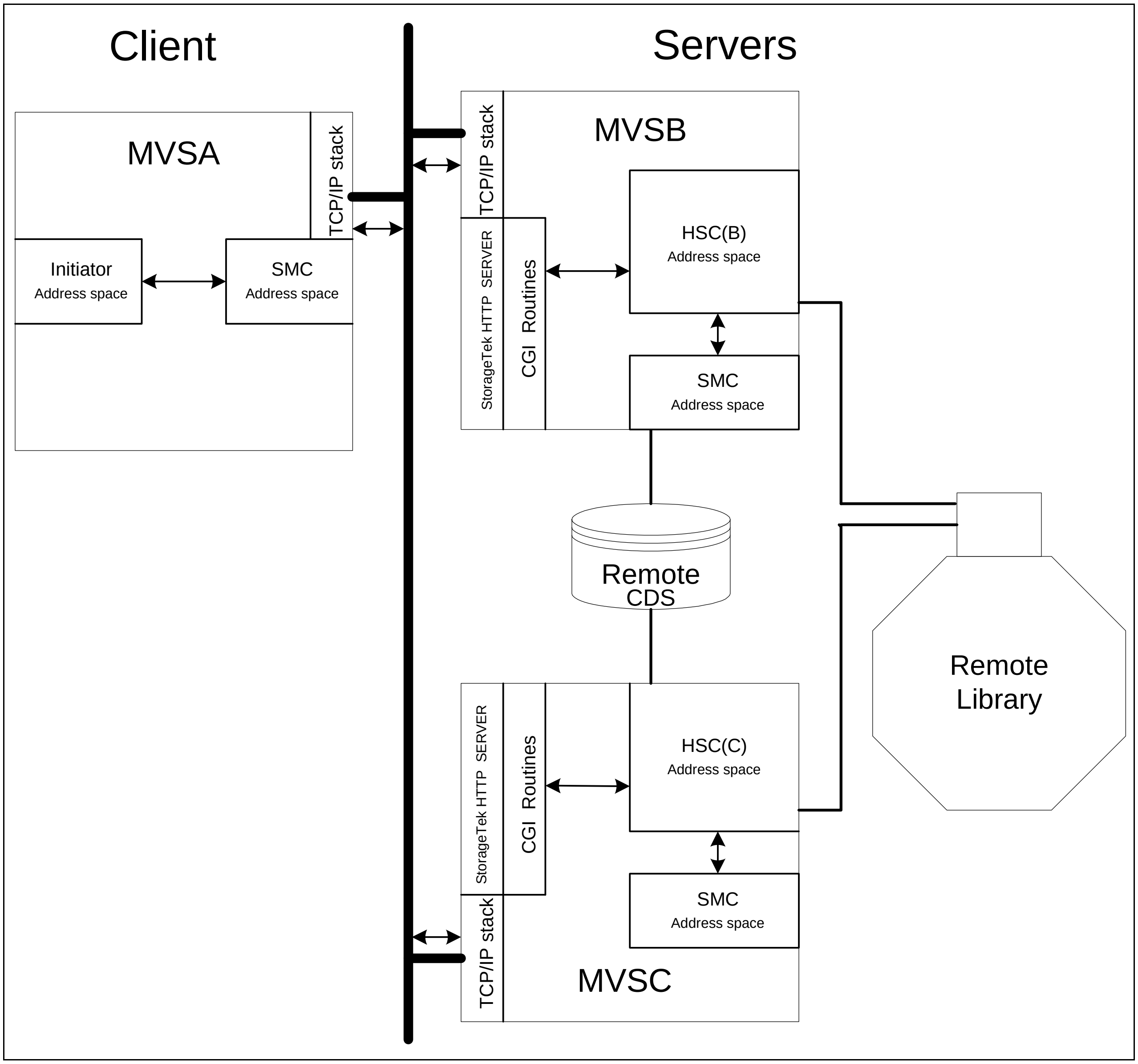


FIGURE 9-3 SMC Configuration: One Client, Two Servers

To make the client/server connections shown In FIGURE 9-3, MVSA a combination of the SMC TAPEPLEX and SERVer commands, for example:

```
TAPEPLEX NAME(SHRLIB)
SERVER NAME(REMPATH1) TAPEPLEX(SHRLIB) HOST(MVSB)
SERVER NAME(REMPATH2) TAPEPLEX(SHRLIB) HOST(MVSC)
```

The TAPEPLEX command lets the client assign a name to the *TapePlex*, or actual hardware configuration: the VTSSs, ACSs, real and virtual drives and volumes defined by the shared remote CDS. The SERVER commands, which point to the TapePlex name, complete the connection by specifying the paths to the HTTP server on the server hosts running the HSC/VTCS.

For full details on implementing SMC client/server, see the *SMC Configuration and Administration Guide*. In the next two sections, you'll see a high-level view of what it takes to convert to a client/server model. But first, a couple of words about client/server that are specific to VTCS:

- Ensure that both the VTCS CONFIG and the MVS HCD on the hosts where you plan to run HSC define **all** VTDs. VTDs do not need to be online to the server host, but the server host must have a CHPID to all VTSSs accessed by client systems for communication with the VTSS.
- In general client server in JES3 has the same requirements as in JES2. However, **note that** in JES3 the allocation is done by a JES3 GLOBAL host while message processing may be done by the JES3 LOCAL host. The HSC designated as the server for **both** the GLOBAL and LOCAL hosts must have all device addresses defined in the HSC LIBGEN and VTCS CONFIG, and all virtual device addresses must be defined in the HCD of the server host.

Back to the conversion procedures. It turns out that there are two general existing configurations, so we provide a conversion method for each:

- "Converting HSC/SMC on Multiple LPARs to SMC Client/Server" on page 153 is just what it sounds like. You have both HSC and SMC installed on multiple LPARs, and you want to move to the advantages of client/server.
- "Converting from MVS/CSC and LibraryStation" on page 154 is the case where you're running MVS/CSC on one or more client hosts and LibraryStation/HSC/VTCS on a server host, and you want to convert to the SMC Client/Server Feature.

Converting HSC/SMC on Multiple LPARs to SMC Client/Server

t To convert from HSC/SMC on multiple LPARs to SMC client/server:

1. **Choose one of your server hosts currently running HSC/SMC, and bring up the HTTP server on it.**
2. **Choose a "pilot" client host to define TAPEPLEX and SERVER statements for a Primary Server.**

   Define HSC on the pilot client host as the local subsystem for the TapePlex. For example:

   ```
   TAPEPLEX NAME(SHRLIB) LOCSUB(HSC0)
   SERVER NAME(REMPATH1) TAPEPLEX(SHRLIB) HOST(MVSA) PORT(8888)
   ```

3. **Start SMC on the client host, then disable the local subsystem and attempt to communicate with the Primary Server:**

   ```
   TAPEPLEX NAME(SHRLIB) LOCDIS
   RESYNC
   ```

   The RESYNC command output should show that the client is successfully communicating with the remote TapePlex.

4. **Next, start the HTTP server on a second HSC server host.**
5. **Define the second server as a backup:**

   ```
   SERVER NAME(REMPATH2) TAPEPLEX(SHRLIB) HOST(MVSB) PORT(8888)
   ```

6. **Disable the Primary Server and attempt to communicate with the Backup Server:**

   ```
   SERVER NAME(REMPATH1) DISABLE
   RESYNC
   ```

   The RESYNC command output should show that the client is successfully communicating with the remote TapePlex.

7. **Re-enable the first server and switch back to it:**

   ```
   SERVER NAME(REMPATH1) ENABLE
   RESYNC RESTART
   ```

8. **Repeat** Step 1 **through** Step 7 **on additional client hosts until you reach your desired final configuration.**

Converting from MVS/CSC and LibraryStation

To convert from MVS/CSC and LibraryStation/HSC to SMC client/server:

1. **Create TAPEPLEX statements for all your MVS/CSC hosts.**

   Although SMC by default will detect the active MVS/CSC systems and communicate with them, you need define the TapePlexes explicitly to use SMC Client/Server, for example:

   ```
   TAPEPLEX NAME(SHRLIB) LOCSUB(HSC1)
   ```

2. **Bring up the HTTP server on the host(s) where you are currently executing LibraryStation/HSC communicating with MVS/CSC.**

3. **Choose one of the hosts currently running MVS/CSC and create a SMC SERVER command to communicate directly with the HTTP server on the server host in Step 2:**

   ```
   SERVER NAME(REMPATH1) TAPEPLEX(SHRLIB) HOST(MVSA) PORT(8888)
   ```

   The SERVER statement points to the TapePlex defined in Step 1.

4. **Enter RESYNC to attempt to communicate with the server defined in Step 3.**

   ```
   RESYNC
   ```

   The RESYNC command output should show that the client is successfully communicating with the remote TapePlex.

5. **If you have multiple LibraryStation servers defined, repeat Step 1 through Step 4 to define additional TapePlexes.**

6. **After your SMC client/server system is operational, you can stop the MVS/CSC executing on each client host.**

7. **After all MVS/CSC systems are no longer active, stop the LibraryStation server(s) on the HSC host(s):**

   ```
   LS STOP
   ```

Connecting Non-MVS/CSC Clients to VSM

The following procedure tells how to connect non-MVS/CSC 4.0 and above clients to VSM and define LibraryStation subpools that contain VTVs. Contact StorageTek Software Support for information on the supported clients.

To connect non-MVS/CSC clients to VSM:

1. **Define a virtual ACS using the LibraryStation VIRTACS statement.**

   For example, to define virtual ACSs 126 and 125 and map them to VTSSs VTSS01 and VTSS02, create the following VIRTACS statements:

```
VIRTACS ID(126) VTSSNAME(VTSS01)
VIRTACS ID(125) VTSSNAME(VTSS02)
```

2. **Define an HSC subpool that contains VTVs.**

Note – If you define this subpool as ANSI label, VTCS will mount scratch VTV as ANSI labeled tapes from this subpool.

   For more information, see *HSC System Programmer's Guide for MVS.*

3. **Define a LibraryStation subpool that corresponds to the HSC subpool in Step 2 using the LibraryStation SPNUM statement.**

   For example, create the following SPNUM statement to:

   - Define VTV subpool 7 that corresponds to HSC subpool LSVIRT1
   - Pass Management Class MGMTCLS7 to VSM when a VTV is mounted
   - Specify VTSSs VTSS01 and VTSS02 are used to satisfy VTV mounts
   - Restrict VTV mount requests to the client at IP address 129.80.57.16

```
SPNUM NUM(07) SPNAME(LSVIRT1) VIRT(YES) MGMT(MGMTCLS7)
VTSSL(VTSS01,VTSS02) IPADDR(129.80.57.16)
```

Updating the Tape Management System

To update your tape management system (such as CA-1, CA-Dynam/TLMS, and DFSMSrmm), do the following:

- Add volser ranges for VTVs to your tape management system. Ensure that you do *not* assign vault codes to VTVs.
- Access to the MVCs via an RTD bypasses the MVS intercepts put in place by the tape management system so that it does *not* record within its database any access to the MVCs by VSM and does *not* automatically provide protection against inadvertent overwrites of non-expired data on MVCs. Therefore, if you choose to define MVCs to the tape management system, StorageTek **strongly recommends** that you define them as non-scratch, non-expiring volumes. **Also note** that `UNITATTR` with `MODEL(IGNORE)` cannot be used with RTDs.
- The tape management system requires an entry in the MVS Subsystem Name Table; this entry must precede the entry for HSC. For more information, see *HSC Configuration Guide.*

Note – If you are using AutoMedia for MVS, ensure that VTVs are defined as virtual volumes to direct AutoMedia to bypass DSN checking, which allows AutoMedia to recall, mount, and reuse non-resident scratch VTVs.

Caution – Note the following:

- VTCS has an automatic interface to notify RMM when a VTV becomes scratch, but RMM does not notify VTCS when you unscratch a VTV by changing the CV status to non-scratch. If this is done, you must also run the HSC SLUADMIN utility to unscratch the VTV for VTCS. Otherwise, you may encounter a mount failure when VTCS attempts to select the VTV to service a scratch mount request.
- RMM (DFSMS/RMM) has additional integrity checks at mount time to ensure that the correct volume has been mounted. Because VTCS has features and optimizations that sometimes present a new initialized version of a VTV rather than the current copy of a VTV, it is necessary for VTCS to override these RMM integrity checks. VTCS does these overrides via the LISTVOLUME and CHANGEVOLUME API calls to update the RMM database. You must therefore ensure that HSC has been given the appropriate security access to the RMM API. For more information, see your RMM documentation.

Defining VSM Security

The following sections tell how to define security for VSM:

- "Defining MVC Pool Volser Authority" on page 157
- "Defining VTCS Command Authority" on page 158

Defining MVC Pool Volser Authority

When VSM needs to mount an MVC and to write to an MVC, a SAF query is issued to verify that the HSC user (see "Defining A Security System User ID for HSC, SMC, and VTCS" on page 86) has UPDATE authority for the MVC. The SAF query is issued on behalf of HSC and passed to the system security product (such as RACF, CA-ACF2, or CA-Top Secret).

VSM requires UPDATE authority for the volsers in the MVC pool. All other users should have an access of NONE for these volsers. Similarly, VSM should not have UPDATE authority for any volsers that are not in the MVC pool. See the documentation for your security product for procedures to add the appropriate TAPEVOL security for VSM. TABLE 9-1 summarizes these definitions.

TABLE 9-1 Security Class, Resource Class, and Access Values for MVC Pool Volser Authority

Class	Resource Name	Recommended User Access Levels
TAPEVOL	MVC Pool Volume Serials	UPDATE - allows VSM to write on MVC

FIGURE 9-4 shows an example of a RACF profile and permissions commands to give the user ID VSM8HSC update access to MVC volser CVC024.

```
*******************************************************************************
* Define a profile in the TAPEVOL class for MVC CVC024 *
*******************************************************************************
RDEFINE TAPEVOL CVC024 UACC(NONE)
*******************************************************************************
************ Allow user ID VSM8HSC update access to MVC CVC024 *
*******************************************************************************
**********PERMIT CVC024 CLASS(TAPEVOL) ACCESS(UPDATE) ID(VSM8HSC)
*******************************************************************************
```

FIGURE 9-4 Example RACF MVC volser access file

Caution – Note the following:

- To ensure that MVCs are not accidentally overwritten, for each MVC volser, you must update your TAPEVOL security as described above and your tape management system. For more information, see "Updating the Tape Management System" on page 156.
- You must also run the HSC UNSCratch Utility to unscratch any current scratch cartridges in the MVC range. For more information, see *HSC System Programmer's Guide for MVS*.
- Depending on the default settings of your security system, VSM may not be able to mount and to write to MVCs until you have defined a security system user ID for HSC and TAPEVOL profiles for the MVCs.
- If you add new ranges of MVCs to your VSM system, remember to update the TAPEVOL profiles to include the new ranges.

Defining VTCS Command Authority

If HSC user exit SLSUX15 sets a return code of UX15CHKA, the exit issues a command authorization request to the system security product. FIGURE 9-5 shows an example of RACF profile and permissions commands to give user SAM15 access to all VTCS commands (those with a VT command prefix). Note that you can only give a user access to *all* VTCS commands; you cannot give access to individual VTCS commands. For more information, see *HSC System Programmer's Guide for MVS*.

```
*******************************************************************************
* Define a profile in the OPERCMDS class for all VTCS commands *
*******************************************************************************
RDEFINE OPERCMDS subsysname.VT UACC (NONE)
*******************************************************************************
************ Allow user SAM15 update access to all VTCS commands
*******************************************************************************
**********PERMIT subsysname.VT CLASS(OPERCMDS) ID(SAM15) ACCESS (UPDATE)
*******************************************************************************
```

FIGURE 9-5 Example RACF VTCS command authorization file

Updating HSM

HSM users that have mixed devices that were “logically” defined as the same type of device, such as 3490E, but are “physically” different, such as T9940, virtual (VTD), or 9490 must set the following parameter in HSM:

```
SETSYS RECYCLEINPUTDEALLOCFREQUENCY(MIGRATION(1))
```

By setting this parameter, when HSM is “recycling”, it will deallocate the input drive after it processes each input tape. This is required where the tapes being recycled are “physically” mixed as described above.

If you do not set this parameter, it is possible that you could allocate a 9490 transport for the first tape, then if the second tape was virtual (VTV) or STK2P, the job would fail due to media incompatibility. That is, you could not physically mount the second tape (virtual or STK2P media) on the 9490 drive that had been allocated for the first tape.

Routing Data Sets to VSM

You recorded your VSM candidate data sets in Table 34. on page 121. To route these data sets to VSM, use any of the techniques described in the following sections:

- "The StorageTek DFSMS Interface" on page 161
- "SMC TAPEREQ Statements" on page 162
- "HSC User Exits" on page 162
- "MVS/CSC User Exits" on page 163

Note – In addition, you can also change your JCL to direct data sets to VSM although StorageTek does not recommend this method.

Caution – StorageTek strongly recommends that you create VTVs as Standard Label (SL) tapes, otherwise unpredictable results can occur.
Also note that VSM does not provide readonly protection for VTVs. That is, even if MVS requests a mount READONLY of a VTV, VSM mounts the VTV as READ/WRITE.

The StorageTek DFSMS Interface

You can use the StorageTek DFSMS interface to route data sets to VSM via Unit Name substitution. These interfaces use names that you code in SMS routines to drive SMC allocation processing. For more information about the StorageTek DFSMS interface, see *SMC Configuration and Administration Guide*.

Use this interface for VSM as follows:

1. Define a Storage Group and Storage Class for data sets to be routed to VSM. The Storage Group must be identical in name to an esoteric that represents VTDs. Use an esoteric name and the data set selections that you recorded in TABLE A-1 on page 165.

   For more information on defining and using VSM esoterics for the StorageTek DFSMS interface, see "VSM Esoterics and Esoteric Substitution" on page 14.

2. Code Storage Class and Storage Group ACS routines to assign the correct Storage Class and Storage Groups to virtual tape data when &ACSENVIR = "STKTAP1". For more information, see *HSC System Programmer's Guide for MVS*.

3. If you want to pass one or more Management Classes to VTCS, create a Management Class that is identical in name to a Management Class you defined on a MGMTclas statement. Then add code to the Management Class ACS routine to assign a VTCS Management Class to selected data sets when &ACSENVIR = "STKTAP1".

Note – The Management Class you define on the MGMTclas statement can specify a VSM Storage Class, which is **not** the same as the ACS Storage Class you define in Step 2

---

4. Set the SMC ALLOCDEF or ALLOCJOB command SMS option to ON so the StorageTek DFSMS interface drives SMS ACS routines.

Note – If you specify a Management Class on a TAPEREQ statement and an SMS routine, the Management Class on the SMS routine takes precedence.

For more information about the StorageTek DFSMS interface, see *SMC Configuration and Administration Guide*.

SMC TAPEREQ Statements

To route data sets to VSM, you can create an SMC TAPEREQ statement. To route data sets to VSM with TAPEREQ statements, do one of the following:

- Specify Virtual on the MEDia, MODel, or RECtech parameter. If you specify Virtual, VSM selects an available VTD in your system and routes the job to that VTD.

  In a multi-VTSS environment, therefore, specifying Virtual does *not* direct the VTD allocation to a specific VTSS, but lets the allocation occur in any VTSS in the configuration.
- Specify an esoteric that represents VTDs on the ESOTeric parameter. You recorded your VSM esoterics in TABLE A-1 on page 165.

  For VSM, esoteric definition and substitution is different in JES2 and JES3. For more information on defining and using VSM esoterics for TAPEREQ statements, see "VSM Esoterics and Esoteric Substitution" on page 14.
- Specify a scratch subpool that contains virtual volumes.

Caution – Multiple TAPEREQ statements that specify the same or overlapping selection criteria (such as jobname, stepname, or data set) can cause undesirable results (such as assignment of MEDia Virtual *and* an esoteric).

HSC User Exits

To route data sets to VSM with HSC User Exits, do one of the following:

- Use return code UX02VIRT (32) in register 15 in HSC User Exit SLSUX02 (JES2) or for SLSUX04 (JES3) use UX04VIRT (24), which you use to control transport allocation for scratch mounts. To satisfy a scratch mount request, return code UX0xVIRT causes VSM to select an available VTD in your system and routes the job to a VTV mounted on that VTD.
- Use esoteric substitution in any of the User Exits that support esoteric substitution. For example, to direct scratch allocation requests to a VTD, specify an esoteric that represents VTDs in the UX02ESO field of SLSUX02 or the UX04ESOT field of SLSUX04.

  For VSM, esoteric definition and substitution is different in JES2 and JES3. For more information on defining and using VSM esoterics for HSC User Exits, see "VSM Esoterics and Esoteric Substitution" on page 14.

For more information about HSC User Exits, see *HSC System Programmer's Guide for MVS*.

MVS/CSC User Exits

To route data sets to VSM with MVS/CSC User Exits, do one of the following:

- MVS/CSC User Exit SCSUX02 (JES2 and JES3 without TAPE setup environments), which you use to control transport allocation for scratch mounts, now supports return code UX02VIRT in register 15. SCSUX04 (JES3 with TAPE setup environment) also supports return code UX04VIRT in register 15. To satisfy a scratch mount request, these return codes cause VSM to select an available VTD in your system and route the data set to a VTV mounted on that VTD.

  In a multi-VTSS environment, therefore, these return codes do not direct the VTD allocation to a specific VTSS, but let the allocation occur in any VTSS in the configuration.

- Use esoteric substitution in any of the User Exits that support esoteric substitution. For example, to direct scratch allocation requests to a VTD, specify an esoteric that represents VTDs in the UX02ESO field of SCSUX02 or the UX04ESOT field of SCSUX04.

  For VSM, esoteric definition and substitution is different in JES2 and JES3. For more information on defining and using VSM esoterics for MVS/CSC User Exits, see "VSM Esoterics and Esoteric Substitution" on page 14.

For more information about MVS/CSC User Exits, see MVS/CSC System Programmer's Guide.

Restarting NCS/VTCS

Ensure that you modified the HSC startup procedure as described in "Modifying the HSC Startup Procedure to include the VTCS 6.2.0 LINKLIB" on page 102. HSC initialization automatically starts VTCS, and HSC termination automatically terminates VTCS.

To complete the NCS reconfiguration, start one or more of the following on all hosts that are using the new CDS data sets:

- SMC, for more information, see *SMC Configuration and Administration Guide*.
- LibraryStation; for more information, see *LibraryStation Operator and System Programmer's Guide*.
- MVS/CSC; for more information, see *MVS/CSC Operator's Guide*.

APPENDIX **A**

VSM Configuration Record

TABLE A-1 lists the installation and configuration values you determined. It also provides a record of your site's VSM configuration, which can help you and StorageTek service troubleshoot problems with your VSM system.

TABLE A-1 VSM Configuration Record

Configuration Value	**Planning Information**	**Your Site's Selection**
VTSS names	"VTSS Names" on page 12	
VTD unit addresses	"VTD Unit Addresses" on page 13	
VSM esoteric names	"VSM Esoterics and Esoteric Substitution" on page 14	
VTV volsers (all)	"VTV Definitions" on page 19	
VTV volsers (scratch pool ranges)		
RTD unit addresses	"RTD Definitions" on page 21	
MVC volsers - VOLATTR statements and CONFIG	"MVC Definitions" on page 22	
MVC volsers - MVCPool statements		
CDS VTCS Level	"CDS VTCS Level" on page 32	
MVS Coupling Facility Structure	"Storing VTCS Locks in a Coupling Facility (Optional)" on page 35	
Virtual ACS IDs	"Virtual ACS IDs" on page 39	
HSC CDS DASD size	"HSC CDS DASD Space" on page 49	
Tape management system DASD size	"Tape Management System DASD Space" on page 49	
VSM candidate data sets	"VSM Candidate Data Sets" on page 49	
HSC COMMpath METHod value	"HSC COMMPath METHod Value" on page 49	

TABLE A-1 VSM Configuration Record

	VTSS Policies	
AMT settings	"AMT Settings" on page 53	
VTV Page Size	"VTV Page Size" on page 59	
Maximum and minimum concurrent automatic migration, immediate migration, and migrate-to-threshold tasks (CONFIG MAXMIG/MINMIG)	"Maximum and Minimum Concurrent Migration Tasks" on page 67	
	VTV Policies	
Maximum VTVs per MVC (CONFIG MAXMVC)	"Maximum VTVs per MVC" on page 70	
Hosts disabled from migration, consolidation, and export by VTV or Management Class (CONFIG NOMIGRAT)	"Hosts Disabled from Reclamation" on page 78	
Recall VTVs with Read Data Checks	"Recall VTVs with Read Data Checks" on page 71	
	MVC Space Reclamation Policies	
MVC fragmented space threshold (CONFIG THRESHld)	"MVC Fragmented Space Threshold - Determines MVC Eligibility for Reclamation" on page 73	
Free MVCs threshold (CONFIG MVCFREE)	"Free MVCs Threshold - Starts Automatic Space Reclamation" on page 74	
Eligible/Total MVCs threshold (CONFIG START)	"Eligible/Total MVCs Threshold - Starts Automatic Space Reclamation" on page 76	
Maximum MVCs processed per reclaim (CONFIG MAXMVC)	"Maximum MVCs Processed Per Reclaim" on page 77	
Maximum MVCs Concurrently Processed for Reclamation and Drain	"Maximum MVCs Processed Per Reclaim" on page 77	
Hosts disabled from reclamation (CONFIG NORECLAM)	"Hosts Disabled from Reclamation" on page 78	
MVC retain interval	"MVC Retain Interval" on page 79	

APPENDIX **B**

VSM4 ESCON Configuration

The newest generation VTSS is the VSM4, which provides the following advantages over its predecessors:

- Enhanced connectivity options.
- Greater throughput.
- Greater VTSS capacity.
- 4x the number of VTDs and 3x the maximum number of VTVs per VTSS.
- Improved reliability and serviceability.

TABLE B-1 summarizes the VSM3 to VSM4 ESCON enhancements that you see from a software and system configuration perspective.

TABLE B-1 VSM3 to VSM4 Comparison: Software and System Configuration ESCON Enhancements

Product Feature	VSM3	VSM4
ESCON Interfaces	16 **total** where: ▪ 2 to 14 can be host channels ▪ 2 to 8 can be Nearlink/CLINK connections	32 **total** where: ▪ 2 to 28 can be host channels ▪ 2 to 16 can be Nearlink/CLINK connections **Note**: VSM4s are shipped with 16 ports enabled.With the 16 ports enabled option, **only** the top port on each CIP is enabled (Port 0 or Port 2). 32 ports enabled is an **optional, separately priced feature** that is activated via microcode diskette. On a VSM4 with 32 ports enabled, each ICE3 ESCON interface card contains two pairs of ESCON ports. Each pair is controlled by its own Channel Interface Processor (CIP). Each CIP switches between the two ports, so that only one port can transfer data at a time.
Maximum Logical Paths	128	16 per port for the 16 port standard configuration = 256 logical paths 16 per port for the 32 port optional configuration= 512 logical paths **Note**: VSM4 provides a theoretical maximum of 512 logical paths per VTSS, but you cannot allocate all 512 logical paths for host-to-VTSS connections.
VTDs per VTSS	64	256
Maximum resident VTVs per VTSS	100,000	300,000

VSM4 with 32 Ports

For the 32 port option, the 8 ICE3 cards have four ESCON ports per card as shown in FIGURE B-1.

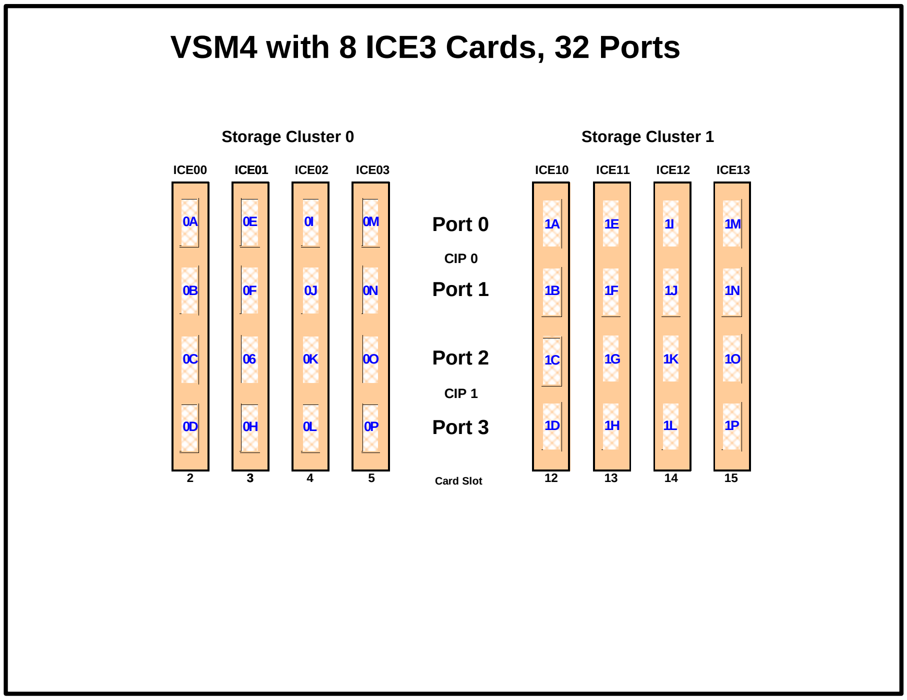


FIGURE B-1 VSM4 with 32 Ports

Note – In FIGURE B-1 and all the other figures in this appendix, the ports are shown with their channel interface identifiers for **enabled** ports (32 in FIGURE B-1). These channel interface identifiers are the values that are required for the CHANIF values that you code for the CONFIG utility. Each value is two characters in length and has a value from 0A to 1P. The first digit is the VTSS cluster ID (valid values are 0 or 1). The second digit is the group or adapter ID (valid values are A to P).

In FIGURE B-1, note the following:

- Each ICE3 card contains two pairs of ESCON ports. Each pair is controlled by its own Channel Interface Processor (CIP). Each CIP switches between the two ports, so that **only one port** can transfer data at a time.
- For a VSM4, each CIP can operate with only *one* of two “personalities”, which is set at the VTSS LOP:

- *Host Mode*. In Host Mode, either or both ESCON ports can connect to host CPU channels, including via ESCON Director(s) or channel extenders. Ports of a CIP in Host mode **cannot** connect to RTDs or to Secondary VTSSs via CLINKS. Note, however, that Secondary VTSSs must have an ESCON port in Host Mode to connect via a CLINK *from* an ESCON port in Nearlink Mode in a Primary VTSS.

  Also note that you can have two physical paths from the same LPAR to the same CIP, as long as the two physical paths address different (not overlapping) logical control units. For example, a single host LPAR can address logical control units 0-7 on one CIP port, and 8-F on the other CIP port of the same CIP.

- *Nearlink Mode*. In Nearlink Mode, either or both ESCON ports can connect to an RTD or via a CLINK to a Secondary VTSS. Ports of a CIP in Nearlink mode **cannot** connect to host CPU channels.You can set a **maximum** of 8 CIPs to Nearlink Mode, and here's the important fine print: only **one** Nearlink port per CIP is active at one time. What are Best Practices for optimizing port operations? See TABLE B-2...

TABLE B-2 Optimizing VSM4 Port Operations

Configuration - Two Ports on a CIP	Best Practices
Two CLINKs	**Don't use**....because only one port can be active at a time. If you're doing Clustered VTSS, you want all CLINK connections to be active all the time.
CLINK and RTD	**An advantage** in Degraded Cluster Mode. You normally have fewer RTDs on the Primary VTSS because the Secondary is doing most of the migrations. If you have an offline RTD on the same CIP as an active CLINK, if the Secondary fails you can vary the CLINK offline and bring the RTD online to handle more workload on the Primary. **Note that** while the CLINK is active, the RTD is unavailable and is reported as suspended via DISPLAY RTD.
Two RTDs	**An advantage** for the following: **Optimize use of local and remote RTDs**. During busy shifts, use only the local RTD on the CIP. During quiet periods, switch to the remote RTD for deep archive and DR work. **Optimize use of different drive technologies**. As described in the previous bullet, use a T9840 as a local RTD, then switch to a T9940 for deep archive. You can also use this feature to migrate from older drive technology (such as 9490) to newer technology (such as 9840). Use Management and Storage Classes to read in data from older media, then switch to the newer technology drive to place data on new media. This technique effectively gives you greater physical connectivity to different drive technologies without incurring the overhead of full time, real time ESCON connections to each drive type. **Note that** Because of the "only one active" rule, if an RTD on one port is migrating or recalling a VTV, the RTD on the second port cannot be accessed until the operation on the first port completes (the RTD on the second port is in "suspend" mode, as shown by the D RTD command/utility). Best Practices suggests, therefore, that RTDs that **must** be active simultaneously should connect to different CIPs. One more piece of fine print: If you have two RTDs on a CIP, you can't share them between VTSSs.

- On a VSM4 with 32 ports enabled, you have a theoretical maximum of 512 logical paths on the VSM4. However, you must have some RTD connections so you cannot allocate all 512 logical paths for host-to-VTSS connections. What's the minimum number of RTDs? Well, it's like this: (1) CONFIG will not allow fewer than 2 RTDS per VTSS. (2) CONFIG cannot check device type, but StorageTek strongly recommends at least two RTDS of each device type in each ACS to which the VTSS is attached....otherwise, you can seriously compromise error recovery, and also impact the efficiency of space reclamation. If you had only two RTDs, Best Practices would suggest that you connect them to different ICE3 cards...and once you've done that, you've effectively used up 4 Nearlink ports due to the "CIP personality" nature of the ICE3 card. Therefore, in an 8 ICE card configuration, this leaves 28 available ports for host-to-VTSS ESCON channel connections, which equals a maximum of 16 x 28 or 448 logical paths. For more information, see "Logical Paths for VSM 4 with 32 Ports" on page 186.

- A host logical path is the communication path between a host and all of the 256 VTDs within the VSM4. TABLE B-3 summarizes the configuration options and maximum host logical paths for a VSM4 with 32 enabled ports.

TABLE B-3 VSM4 Configuration Options - 32 Ports

Host CIPs	Maximum Host Connections	Nearlink CIPs	Max Nearlink Connections	Maximum Host Logical Paths
8	16	8	16	256
9	18	7	14	288
10	20	6	12	320
11	22	5	10	352
12	24	4	8	384
14	28	2	4	448

- In HCD:
  - From a single MVS host, you can only define 8 channels (CHPIDs) running to a single control unit (single VSM4). **Also note that** ICE3 cards **cannot** have 2 paths from the same LPAR connected to two ports with a common CIP.
  - You use the CNTLUNIT statement to define each VSM4 as 16 3490 images.
  - You use the IODEVICE statement to define the 16 VTDs that are associated with each 3490 image.

VSM4 Configuration Examples - 32 Ports

For VSM4s with 32 ports, let's look at two examples of port configurations:

- "VSM4 Configuration Example: 16 Host Ports, 16 RTD Ports" on page 173
- ""VSM 4 Configuration Example: 20 Host Ports, 12 RTD Ports" on page 175

For a VSM4 host gen example, see "IOCP Example for Single MVS Host Connected to a VSM4 Via ESCON Directors" on page 184.

VSM4 Configuration Example: 16 Host Ports, 16 RTD Ports

FIGURE B-2 shows CONFIG channel interface identifiers of 16 for hosts, 16 for RTDs for a VSM4.

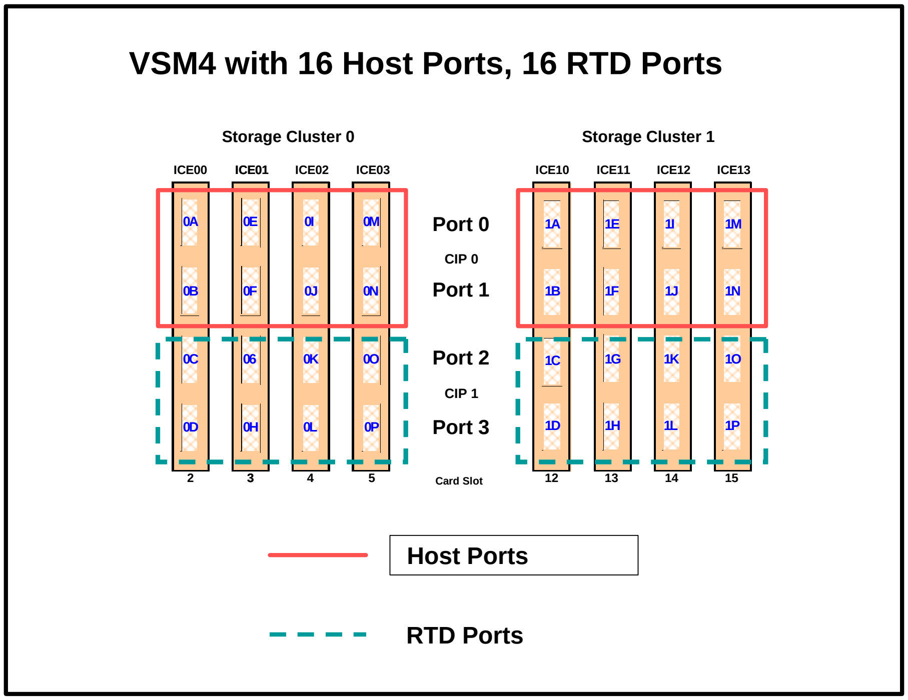


FIGURE B-2 VSM4 with 16 Host Ports, 16 RTD Ports

CONFIG Example for VSM4 with 16 Host Ports, 16 RTD Ports

FIGURE B-3 shows example CONFIG JCL to define the VSM4 configuration shown in FIGURE B-2 on page 173.

```
//CREATECF EXEC PGM=SWSADMIN,PARM='MIXED'
//STEPLIB DD DSN=hlq.SLSLINK,DISP=SHR
//SLSCNTL DD DSN=FEDB.VSMLMULT.DBASEPRM,DISP=SHR
//SLSCNTL2 DD DSN=FEDB.VSMLMULT.DBASESEC,DISP=SHR
//SLSSTBY DD DSN=FEDB.VSMLMULT.DBASETBY,DISP=SHR
//SLSPRINT DD   SYSOUT=*
//SLSIN DD   *
 CONFIG
 GLOBALMAXVTV=32000MVCFREE=40
 RECLAIMTHRESHLD=70MAXMVC=40  START=35
 VTVVOL LOW=905000 HIGH=999999 SCRATCH
 VTVVOL LOW=C00000 HIGH=C25000 SCRATCH
 VTVVOL LOW=RMM000 HIGH=RMM020 SCRATCH
 MVCVOL LOW=N25980 HIGH=N25989
 MVCVOL LOW=N35000 HIGH=N35999
 VTSS NAME=VSM401 LOW=70 HIGH=80 MAXMIG=8 RETAIN=5
 RTD  NAME=VSM42A00 DEVNO=2A00 CHANIF=0C
 RTD  NAME=VSM42A01 DEVNO=2A01 CHANIF=0D
 RTD  NAME=VSM42A02 DEVNO=2A02 CHANIF=0G
 RTD  NAME=VSM42A03 DEVNO=2A03 CHANIF=0H
 RTD  NAME=VSM42A04 DEVNO=2A04 CHANIF=0K
 RTD  NAME=VSM42A05 DEVNO=2A05 CHANIF=0L
 RTD  NAME=VSM42A06 DEVNO=2A06 CHANIF=0O
 RTD  NAME=VSM42A07 DEVNO=2A07 CHANIF=0P
 RTD  NAME=VSM42A08 DEVNO=2A08 CHANIF=1C
 RTD  NAME=VSM42A09 DEVNO=2A09 CHANIF=1D
 RTD  NAME=VSM42A0A DEVNO=2A0A CHANIF=1G
 RTD  NAME=VSM42A0B DEVNO=2A0B CHANIF=1H
 RTD  NAME=VSM42A0C DEVNO=2A0C CHANIF=1K
 RTD  NAME=VSM42A0D DEVNO=2A0D CHANIF=1L
 RTD  NAME=VSM42A0E DEVNO=2A0E CHANIF=1O
 RTD  NAME=VSM42A0F DEVNO=2A0F CHANIF=1P
 VTD LOW=9900 HIGH=99FF
```

FIGURE B-3 CONFIG example: VSM4 with 16 Host Ports, 16 RTD Ports

VSM 4 Configuration Example: 20 Host Ports, 12 RTD Ports

FIGURE B-4 shows port assignments of 20 for hosts, 12 for RTDs for a VSM4.

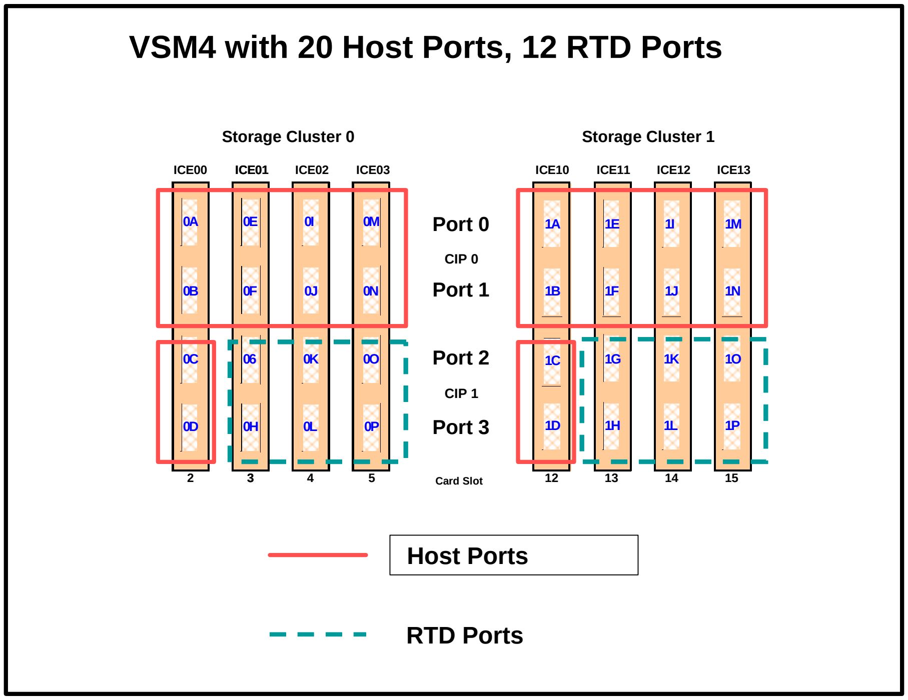


FIGURE B-4 VSM4 with 20 Host Ports, 12 RTD Ports

CONFIG Example for VSM4 with 20 Host Ports, 12 RTD Ports

FIGURE B-5 shows example CONFIG JCL to define the VSM4 configuration shown in FIGURE B-4 on page 175.

```
//CREATECF EXEC PGM=SWSADMIN,PARM='MIXED'
//STEPLIB DD DSN=hlq.SLSLINK,DISP=SHR
//SLSCNTL DD DSN=FEDB.VSMLMULT.DBASEPRM,DISP=SHR
//SLSCNTL2 DD DSN=FEDB.VSMLMULT.DBASESEC,DISP=SHR
//SLSSTBY DD DSN=FEDB.VSMLMULT.DBASETBY,DISP=SHR
//SLSPRINT DD   SYSOUT=*
//SLSIN DD   *
 CONFIG
 GLOBALMAXVTV=32000 MVCFREE=40
 RECLAIMTHRESHLD=70 MAXMVC=40  START=35
 VTVVOL LOW=905000 HIGH=999999 SCRATCH
 VTVVOL LOW=C00000 HIGH=C25000 SCRATCH
 VTVVOL LOW=RMM000 HIGH=RMM020 SCRATCH
 MVCVOL LOW=N25980 HIGH=N25989
 MVCVOL LOW=N35000 HIGH=N35999
 VTSS NAME=VSM401 LOW=70 HIGH=80 MAXMIG=6 RETAIN=5
 RTD  NAME=VSM42A00 DEVNO=2A00 CHANIF=0G
 RTD  NAME=VSM42A01 DEVNO=2A01 CHANIF=0H
 RTD  NAME=VSM42A02 DEVNO=2A02 CHANIF=0K
 RTD  NAME=VSM42A03 DEVNO=2A03 CHANIF=0L
 RTD  NAME=VSM42A04 DEVNO=2A04 CHANIF=0O
 RTD  NAME=VSM42A05 DEVNO=2A05 CHANIF=0P
 RTD  NAME=VSM42A06 DEVNO=2A06 CHANIF=1G
 RTD  NAME=VSM42A07 DEVNO=2A07 CHANIF=1H
 RTD  NAME=VSM42A08 DEVNO=2A08 CHANIF=1K
 RTD  NAME=VSM42A09 DEVNO=2A09 CHANIF=1L
 RTD  NAME=VSM42A0A DEVNO=2A0A CHANIF=1O
 RTD  NAME=VSM42A0B DEVNO=2A0B CHANIF=1P
 VTD LOW=9900 HIGH=99FF
```

FIGURE B-5 CONFIG example: VSM4 with 20 Host Ports, 12 RTD Ports

VSM4 with 16 Ports

For the 16 port option, the 8 ICE3 cards have two ESCON ports per card as shown in FIGURE B-6.

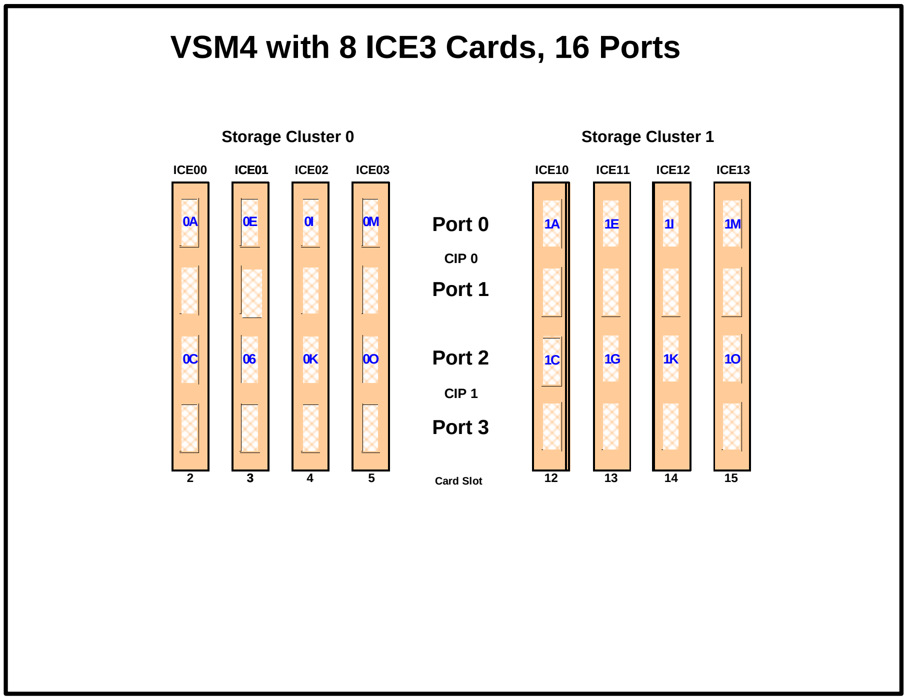


FIGURE B-6 VSM4 with 16 Ports

Note – In FIGURE B-6 and all the other figures in this appendix, the ports are shown with their channel interface identifiers for **enabled** ports (16 in FIGURE B-6). These channel interface identifiers are the values that are required for the CHANIF values that you code for the CONFIG utility. Each value is two characters in length and has a value from 0A to 1P. The first digit is the VTSS cluster ID (valid values are 0 or 1). The second digit is the group or adapter ID (valid values are A to P).

In FIGURE B-6, note the following:

- Each ICE3 card has two CIPs with **a single port enabled** on each CIP. As with the 32 port option, each CIP can operate with only *one* of two “personalities”, which is set at the VTSS LOP:
  - *Host Mode*. In Host Mode, the single ESCON port can connect to host CPU channels, including via ESCON Director(s) or channel extenders. Ports of a CIP in Host mode **cannot** connect to RTDs or to Secondary VTSSs via CLINKS. Note, however, that Secondary VTSSs must have an ESCON port in Host Mode to connect via a CLINK *from* an ESCON port in Nearlink Mode in a Primary VTSS.

- *Nearlink Mode*. In Nearlink Mode, the single ESCON port can connect to an RTD or via a CLINK to a Secondary VTSS. Ports of a CIP in Nearlink mode **cannot** connect to host CPU channels.

  You can set a **maximum** of 8 CIPs to Nearlink Mode. Therefore, in a 16 port configuration, the single port on a CIP can be either a CLINK or an RTD connection.

- On a VSM4 with 16 ports enabled, you have a **theoretical** maximum of 256 logical paths on the VSM4. However, you must have *some* RTD connections so you cannot allocate *all* 256 logical paths for host-to-VTSS connections. What's the **minimum** number of RTDs? Well, it's like this: (1) CONFIG will not allow fewer than 2 RTDS per VTSS. (2) CONFIG cannot check device type, but StorageTek s**trongly recommends** at least two RTDS of each device type in each ACS to which the VTSS is attached....otherwise, you can seriously compromise error recovery and also impact the efficiency of space reclamation. If you had only two RTDs, Best Practices would suggest that you connect them to different ICE3 cards...and once you've done that, you've effectively used up 4 Nearlink ports. Therefore, in an 8 ICE card 16 port configuration, this leaves 12 available ports for host-to-VTSS ESCON channel connections, which equals a **maximum** of 16 x 12 or 192 logical paths. For more information, see "Logical Paths for VSM 4 with 32 Ports" on page 186.

- A host logical path is the communication path between a host and all of the 256 VTDs within the VSM4. TABLE B-4 summarizes the configuration options and maximum host logical paths for a VSM4 with 16 enabled ports.

TABLE B-4 VSM4 Configuration Options - 16 Ports

Host CIPs	Maximum Host Connections	Nearlink CIPs	Max Nearlink Connections	Maximum Host Logical Paths
8	8	8	8	128
9	9	7	7	144
10	10	6	6	160
11	11	5	5	176
12	12	4	4	192
14	14	2	2	224

- In HCD:
  - From a single MVS host, you can only define 8 channels (CHPIDs) running to a single control unit (single VSM4). **Also note that** ICE3 cards **cannot** have 2 paths from the same LPAR connected to two ports with a common CIP.
  - You use the CNTLUNIT statement to define each VSM4 as 16 3490 images.
  - You use the IODEVICE statement to define the 16 VTDs that are associated with each 3490 image.

VSM4 Configuration Examples - 16 Ports

For VSM4s with 16 ports, let's look at two examples of port configurations:

- "VSM4 Configuration Example: 8 Host Ports, 8 RTD Ports" on page 180
- "VSM 4 Configuration Example: 10 Host Ports, 6 RTD Ports" on page 182

For a VSM4 host gen example, see"IOCP Example for Single MVS Host Connected to a VSM4 Via ESCON Directors" on page 184.

VSM4 Configuration Example: 8 Host Ports, 8 RTD Ports

FIGURE B-7 shows CONFIG channel interface identifiers of 8 for hosts, 8 for RTDs for an 8 ICE3 card VSM4 with 16 ports.

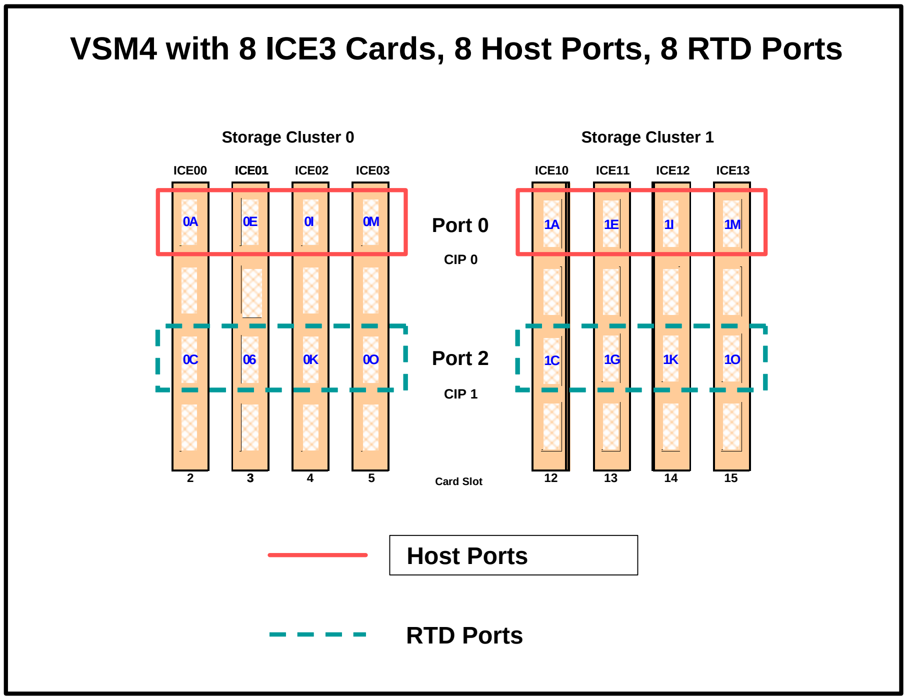


FIGURE B-7 VSM4 with 8 Host Ports, 8 RTD Ports

CONFIG Example for VSM4 with 8 Host Ports, 8 RTD Ports

FIGURE B-8 shows example CONFIG JCL to define the VSM4 configuration shown in FIGURE B-7 on page 180.

```
//CREATECF EXEC PGM=SWSADMIN,PARM='MIXED'
//STEPLIB DD DSN=hlq.SLSLINK,DISP=SHR
//SLSCNTL DD DSN=FEDB.VSMLMULT.DBASEPRM,DISP=SHR
//SLSCNTL2 DD DSN=FEDB.VSMLMULT.DBASESEC,DISP=SHR
//SLSSTBY DD DSN=FEDB.VSMLMULT.DBASETBY,DISP=SHR
//SLSPRINT DD   SYSOUT=*
//SLSIN DD   *
 CONFIG
 GLOBALMAXVTV=32000MVCFREE=40
 RECLAIMTHRESHLD=70MAXMVC=40  START=35
 VTVVOL LOW=905000 HIGH=999999 SCRATCH
 VTVVOL LOW=C00000 HIGH=C25000 SCRATCH
 VTVVOL LOW=RMM000 HIGH=RMM020 SCRATCH
 MVCVOL LOW=N25980 HIGH=N25989
 MVCVOL LOW=N35000 HIGH=N35999
 VTSS NAME=VSM401 LOW=70 HIGH=80 MAXMIG=8 RETAIN=5
 RTD  NAME=VSM42A00 DEVNO=2A00 CHANIF=0C
 RTD  NAME=VSM42A02 DEVNO=2A02 CHANIF=0G
 RTD  NAME=VSM42A04 DEVNO=2A04 CHANIF=0K
 RTD  NAME=VSM42A06 DEVNO=2A06 CHANIF=0O
 RTD  NAME=VSM42A08 DEVNO=2A08 CHANIF=1C
 RTD  NAME=VSM42A0A DEVNO=2A0A CHANIF=1G
 RTD  NAME=VSM42A0C DEVNO=2A0C CHANIF=1K
 RTD  NAME=VSM42A0E DEVNO=2A0E CHANIF=1O
 VTD LOW=9900 HIGH=99FF
```

FIGURE B-8 CONFIG example: VSM4 with 8 Host Ports, 8 RTD Ports

VSM 4 Configuration Example: 10 Host Ports, 6 RTD Ports

FIGURE B-9 shows port assignments of 10 for hosts, 6 for RTDs for a VSM4 with 16 ports.

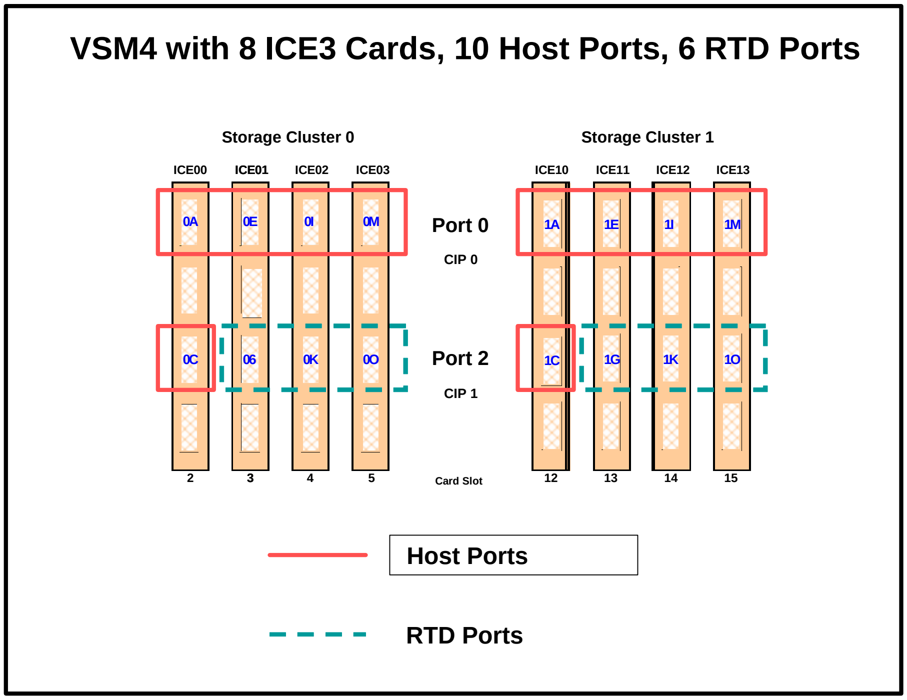


FIGURE B-9 VSM4 with 10 Host Ports, 6 RTD Ports

CONFIG Example for VSM4 with 10 Host Ports, 6 RTD Ports

FIGURE B-10 shows example CONFIG JCL to define the VSM4 configuration shown in FIGURE B-9 on page 182.

```
//CREATECF EXEC PGM=SWSADMIN,PARM='MIXED'
//STEPLIB DD DSN=hlq.SLSLINK,DISP=SHR
//SLSCNTL DD DSN=FEDB.VSMLMULT.DBASEPRM,DISP=SHR
//SLSCNTL2 DD DSN=FEDB.VSMLMULT.DBASESEC,DISP=SHR
//SLSSTBY DD DSN=FEDB.VSMLMULT.DBASETBY,DISP=SHR
//SLSPRINT DD   SYSOUT=*
//SLSIN DD   *
 CONFIG
 GLOBALMAXVTV=32000MVCFREE=40
 RECLAIMTHRESHLD=70MAXMVC=40  START=35
 VTVVOL LOW=905000 HIGH=999999 SCRATCH
 VTVVOL LOW=C00000 HIGH=C25000 SCRATCH
 VTVVOL LOW=RMM000 HIGH=RMM020 SCRATCH
 MVCVOL LOW=N25980 HIGH=N25989
 MVCVOL LOW=N35000 HIGH=N35999
 VTSS NAME=VSM401 LOW=70 HIGH=80 MAXMIG=8 RETAIN=5
 RTD  NAME=VSM42A02 DEVNO=2A02 CHANIF=0G
 RTD  NAME=VSM42A04 DEVNO=2A04 CHANIF=0K
 RTD  NAME=VSM42A06 DEVNO=2A06 CHANIF=0O
 RTD  NAME=VSM42A0A DEVNO=2A0A CHANIF=1G
 RTD  NAME=VSM42A0C DEVNO=2A0C CHANIF=1K
 RTD  NAME=VSM42A0E DEVNO=2A0E CHANIF=1O
 VTD LOW=9900 HIGH=99FF
```

FIGURE B-10 CONFIG example: VSM4 with 10 Host Ports, 6 RTD Ports

IOCP Example for Single MVS Host Connected to a VSM4 Via ESCON Directors

FIGURE B-11 shows a configuration diagram for a single MVS host connected to a VSM4 via ESCON Directors, and FIGURE B-12 on page 185 shows example IOCP statements for this configuration. **Note that**:

- From MVSA, you define 8 CHPIDs, with each path switched in the ESCON Director, for a total of 8 channels running to the VSM4.
- You code 16 CNTLUNIT statements to define the VSM4 as 16 3490 images.
- You code IODEVICE statement to define the 16 VTDs that are associated with each 3490 image.

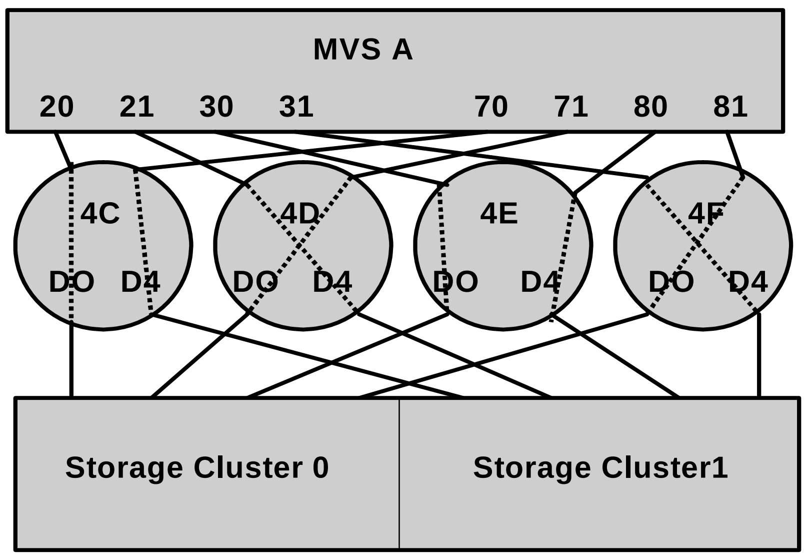


FIGURE B-11 Configuration Diagram: Single MVS Host Connected to a VSM4 via ESCON Directors

```
ESCD4CCHPID PATH=(20,70),TYPE=CNC,SWITCH=4C
ESCD4DCHPID PATH=(21,71),TYPE=CNC,SWITCH=4D
ESCD4ECHPID PATH=(30,80),TYPE=CNC,SWITCH=4E
ESCD4F CHPID PATH=(31,81),TYPE=CNC,SWITCH=4F

CU1CNTLUNIT CUNUMBR=001,
                PATH=(20,21,30,31,70,71,80,81),
                LINK=(D0,D4,D0,D4,D4,D0,D4,D0),
                UNIT=3490,CUADD=0,
                UNITADD=((00,16))

STRING1  IODEVICE ADDRESS=(0500,16),
           CUNUMBER=(001),
      UNIT=3490,
UNITADD=00,STADET=Y

CU2CNTLUNIT CUNUMBR=002,
                PATH=(20,21,30,31,70,71,80,81),
                LINK=(D0,D4,D0,D4,D4,D0,D4,D0),
                UNIT=3490,CUADD=1,
            UNITADD=((00,16))

STRING2  IODEVICE ADDRESS=(0510,16),
                 CUNUMBER=(002),
        UNIT=3490,
UNITADD=00,STADET=Y
.
.
.
CU15CNTLUNIT CUNUMBR=015,
                PATH=(20,21,30,31,70,71,80,81),
                LINK=(D0,D4,D0,D4,D4,D0,D4,D0),
                UNIT=3490,CUADD=E,
                  UNITADD=((00,16))

STRING15  IODEVICE ADDRESS=(05E0,16),
                 CUNUMBER=(015),
       UNIT=3490,
UNITADD=00,STADET=Y

CU16CNTLUNIT CUNUMBR=016,
                PATH=(20,21,30,31,70,71,80,81),
                LINK=(D0,D4,D0,D4,D4,D0,D4,D0),
                UNIT=3490,CUADD=F,
                  UNITADD=((00,16))

STRING16  IODEVICE ADDRESS=(05F0,16),
              CUNUMBER=(016),
        UNIT=3490,
UNITADD=00,STADET=Y
```

FIGURE B-12 IOCP Example: Single MVS Host Connected to a VSM4 via ESCON Directors

Logical Paths for VSM 4 with 32 Ports

A VSM4 with 32 ports has 4x the number of logical paths available to VSM2s and VSM3s. Does this mean that a VSM4 has enough logical paths for connectivity, redundancy, and throughput for *all* attached hosts? Even with 16 RTDs and 31 hosts attached, the answer is "yes" as shown in FIGURE B-13.

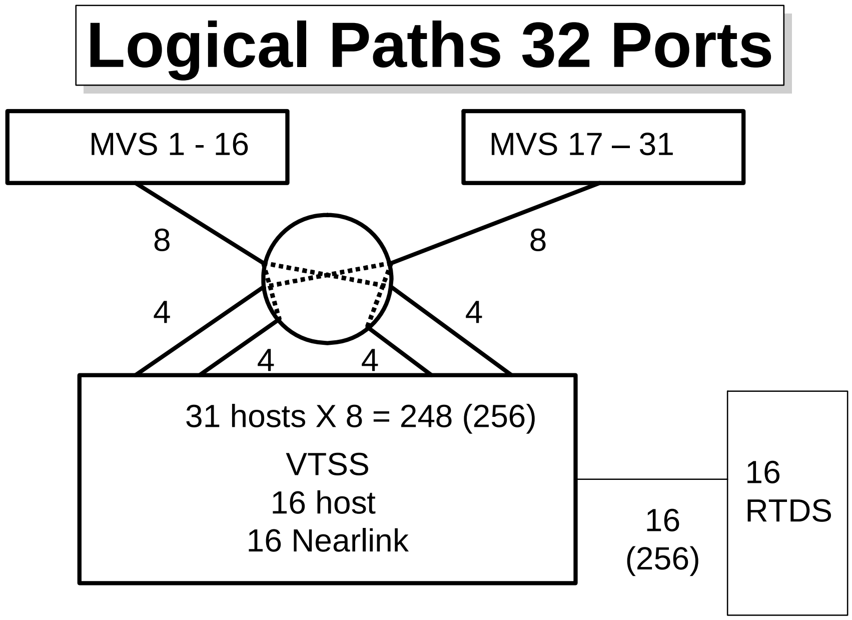


FIGURE B-13 Logical Paths for VSM 4 with 32 Ports, 31 Hosts, 16 RTDs

In FIGURE B-13:

- The 16 RTDs consume 16 x 16 or 256 logical paths.
- The **maximum** logical paths we allocated for a VSM2/3 was 4 to a host requiring maximum throughput (which also satisfied the redundancy/connectivity requirements). Therefore, if we allocated **double** that number, or 8 logical paths, for each of the 31 hosts in this configuration, we only consume 248, or 8 less than the logical paths remaining for host connections.

Therefore, logical path allocation isn't an issue, as it was with VSM2s and VSM3s.

APPENDIX **C**

VSM4 FICON Front-End and Back-End Configuration

The VSM4 FICON Back-End connectivity feature adds value to the previously available FICON front-end connectivity. TABLE C-1 summarizes the supported card configurations for VSM4 FICON Front-End plus Back-End connectivity.

TABLE C-1 Supported Card Configurations for VSM4 FICON Front-End plus Back-End Connectivity

VCF Cards	FICON Ports	ICE Cards	ESCON Ports	Total Ports	Total Logical Paths (16 per ICE Port, 64 per VCF Port)
2	4	6	24	28	640
4	8	4	16	24	768
6	12	2	8	20	896
8	16	0	0	16	1024

VSM4 FICON VCF Card Options

VSM4 supports the following FICON VCF card options:

- FIGURE C-1 shows a VSM4 with 6 ICE cards, 2 VCF cards.
- FIGURE C-2 on page 189 shows a VSM4 with 4 ICE cards, 4 VCF cards.
- FIGURE C-3 on page 190 shows a VSM4 with 2 ICE cards, 6 VCF cards.
- FIGURE C-4 on page 191 shows a VSM4 with 8 VCF cards.

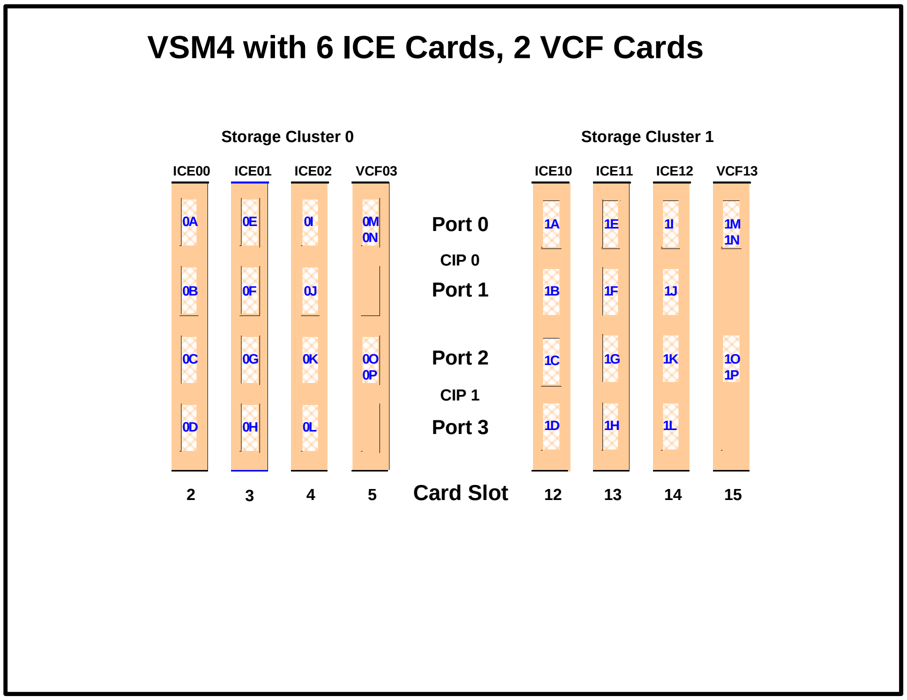


FIGURE C-1 VSM4 with 6 ICE cards, 2 VCF cards

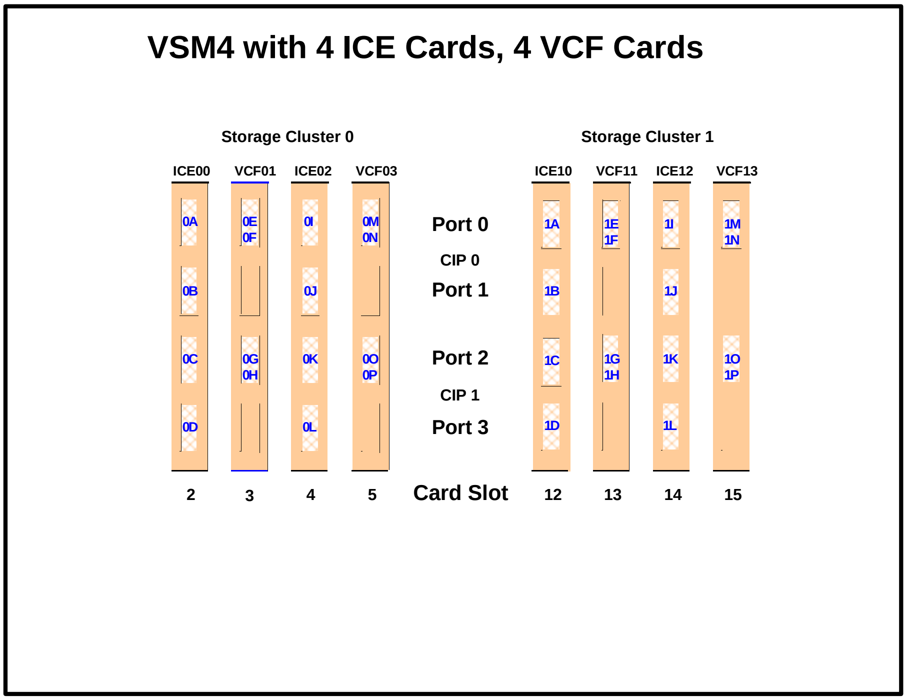


FIGURE C-2 VSM4 with 4 ICE cards, 4 VCF cards

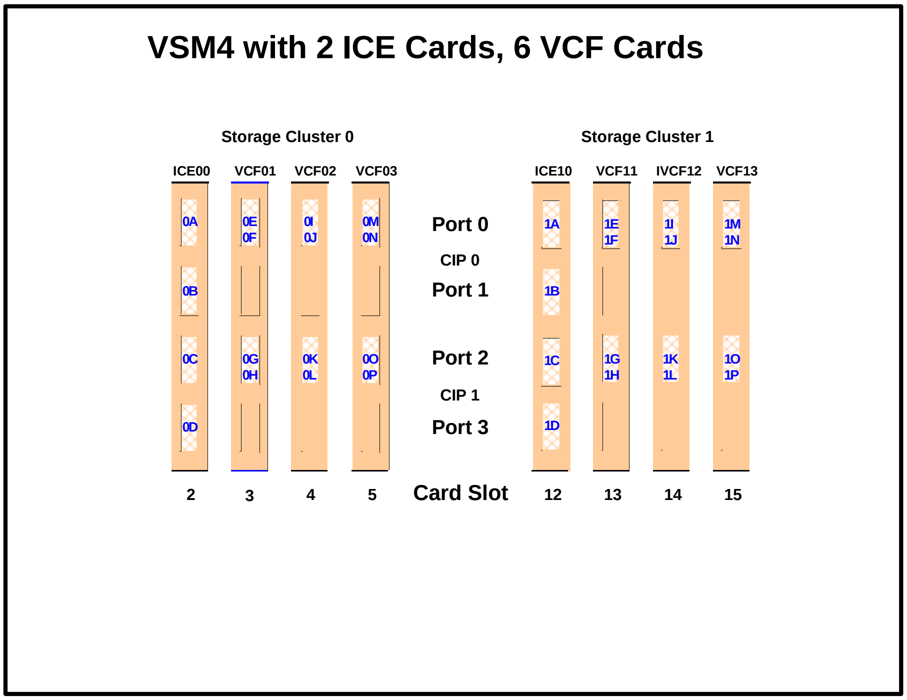


FIGURE C-3 VSM4 with 2 ICE cards, 6 VCF cards

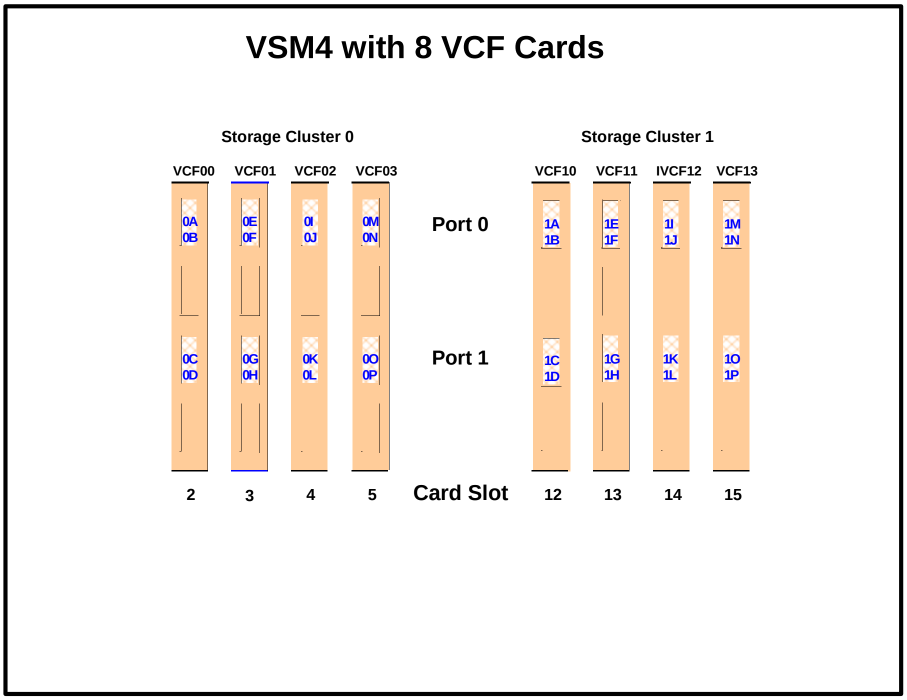


FIGURE C-4 VSM4 with 8 VCF cards

Note –

- In FIGURE C-1 on page 188 through FIGURE C-4 on page 191, the VCF cards must go in:
  - Slots 5 and 15 in a two-VCF card configuration
  - Slots 3, 5, 13, and 15 in a four-VCF card configuration.
  - Slots 3, 4, 5, 13, 14, and 15 in a six-VCF card configuration.
  - All slots in an eight-VCF card configuration.
- FICON ports are controlled by a FICON Interface processor (FIP), ESCON ports are controlled by a CIP. Regardless of the card configuration, there can be only a total of 14 Nearlink FIPs and/or CIPs.

Note – Multiple Nearlink device connections via a FICON switch or Director on the same port now allow:

- **Up to a total of 16 simultaneous NearLink I/O transfers**, which can be spread across multiple targets on as many as 14 NearLink ports.
- **Up to a total of 2 simultaneous NearLink I/O transfer**s are allowed per port.

- All FICON ports can be configured as either a Host port or Nearlink (RTD/CLINK origination) port. All ESCON ports continue to be configurable as host or Nearlink ports in pairs on a per CIP basis.
- As shown in FIGURE C-1 on page 188 through FIGURE C-4 on page 191, the ports are shown with their channel interface identifiers where **all ports are enabled**. These channel interface identifiers are the values that are required for the CHANIF values that you code for

the CONFIG utility. Each value is two characters in length and has a value from 0A to 1O. The first digit is the VTSS cluster ID (valid values are 0 or 1). The second digit is the group or adapter ID (valid values are A to P).

Each FICON port can attach to two RTDs, or two CLINKs, or an RTD/CLINK combination via a FICON director or supported switch (in FICON mode). **Note that**, as shown in these figures, **for RTDs only**, each FICON port has two CHANIF values **only if** the port is connected to a FICON director which is then connected to two RTDs.

- **Each ICE card** contains two pairs of ESCON ports. Each pair is controlled by its own Channel Interface Processor (CIP).
- Each host FICON channel supports 64 logical paths (times 16 logical units). However, in HCD:
  - From a single MVS host, you can only define 8 channels (CHPIDs) running to a single control unit (single VSM4).
  - You use the CNTLUNIT statement to define each VSM4 as 16 3490 control unit images.
  - You use the IODEVICE statement to define the 16 VTDs that are associated with each 3490 control unit image.
- For a VSM4, each ESCON CIP or FICON FIP can operate with only *one* of two "personalities", which is set at the VTSS LOP:
  - *Host Mode*. In Host Mode, ports can connect to the host CPU channels, including via Director(s) or channel extenders. A port in Host Mode can also serve as a CLINK terminator.

    Also note that for ESCON ports, you can have two physical paths from the same LPAR to the same CIP, as long as the two physical paths address different (not overlapping) logical control units. For example, a single host LPAR can address logical control units 0-7 on one CIP port, and 8-F on the other CIP port of the same CIP.
  - *Nearlink Mode*. In Nearlink Mode, ports can connect to an RTD. A port in Nearlink Mode can also serve as a CLINK originator.
  - **For clustering**, you need an originator port in Nearlink mode on one VTSS connected via a CLINK to a terminator port in Host mode on the other VTSS.

Caution – In bi-directional clustering, each CLINK **must be** attached to **the same Storage Cluster** on each VTSS, **which is a requirement**. Failure to configure in this manner can produce Replicate, Channel, and Communication errors! For more information and examples, see *Beyond the Basics: VTCS Leading Edge Techniques*.

In FICON, what are Best Practices for optimizing port operations? See TABLE C-2...

TABLE C-2 Optimizing VSM4 FICON Port Operations

Configuration - FICON port attached to a FICON Director (VCF)	**Best Practices**
Two CLINKs	**Attach a maximum of 2** ....because each port allows two active operations. **Note, however**, that these operations share the bandwidth of the port.
CLINK and RTD	**An advantage** if you attach one CLINK originator/one RTD per director, because both can be active.
Two RTDs	**An advantage** for the following: **Optimize use of local and remote RTDs**. During busy shifts, use only local RTDs on the FIP. During quiet periods, switch to remote RTDs for deep archive and DR work. Because you can have two active devices, you can also simultaneously run one local and one remote RTD. **Note, however**, that these operations share the bandwidth of the port. **Optimize use of different drive technologies**. As described in the previous bullet, use a T9840 as a local RTD, then switch to a T9940 for deep archive. You can also use this feature to migrate from older drive technology (such as 9490) to newer technology (such as 9840). Use Management and Storage Classes to read in data from older media, then switch to the newer technology drive to place data on new media. This technique effectively gives you greater physical connectivity to different drive technologies without incurring the overhead of full time, real time FICON connections to each drive type. Also as above, because you can have two active devices, you can also simultaneously run two RTDs with different drive technologies. **Note, however**, that these operations share the bandwidth of the port.

VSM4 FICON Front-End and Back-End Configuration Examples

For VSM4s with both FICON Front-End and Back-End connectivity, let's look at two examples of VCF card configurations and implementation:

- "VSM4 Configuration Example: 8 VCF Cards, FICON Directors, 16 RTDs" on page 195
- "VSM4 Configuration Example: 8 VCF Cards, 4 CLINKs, FICON Directors for 8 RTDs" on page 197

For a VSM4 host gen example, see "IOCP Example for Single MVS Host Connected to a VSM4 Via FICON Directors" on page 200.

VSM4 Configuration Example: 8 VCF Cards, FICON Directors, 16 RTDs

FIGURE C-5 shows CONFIG channel interface identifiers for a VSM4 with 8 VCF cards. In this configuration, we've allocated 8 ports to RTDs and 8 ports to host connections. The RTD ports are all connected to FICON directors, each of which is attached to RTDs, so the CHANIF identifiers for both RTDs are shown on each port. This allows Back-End connection to 16 RTDs, although, as with ESCON, only one RTD per port/Director can be active at a time.

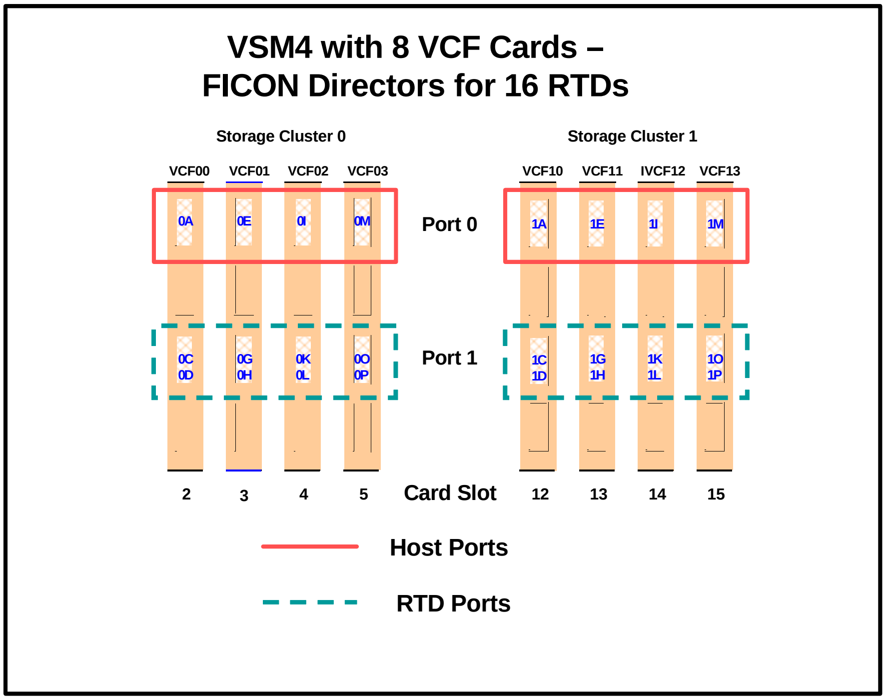


FIGURE C-5 VSM4 with 8 VCF Cards, FICON Directors for 16 RTDs

CONFIG Example for VSM4 FICON with 8 VCF Cards, FICON Directors, 16 RTDs

FIGURE C-6 shows example CONFIG JCL to define the VSM4 configuration shown in FIGURE C-5 on page 195.

```
//CREATECF EXEC PGM=SWSADMIN,PARM='MIXED'
//STEPLIB DD DSN=hlq.SLSLINK,DISP=SHR
//SLSCNTL DD DSN=FEDB.VSMLMULT.DBASEPRM,DISP=SHR
//SLSCNTL2 DD DSN=FEDB.VSMLMULT.DBASESEC,DISP=SHR
//SLSSTBY DD DSN=FEDB.VSMLMULT.DBASETBY,DISP=SHR
//SLSPRINT DD   SYSOUT=*
//SLSIN DD   *
 CONFIG
 GLOBALMAXVTV=32000 MVCFREE=40
 RECLAIMTHRESHLD=70 MAXMVC=40  START=35
 VTVVOL LOW=905000 HIGH=999999 SCRATCH
 VTVVOL LOW=C00000 HIGH=C25000 SCRATCH
 VTVVOL LOW=RMM000 HIGH=RMM020 SCRATCH
 MVCVOL LOW=N25980 HIGH=N25989
 MVCVOL LOW=N35000 HIGH=N35999
 VTSS NAME=VSM401 LOW=70 HIGH=80 MAXMIG=8 RETAIN=5
 RTD  NAME=VSM42A00 DEVNO=2A00 CHANIF=0C
 RTD  NAME=VSM42A01 DEVNO=2A01 CHANIF=0D
 RTD  NAME=VSM42A02 DEVNO=2A02 CHANIF=0G
 RTD  NAME=VSM42A03 DEVNO=2A03 CHANIF=0H
 RTD  NAME=VSM42A04 DEVNO=2A04 CHANIF=0K
 RTD  NAME=VSM42A05 DEVNO=2A05 CHANIF=0L
 RTD  NAME=VSM42A06 DEVNO=2A06 CHANIF=0O
 RTD  NAME=VSM42A07 DEVNO=2A07 CHANIF=0P
 RTD  NAME=VSM42A08 DEVNO=2A08 CHANIF=1C
 RTD  NAME=VSM42A09 DEVNO=2A09 CHANIF=1D
 RTD  NAME=VSM42A0A DEVNO=2A0A CHANIF=1G
 RTD  NAME=VSM42A0B DEVNO=2A0B CHANIF=1H
 RTD  NAME=VSM42A0C DEVNO=2A0C CHANIF=1K
 RTD  NAME=VSM42A0D DEVNO=2A0D CHANIF=1L
 RTD  NAME=VSM42A0E DEVNO=2A0E CHANIF=1O
 RTD  NAME=VSM42A0F DEVNO=2A0F CHANIF=1P
 VTD LOW=9900 HIGH=99FF
```

FIGURE C-6 CONFIG example: VSM4 with 8 VCF cards, FICON Directors, 16 RTDs

VSM4 Configuration Example: 8 VCF Cards, 4 CLINKs, FICON Directors for 8 RTDs

FIGURE C-7 shows CONFIG channel interface identifiers for a VSM4 with 8 VCF cards. In this configuration, we've allocated:

- 8 Host ports.
- 4 ports for RTDs. The RTD ports are all connected to FICON directors, each of which is attached to RTDs, so the CHANIF identifiers for both RTDs are shown on each port. This allows Back-End connection to 8 RTDs. As with ESCON, only one RTD per port/Director can be active at a time.
- 2 Nearlink mode ports for outgoing replications.
- 2 Host mode ports for incoming replications.

To form the clustered VTSS, we'll have two VSM4s (VSMPR1 and VSMPR2) configured identically as shown in FIGURE C-7. As shown in Figure 46 on page 150, Bi-Directional Clustering requires pairs of Uni-Directional CLINKs with the FIPSs configured so that the data flows in **opposite directions** on the CLINKs. To make that happen, let's make 0G and 0O the outbound (Nearlink Mode) CLINK ports on both VTSSs and 1G and 1O the inbound (Host Mode) CLINK ports on both VTSSs.

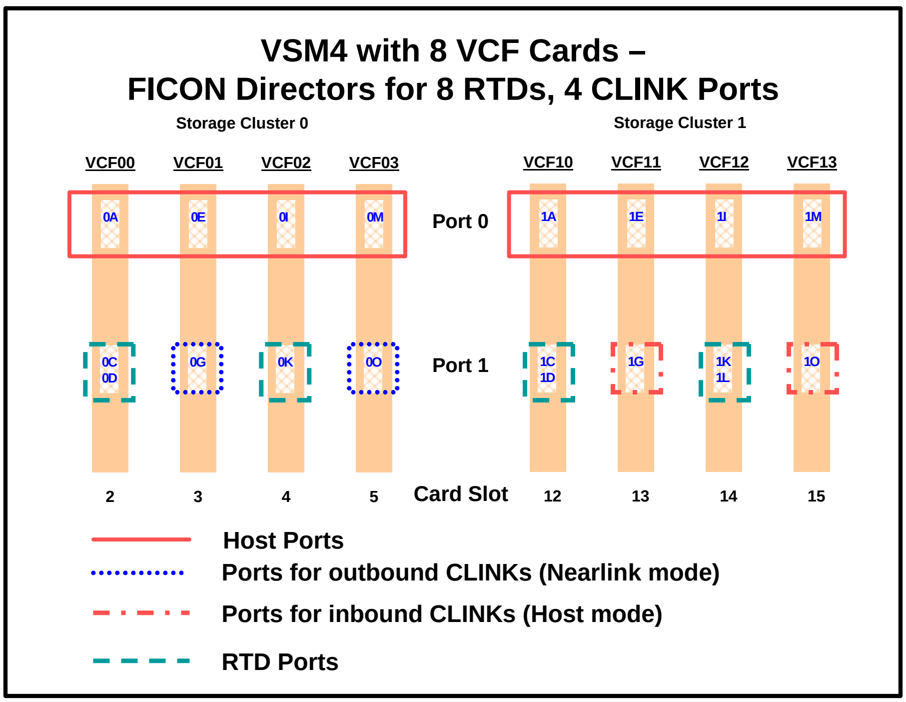


FIGURE C-7 VSM4 with 8 VCF Cards, 8 Host Ports, FICON Directors for 8 RTDs, 4 CLINK Ports

CONFIG Example for Bi-Directional Clustered VSM4 FICON Back-End

FIGURE C-8 shows example CONFIG JCL to define a Bi-Directional Cluster of two VSM4s (VSMPR1 and VSMPR2) with identical VCF card configurations shown in FIGURE C-7 on page 197.

Caution – Bi-Directional Clustering **requires** VTCS 6.1! You **cannot** configure a Bi-Directional Cluster at releases lower than VTCS 6.1! **Also note** that the Clustered VTSSs require the Advanced Management Feature.

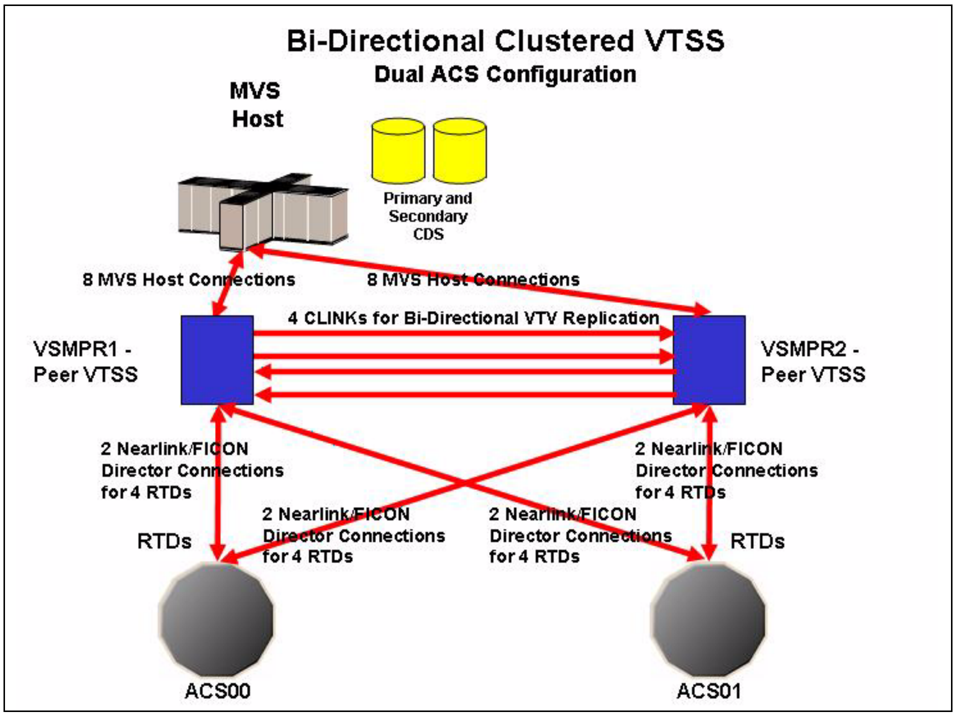


FIGURE C-8 Dual ACS Bi-Directional Clustered VTSS Configuration

FIGURE C-9 shows example CONFIG JCL to define a Bi-Directional Cluster of two VSM4s (VSMPR1 and VSMPR2) as shown in FIGURE C-8 on page 198. **Note that**:

- The CLUSTER statement defines the Cluster as consisting of VSMPR1 and VSMPR2.
- There are CLINK statements using the sending (Nearlink Mode) ports of **both VTSSs** to enable the Cluster as Bi-Directional. As described on page 156, the Nearlink ports are 0G and 0O on both VTSSs.

```
//CREATECF EXEC PGM=SWSADMIN,PARM='MIXED'
//STEPLIB DD DSN=hlq.SLSLINK,DISP=SHR
//SLSCNTL DD DSN=FEDB.VSMLMULT.DBASEPRM,DISP=SHR
//SLSCNTL2 DD DSN=FEDB.VSMLMULT.DBASESEC,DISP=SHR
//SLSSTBY DD DSN=FEDB.VSMLMULT.DBASETBY,DISP=SHR
//SLSPRINT DD   SYSOUT=*
//SLSIN DD   *
 CONFIG RESET CDSLEVEL(V61ABOVE)
 GLOBALMAXVTV=32000MVCFREE=40
 RECLAIMTHRESHLD=70MAXMVC=40  START=35
 VTVVOL LOW=905000 HIGH=999999 SCRATCH
 VTVVOL LOW=C00000 HIGH=C25000 SCRATCH
 VTVVOL LOW=RMM000 HIGH=RMM020 SCRATCH
 MVCVOL LOW=N25980 HIGH=N25989
 MVCVOL LOW=N35000 HIGH=N35999
 VTSS NAME=VSMPR1 LOW=70 HIGH=80 MAXMIG=8 MINMIG=4 RETAIN=5
 RTD  NAME=PR11A00 DEVNO=1A00 CHANIF=0C
 RTD  NAME=PR11A01 DEVNO=1A01 CHANIF=0D
 RTD  NAME=PR11A02 DEVNO=1A02 CHANIF=0K
 RTD  NAME=PR11A03 DEVNO=1A03 CHANIF=0L
 RTD  NAME=PR12A08 DEVNO=2A08 CHANIF=1C
 RTD  NAME=PR12A09 DEVNO=2A09 CHANIF=1D
 RTD  NAME=PR12A0A DEVNO=2A0A CHANIF=1K
 RTD  NAME=PR12A0B DEVNO=2A0B CHANIF=1L
 VTD LOW=9900 HIGH=99FF
 VTSS NAME=VSMPR2 LOW=70 HIGH=80 MAXMIG=8 MINMIG=4 RETAIN=5
 RTD  NAME=PR23A00 DEVNO=3A00 CHANIF=0C
 RTD  NAME=PR23A01 DEVNO=3A01 CHANIF=0D
 RTD  NAME=PR23A02 DEVNO=3A02 CHANIF=0K
 RTD  NAME=PR23A03 DEVNO=3A03 CHANIF=0L
 RTD  NAME=PR24A08 DEVNO=4A08 CHANIF=1C
 RTD  NAME=PR24A09 DEVNO=4A09 CHANIF=1D
 RTD  NAME=PR24A0A DEVNO=4A0A CHANIF=1K
 RTD  NAME=PR24A0B DEVNO=4A0B CHANIF=1L
 VTD LOW=9900 HIGH=99FF
 CLUSTER NAME=CLUSTER1 VTSSs(VSMPR1,VSMPR2)
 CLINK VTSS=VSMPR1 CHANIF=0G
 CLINK VTSS=VSMPR1 CHANIF=0O
 CLINK VTSS=VSMPR2 CHANIF=0G
 CLINK VTSS=VSMPR2 CHANIF=0O
```

FIGURE C-9 CONFIG example: Dual ACS Bi-Directional Clustered VTSS System, VSM4 FICON Back-End

IOCP Example for Single MVS Host Connected to a VSM4 Via FICON Directors

FIGURE C-10 shows a configuration diagram for a single MVS host connected to a VSM4 via FICON Directors, and FIGURE C-11 on page 201 shows example IOCP statements for this configuration. **Note that**:

- From MVSA, you define 8 CHPIDs, with each path switched in the FICON Director, for a total of 8 channels running to the VSM4.
- You code 16 CNTLUNIT statements to define the VSM4 as 16 3490 images.
- You code IODEVICE statement to define the 16 VTDs that are associated with each 3490 image.
- If ESCON and FICON channels are configured to the same logical control unit, MVS issues message CBDG489I, which indicates that mixing ESCON and FICON channel paths on a logical control unit should be used only for the migration from ESCON to native FICON, but should not be used permanently. This is a warning message only, and does not indicate an error.

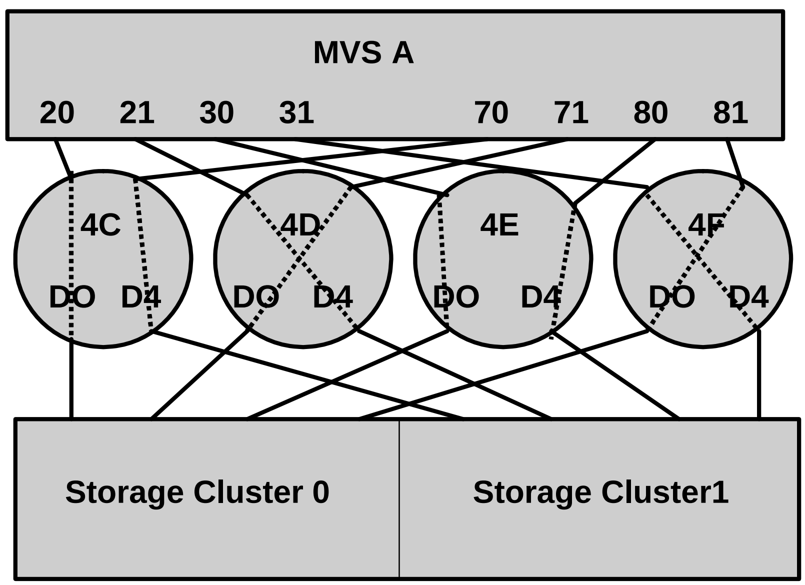


FIGURE C-10 Configuration Diagram: Single MVS Host Connected to a VSM4 via FICON Directors

```
ESCD4CCHPID PATH=(20,70),TYPE=FC,SWITCH=4C
ESCD4DCHPID PATH=(21,71),TYPE=FC,SWITCH=4D
ESCD4ECHPID PATH=(30,80),TYPE=FC,SWITCH=4E
ESCD4F CHPID PATH=(31,81),TYPE=FC,SWITCH=4F

CU1CNTLUNIT CUNUMBR=001,
                PATH=(20,21,30,31,70,71,80,81),
                LINK=(D0,D4,D0,D4,D4,D0,D4,D0),
                UNIT=3490,CUADD=0,
                UNITADD=((00,16))

STRING1  IODEVICE ADDRESS=(0500,16),
           CUNUMBER=(001),
      UNIT=3490,
UNITADD=00,STADET=Y

CU2CNTLUNIT CUNUMBR=002,
                PATH=(20,21,30,31,70,71,80,81),
                LINK=(D0,D4,D0,D4,D4,D0,D4,D0),
                UNIT=3490,CUADD=1,
            UNITADD=((00,16))

STRING2  IODEVICE ADDRESS=(0510,16),
                 CUNUMBER=(002),
        UNIT=3490,
UNITADD=00,STADET=Y
.
.
.
CU15CNTLUNIT CUNUMBR=015,
                PATH=(20,21,30,31,70,71,80,81),
                LINK=(D0,D4,D0,D4,D4,D0,D4,D0),
                UNIT=3490,CUADD=E,
                  UNITADD=((00,16))

STRING15  IODEVICE ADDRESS=(05E0,16),
                 CUNUMBER=(015),
       UNIT=3490,
UNITADD=00,STADET=Y

CU16CNTLUNIT CUNUMBR=016,
                PATH=(20,21,30,31,70,71,80,81),
                LINK=(D0,D4,D0,D4,D4,D0,D4,D0),
                UNIT=3490,CUADD=F,
                  UNITADD=((00,16))

STRING16  IODEVICE ADDRESS=(05F0,16),
              CUNUMBER=(016),
        UNIT=3490,
UNITADD=00,STADET=Y
```

FIGURE C-11 IOCP Example: Single MVS Host Connected to a VSM4 via FICON Directors

---

Tip – Unlike ESCON, FICON supports multiple active I/Os per channel. If the number of active VTDs is less than the number of channels configured to the VTSS, the I/Os to those VTDs may not be evenly spread across all the channels. As the number of active VTDs increases to be greater than the number of channels configured to the VTSS, the channel subsystem will spread the I/Os across all the channels. If it is desired to spread the I/Os across all of the channels even when only a few VTDs are active, it is necessary to use the preferred path feature to force the channel subsystem to spread the I/Os across the channels. The preferred path feature is specified via the PATH= parameter on the IODEVICE statement. When you specify preferred path on the IODEVICE statement, the channel subsystem always tries the preferred path first. If it is busy or unavailable, the channel subsystem next tries the channel path following the preferred path in the rotation order, and so on.

FIGURE C-11 on page 201 (repeated in FIGURE C-12) shows IODEVICE statements for STRING1 **without** using preferred pathing.

```
STRING1  IODEVICE ADDRESS=(0500,16),
             CUNUMBER=(001),
       UNIT=3490,
UNITADD=00,STADET=Y
```

FIGURE C-12 IODEVICE Statements for STRING 1 without Preferred Pathing

FIGURE C-13 shows IODEVICE statements for STRING1 **using** preferred pathing. If you're using preferred pathing, you need to use these kind of IODEVICE statements for **all** paths, such as STRING2 through STRING16 in FIGURE C-11 on page 201.

```
STRING10 IODEVICE ADDRESS=(0500,2),
CUNUMBER=(001),
UNIT=3490,
UNITADD=00,STADET=Y,
PATH=20

STRING12 IODEVICE ADDRESS=(0502,2),
CUNUMBER=(001),
UNIT=3490,
UNITADD=00,STADET=Y,
PATH=21

STRING14 IODEVICE ADDRESS=(0504,2),
CUNUMBER=(001),
UNIT=3490,
UNITADD=00,STADET=Y,
PATH=30

STRING16 IODEVICE ADDRESS=(0506,2),
CUNUMBER=(001),
UNIT=3490,
UNITADD=00,STADET=Y,
PATH=31

STRING18 IODEVICE ADDRESS=(0508,2),
CUNUMBER=(001),
UNIT=3490,
UNITADD=00,STADET=Y,
PATH=70

STRING1A IODEVICE ADDRESS=(050A,2),
CUNUMBER=(001),
UNIT=3490,
UNITADD=00,STADET=Y,
PATH=71

STRING1C IODEVICE ADDRESS=(050C,2),
CUNUMBER=(001),
UNIT=3490,
UNITADD=00,STADET=Y,
PATH=80

STRING1E IODEVICE ADDRESS=(050E,2),
CUNUMBER=(001),
UNIT=3490,
UNITADD=00,STADET=Y,
PATH=81
```

FIGURE C-13 IODEVICE Statements for STRING 1 Using Preferred Pathing

APPENDIX **D**

VSM5 FICON Configuration

The VSM5, provides greater capacity and throughput than the VSM4, while retaining its advantages over the VSM3. TABLE D-1 summarizes the VSM5 features.

TABLE D-1 VSM5 Features

Feature	**Description**
Host/Nearlink Interfaces	Up to 16 (FICON only)
RTDs supported	Up to 32 via FICON directors (in 3490-emulation mode only), can be a mixture of the following: 9840A, 9840B, 9840C, 9840D, 9840A, 9940B, T10000.
LSMs supported	9740, 9360, 4410, 9310, SL8500, SL3000
Maximum VTDs per VTSS	256
Maximum VTVs per VTSS	300,000

VSM5 FICON VCF Card Options - Maximum 16 RTDs

VSM5 is available **only** with VCF (FICON) cards in the following configurations for a maximum of 16 RTDs:

- FIGURE D-1 shows a VSM5 with 8 VCF cards.
- FIGURE D-2 on page 207 shows a VSM5 with 6 VCF cards, 2 empty card slots.
- FIGURE D-3 on page 208 shows a VSM5 with 4 VCF cards, 4 empty card slots.

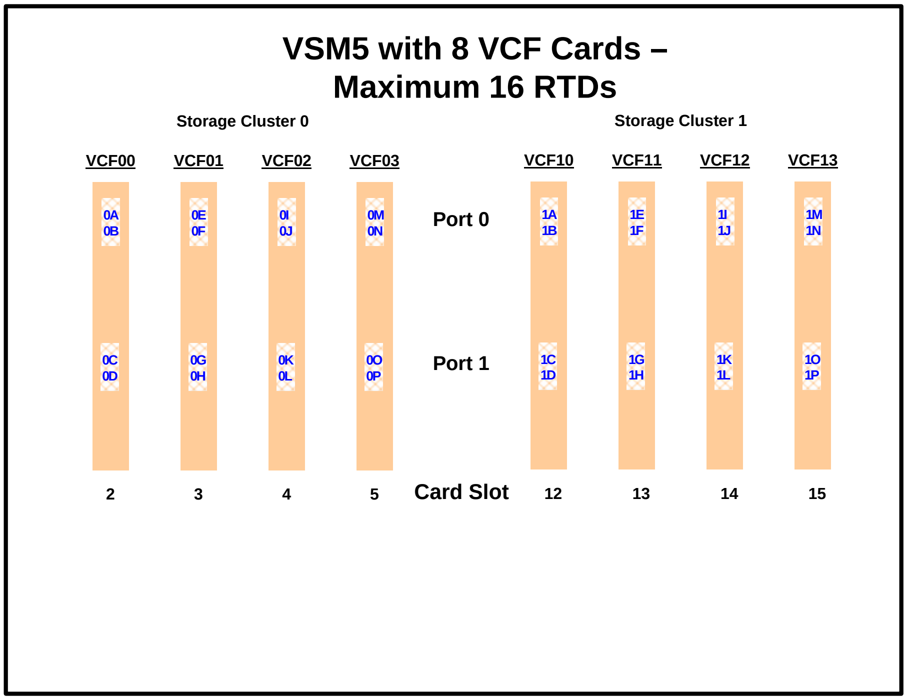


FIGURE D-1 VSM5 with 8 VCF cards - Max 16 RTDs

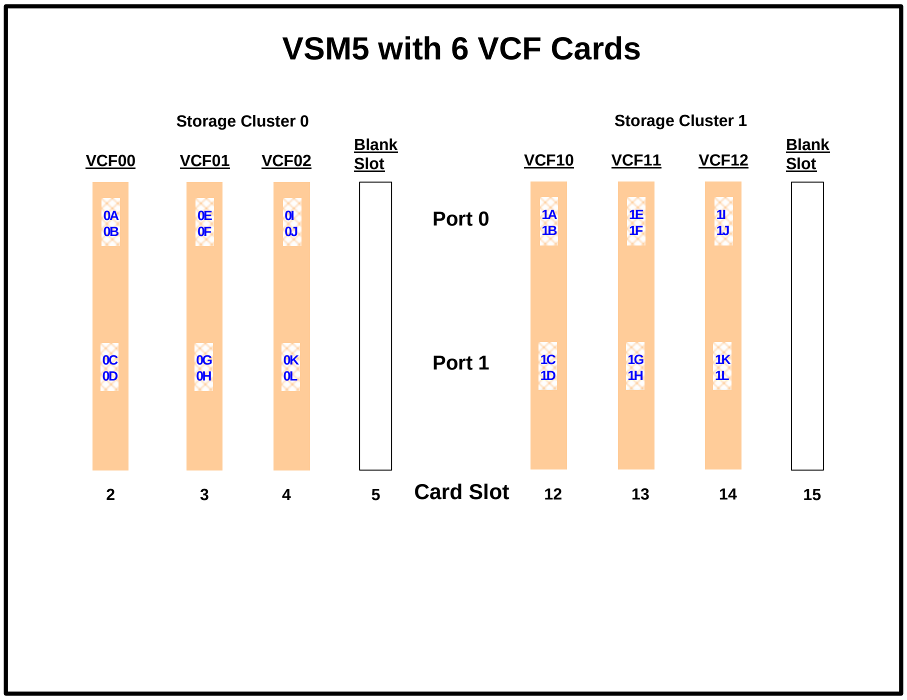


FIGURE D-2 VSM5 with 6 VCF cards, 2 empty card slots

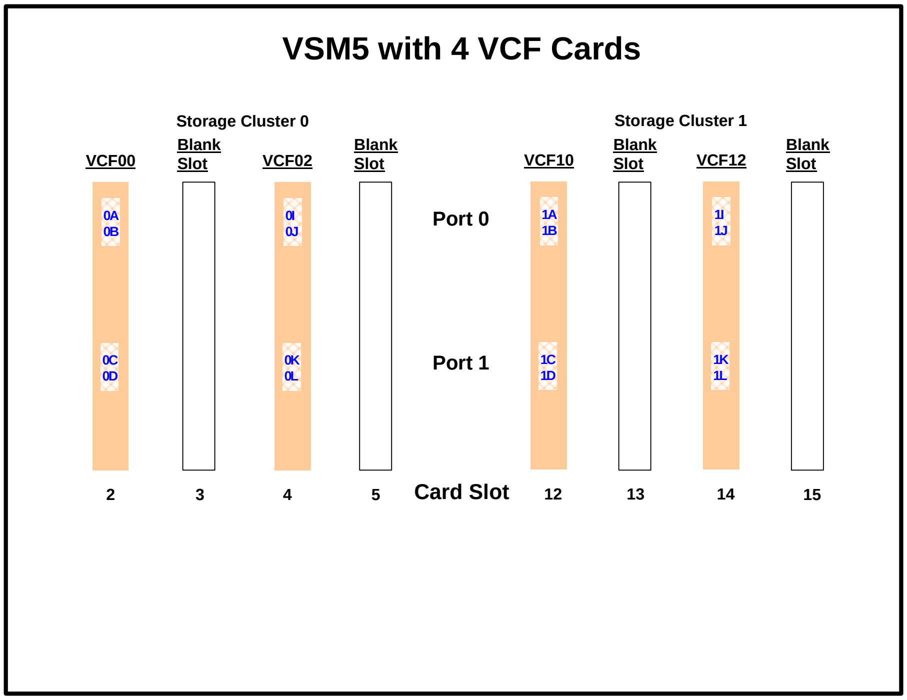


FIGURE D-3 VSM5 with 4 VCF cards, 4 empty card slots

Note –

- In FIGURE D-1 on page 206 through FIGURE D-3 on page 208, the VCF cards must go in:
  - All slots in an eight-VCF card configuration.
  - Slots 2, 3, 4, 13, 14, and 15 in a six-VCF card configuration.
  - Slots 2, 4, 14, and 15 in a four-VCF card configuration.

VSM5 FICON VCF Card Options - Maximum 32 RTDs

VSM5 is available **only** with 8 VCF (FICON) cards in the configuration for a maximum of 32 RTDs shown in FIGURE D-4. For more information on device addressing, see "RTD/CLINK Addresses - Maximum 32 RTDs" on page 63.

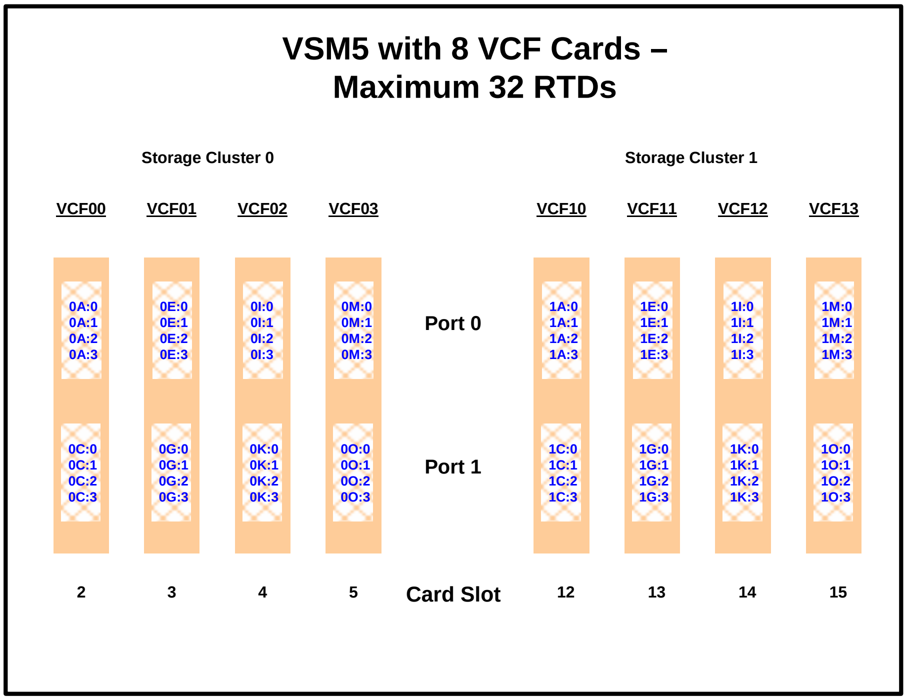


FIGURE D-4 VSM5 with 8 VCF cards - Max 32 RTDs

FICON Port Processing

Note the following:

- FICON ports are controlled by a FICON Interface processor (FIP) and there can be a total of up to 14 Nearlink FIPs.
- For a VSM5, each FIP can operate with only *one* of two “personalities”, which is set at the VTSS DOP:
  - *Host Mode*. In Host Mode, ports can connect to the host CPU channels, including via Director(s) or channel extenders. A port in Host Mode can also serve as a CLINK terminator.
  - *Nearlink Mode*. In Nearlink Mode, ports can connect to an RTD or serve as a CLINK originator, also via FICON Director(s) or channel extenders.
  - **For clustering**, you need an originator port in Nearlink mode on one VTSS connected via a CLINK to a terminator port in Host mode on the other VTSS.
- In FIGURE D-1 on page 206 through FIGURE D-4 on page 209, the ports are shown with their channel interface identifiers where **all ports are enabled**. For more information on device addressing, see “RTD/CLINK Addresses - Maximum 32 RTDs” on page 63.

  Each FICON port can attach to up to 4 RTDs, or up to 4 CLINKs, or up to 4 RTD/CLINK combinations via a FICON director or supported switch (in FICON mode). **Note that**, as shown in these figures, each FICON port has multiple device addresses **only if** the port is connected to a FICON director which is then connected to multiple devices.

Note – Multiple Nearlink device connections via a FICON switch or Director on the same port now allow:

- **Up to a total of 16 simultaneous NearLink I/O transfers**, which can be spread across multiple targets on as many as 14 NearLink ports.
- **Up to a total of 2 simultaneous NearLink I/O transfer**s are allowed per port.

See also TABLE D-2 on page 211.

- Each host FICON channel supports 64 logical paths (times 16 logical units). However, in HCD:
  - From a single MVS host, you can only define 8 channels (CHPIDs) running to a single control unit (single VSM5).
  - You use the CNTLUNIT statement to define each VSM5 as 16 3490 control unit images.
  - You use the IODEVICE statement to define the 16 VTDs that are associated with each 3490 control unit image.

Caution – In bi-directional clustering, each CLINK **must be** attached to **the same Storage Cluster** on each VTSS, **which is a requirement**. Failure to configure in this manner can produce Replicate, Channel, and Communication errors! For more information and examples, see *Beyond the Basics: VTCS Leading Edge Techniques*.

FICON Port Operations Best Practices

For FICON, what are Best Practices for optimizing port operations? See TABLE D-2...

TABLE D-2 Optimizing VSM5 FICON Port Operations

Configuration - FICON port attached to a FICON Director	Best Practices
Multiple CLINKs (up to 4)	**Attach a maximum of 2** ....because each port allows two active operations. **Note, however**, that these operations share the bandwidth of the port.
CLINK and RTD combinations	**An advantage** if you attach one CLINK originator/one RTD per director, because both can be active.
Up to 4 RTDs	**An advantage** for the following: **Optimize use of local and remote RTDs**. During busy shifts, use only local RTDs on the FIP. During quiet periods, switch to remote RTDs for deep archive and DR work. Because you can have two active devices, you can also simultaneously run one local and one remote RTD. **Note, however**, that these operations share the bandwidth of the port. **Optimize use of different drive technologies**. As described in the previous bullet, use a T9840 as a local RTD, then switch to a T9940 for deep archive. You can also use this feature to migrate from older drive technology (such as 9490) to newer technology (such as 9840). Use Management and Storage Classes to read in data from older media, then switch to the newer technology drive to place data on new media. This technique effectively gives you greater physical connectivity to different drive technologies without incurring the overhead of full time, real time FICON connections to each drive type. Also as above, because you can have two active devices, you can also simultaneously run two RTDs with different drive technologies. **Note, however**, that these operations share the bandwidth of the port.

VSM5 FICON Front-End and Back-End Configuration Examples

For VSM5s, let's look at an example of VCF card configurations and implementation in "VSM5 Configuration Example: 8 VCF Cards, FICON Directors, 32 RTDs" on page 213.

For a VSM5 host gen example, see "IOCP Example for Single MVS Host Connected to a VSM5 Via FICON Directors" on page 218.

t Implementing Support for Maximum of 32 RTDs

1. **Ensure that your system has the Maximum 32 RTDs requirements described in** TABLE 2-10 on page 62**.**

2. **Use CONFIG GLOBAL to enable support for maximum of 32 RTDs.**

```
CONFIG GLOBAL MAXRTDS=32
```

Note – Enabling support for a maximum of 32 RTDs **does not** require CONFIG RESET. However, regressing from 32 RTDs supported to 16 RTDs supported **does** require CONFIG RESET.

---

3. **Update your CONFIG RTD and CONFIG CLINK statements as required.**

   For more information, see:

   - "Maximum RTDs per VTSS" on page 62
   - "CONFIG Example for VSM5 FICON with 8 VCF Cards, FICON Directors, 32 RTDs" on page 214.

Note – The CONFIG utility RTD statement defines the RTDs connected to the VTSS. Specifically, the CONFIG RTD CHANIF parameter specifies the channel interface on the VTSS that communicates with the RTD.

Similarly, the The CONFIG utility CLINK statement defines the channel interface for a CLINK originator via the CONFIG CLINK CHANIF parameter.

Code values for the CHANIF parameter as follows:

- Regardless of whether the Maximum 32 RTDs feature is enabled, if you do not have a total of greater than 16 (RTDs, CLINK originators, or a combination of RTDs and CLINK originators) on that VTSS, you can use the "old" addressing scheme on the CHANIF parameters.
- If, however, the Maximum 32 RTDs feature is enabled and you have total of greater than 16 (RTDs, CLINK originators, or a combination of RTDs and CLINK originators) on that VTSS, you must use the "new" addressing scheme on the corresponding CHANIF parameters.

For more information, see "Maximum RTDs per VTSS" on page 62.

4. **Use the VSM5 DOP to reenter your RTD device addresses.**

   See "VSM5 DOP Panels for Maximum 32 RTDs" on page 215.

VSM5 Configuration Example: 8 VCF Cards, FICON Directors, 32 RTDs

FIGURE D-5 shows CONFIG channel interface identifiers for a VSM5 with 8 VCF cards and the Maximum 32 RTDs feature enabled. In this configuration, we've allocated 8 ports to RTDs and 8 ports to host connections. The RTD ports are all connected to FICON directors, each of which is attached to 4 RTDs, so the CHANIF identifiers for all 4 RTDs are shown on each port. This allows Back-End connection to 32 RTDs, although only one RTD per port/Director can be active at a time.

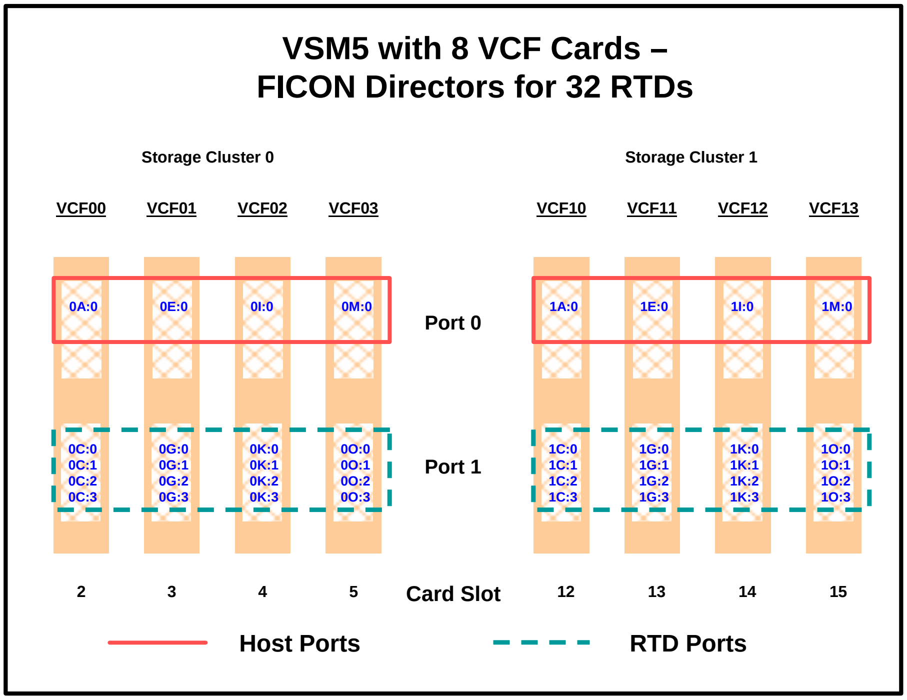


FIGURE D-5 VSM5 with 8 VCF Cards, FICON Directors for 32 RTDs

CONFIG Example for VSM5 FICON with 8 VCF Cards, FICON Directors, 32 RTDs

FIGURE D-6 shows example CONFIG JCL to define the VSM5 configuration shown in FIGURE D-5 on page 213.

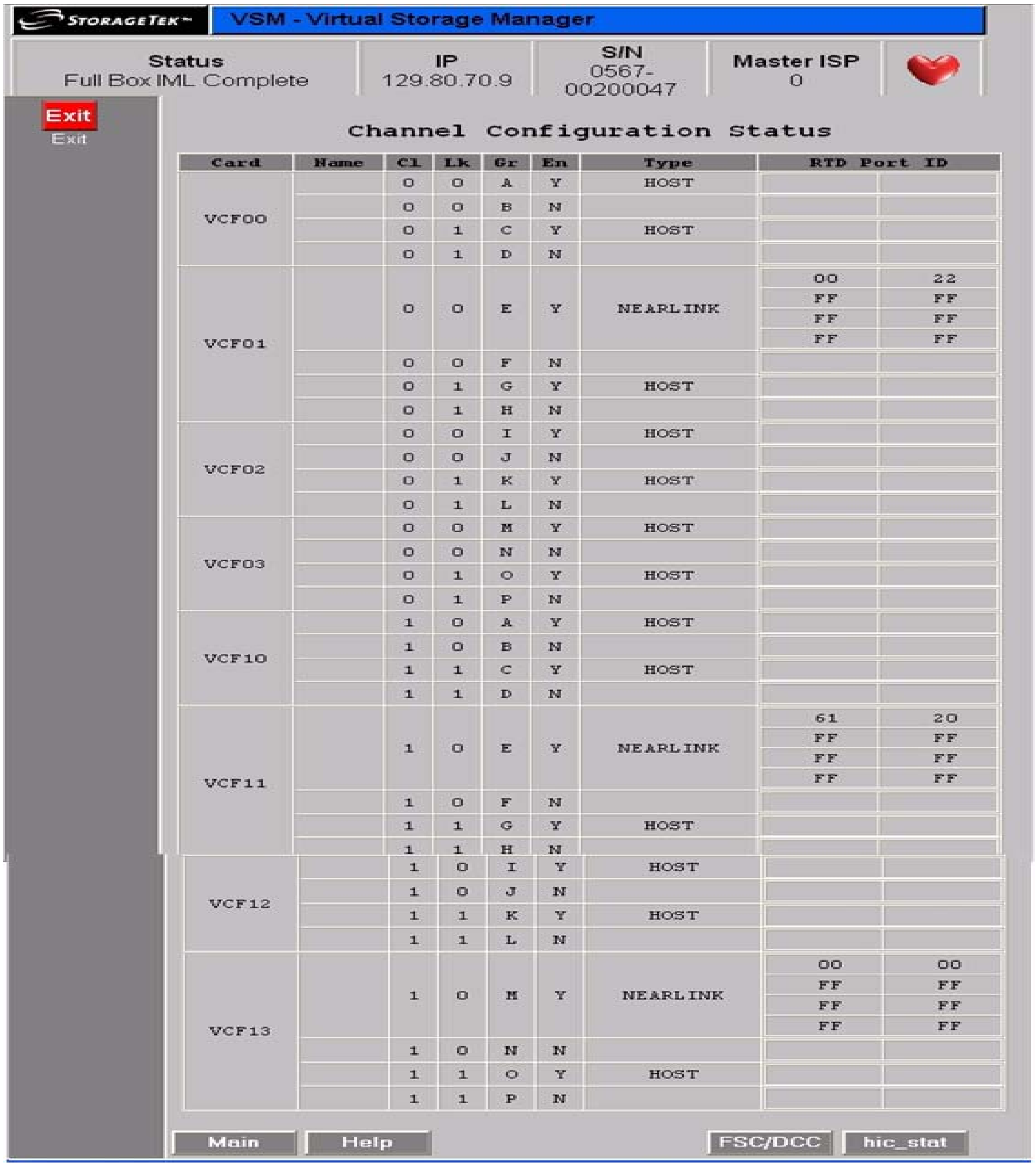
```
//CREATECF EXEC PGM=SWSADMIN,PARM='MIXED'
//STEPLIB DD DSN=hlq.SLSLINK,DISP=SHR
//SLSCNTL DD DSN=FEDB.VSMLMULT.DBASEPRM,DISP=SHR
//SLSCNTL2 DD DSN=FEDB.VSMLMULT.DBASESEC,DISP=SHR
//SLSSTBY DD DSN=FEDB.VSMLMULT.DBASETBY,DISP=SHR
//SLSPRINT DD   SYSOUT=*
//SLSIN DD   *
 CONFIG
GLOBAL MAXVTV=32000 MVCFREE=40 VTVATTR=SCRATCH RECALWER=YES LOCKSTR=VTCS_LOCKS
REPLICAT=ALWAYS VTVPAGE=LARGE SYNCHREP=YES MAXRTDS=32
 RECLAIMTHRESHLD=70 MAXMVC=40  START=35
 VTVVOL LOW=905000 HIGH=999999 SCRATCH
 VTVVOL LOW=C00000 HIGH=C25000 SCRATCH
 VTVVOL LOW=RMM000 HIGH=RMM020 SCRATCH
 MVCVOL LOW=N25980 HIGH=N25989
 MVCVOL LOW=N35000 HIGH=N35999
 VTSS NAME=VSM501 LOW=70 HIGH=80 MAXMIG=8 RETAIN=5
 RTD  NAME=VSM52A00 DEVNO=2A00 CHANIF=0C:0
 RTD  NAME=VSM52A01 DEVNO=2A01 CHANIF=0C:1
 RTD  NAME=VSM52A02 DEVNO=2A02 CHANIF=0C:2
 RTD  NAME=VSM52A03 DEVNO=2A03 CHANIF=0C:3
 RTD  NAME=VSM52A04 DEVNO=2A04 CHANIF=0G:0
 RTD  NAME=VSM52A05 DEVNO=2A05 CHANIF=0G:1
 RTD  NAME=VSM52A06 DEVNO=2A06 CHANIF=0G:2
 RTD  NAME=VSM52A07 DEVNO=2A07 CHANIF=0G:3
 RTD  NAME=VSM52A08 DEVNO=2A08 CHANIF=0K:0
 RTD  NAME=VSM52A09 DEVNO=2A09 CHANIF=0K:1
 RTD  NAME=VSM52A0A DEVNO=2A0A CHANIF=0K:2
 RTD  NAME=VSM52A0B DEVNO=2A0B CHANIF=0K:3
 RTD  NAME=VSM52A0C DEVNO=2A0C CHANIF=0O:0
 RTD  NAME=VSM52A0D DEVNO=2A0D CHANIF=0O:1
 RTD  NAME=VSM52A0E DEVNO=2A0E CHANIF=0O:2
RTD  NAME=VSM52A0F DEVNO=2A0F CHANIF=0O:3
RTD  NAME=VSM53A00 DEVNO=3A00 CHANIF=1C:0
 RTD  NAME=VSM53A01 DEVNO=3A01 CHANIF=1C:1
RTD  NAME=VSM53A02 DEVNO=3A02 CHANIF=1C:2
 RTD  NAME=VSM53A03 DEVNO=3A03 CHANIF=1C:3
 RTD  NAME=VSM53A04 DEVNO=3A04 CHANIF=1G:0
 RTD  NAME=VSM53A05 DEVNO=3A05 CHANIF=1G:1
 RTD  NAME=VSM53A06 DEVNO=3A06 CHANIF=1G:2
 RTD  NAME=VSM53A07 DEVNO=3A07 CHANIF=1G:3
 RTD  NAME=VSM53A08 DEVNO=3A08 CHANIF=1K:0
 RTD  NAME=VSM53A09 DEVNO=3A09 CHANIF=1K:1
 RTD  NAME=VSM53A0A DEVNO=3A0A CHANIF=1K:2
 RTD  NAME=VSM53A0B DEVNO=3A0B CHANIF=1K:3
 RTD  NAME=VSM53A0C DEVNO=3A0C CHANIF=1O:0
 RTD  NAME=VSM53A0D DEVNO=3A0D CHANIF=1O:1
 RTD  NAME=VSM53A0E DEVNO=3A0E CHANIF=1O:2
 RTD  NAME=VSM53A0F DEVNO=3A0F CHANIF=1O:3
 VTD LOW=9900 HIGH=99FF
```

FIGURE D-6 CONFIG example: VSM5 with 8 VCF cards, FICON Directors, 32 RTDs

VSM5 DOP Panels for Maximum 32 RTDs

Channel Configuration Status Screen

To access the *Channel Configuration Status* screen, click the active Channel Status text field on the *Configuration / Status Menu* screen.



Card	Name	Cl	Lk	Gr	En	Type	RTD Port ID	
VCF00		0	0	A	Y	HOST		
		0	0	B	N			
		0	1	C	Y	HOST		
		0	1	D	N			
VCF01		0	0	E	Y	NEARLINK	00 FF FF FF	22 FF FF FF
		0	0	F	N			
		0	1	G	Y	HOST		
		0	1	H	N			
VCF02		0	0	I	Y	HOST		
		0	0	J	N			
		0	1	K	Y	HOST		
		0	1	L	N			
VCF03		0	0	M	Y	HOST		
		0	0	N	N			
		0	1	O	Y	HOST		
		0	1	P	N			
VCF10		1	0	A	Y	HOST		
		1	0	B	N			
		1	1	C	Y	HOST		
		1	1	D	N			
VCF11		1	0	E	Y	NEARLINK	61 FF FF FF	20 FF FF FF
		1	0	F	N			
		1	1	G	Y	HOST		
		1	1	H	N			
VCF12		1	0	I	Y	HOST		
		1	0	J	N			
		1	1	K	Y	HOST		
		1	1	L	N			
VCF13		1	0	M	Y	NEARLINK	00 FF FF FF	00 FF FF FF
		1	0	N	N			
		1	1	O	Y	HOST		
		1	1	P	N			

FIGURE D-7 Channel Configuration Status Screen

Channel Configuration and RTD Path Validation Screen

To access the *Channel Configuration and RTD Path Validation* screen, click on a VCF card shown on the *Channel Configuration Status* screen.

To set the configuration of a VCF card channel for host or Nearlink use, select the channel (0 or 1) and type from the pull-down lists, then click Continue to display a subscreen with the message **Success**, indicating the configuration change completed successfully. Click Cancel to undo changed settings and return to the *Channel Configuration Status* screen.

To validate a RTD path, select a validation path (0 or 1) from the pull-down list, then click Validate RTD Path to display a subscreen with the message **Channel path *n* was successfully validated**, indicating the selected RTD path is operational.

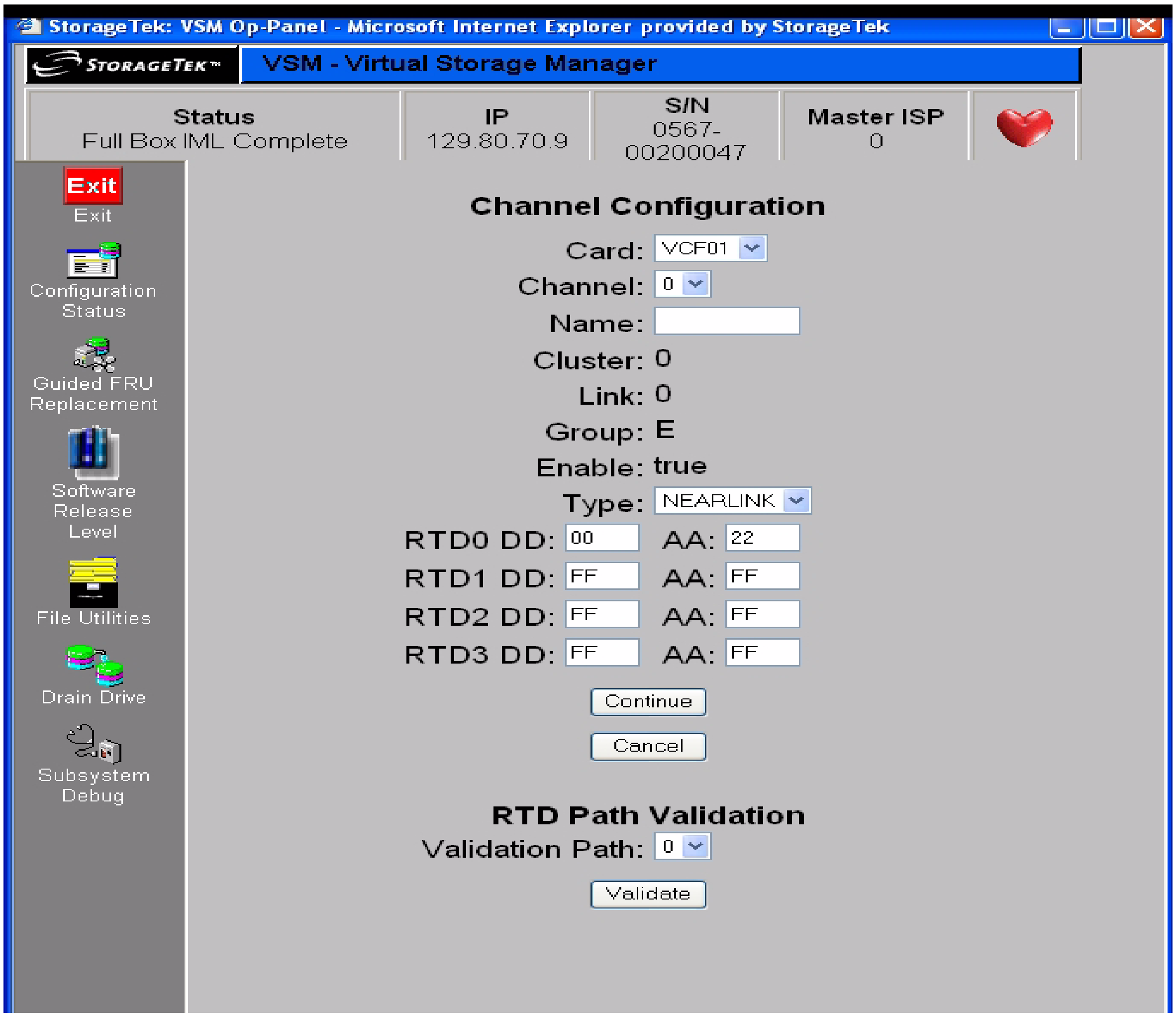


FIGURE D-8 Channel Configuration and RTD Path Validation Screen

Real Tape Drive Status Screen

To access the *Real Tape Drive Status* screen, click the active Real Tape Drive Status text field on the *Configuration / Status Menu* screen, FIGURE D-9. To validate a real tape drive (RTD), click the active button in the Valid column for the RTD. The VTSS support facility validates the RTD, then displays a subscreen with the message **RTD *n* was successfully validated. See hic_stat for details**.

Note – RTD configuration is preserved/restored across cold IML (EPO or CPD), but the links are reset and RTDs are offline until you vary them online with the VTCS VARY RTD ONLINE command.

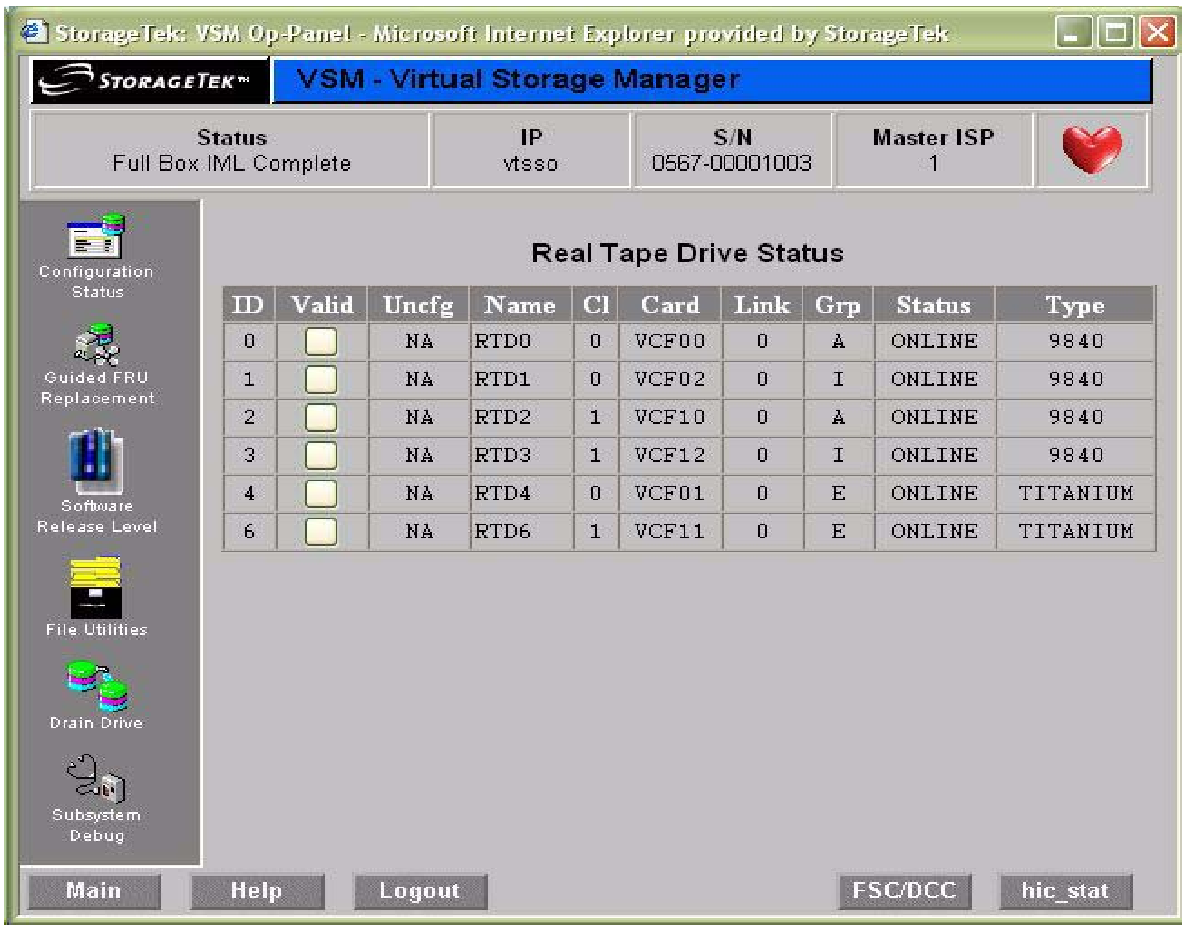


FIGURE D-9 Real Tape Drive Status Screen

IOCP Example for Single MVS Host Connected to a VSM5 Via FICON Directors

FIGURE D-10 shows a configuration diagram for a single MVS host connected to a VSM5 via FICON Directors, and FIGURE D-11 on page 219 shows example IOCP statements for this configuration. **Note that**:

- From MVSA, you define 8 CHPIDs, with each path switched in the FICON Director, for a total of 8 channels running to the VSM5.
- You code 16 CNTLUNIT statements to define the VSM5 as 16 3490 images.
- You code IODEVICE statement to define the 16 VTDs that are associated with each 3490 image.
- If ESCON and FICON channels are configured to the same logical control unit, MVS issues message CBDG489I, which indicates that mixing ESCON and FICON channel paths on a logical control unit should be used only for the migration from ESCON to native FICON, but should not be used permanently. This is a warning message only, and does not indicate an error.

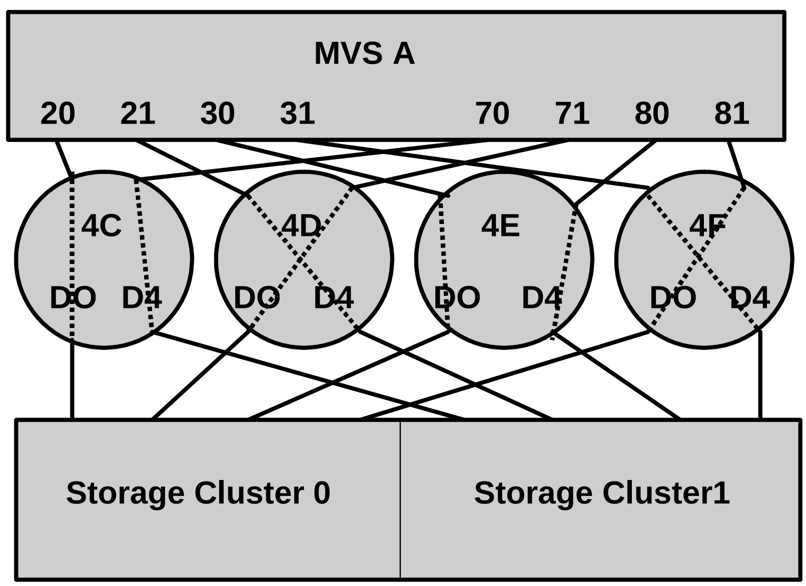


FIGURE D-10 Configuration Diagram: Single MVS Host Connected to a VSM5 via FICON Directors

```
ESCD4CCHPID PATH=(20,70),TYPE=FC,SWITCH=4C
ESCD4DCHPID PATH=(21,71),TYPE=FC,SWITCH=4D
ESCD4ECHPID PATH=(30,80),TYPE=FC,SWITCH=4E
ESCD4F CHPID PATH=(31,81),TYPE=FC,SWITCH=4F

CU1CNTLUNIT CUNUMBR=001,
                PATH=(20,21,30,31,70,71,80,81),
                LINK=(D0,D4,D0,D4,D4,D0,D4,D0),
                UNIT=3490,CUADD=0,
                UNITADD=((00,16))

STRING1  IODEVICE ADDRESS=(0500,16),
           CUNUMBER=(001),
      UNIT=3490,
UNITADD=00,STADET=Y

CU2CNTLUNIT CUNUMBR=002,
                PATH=(20,21,30,31,70,71,80,81),
                LINK=(D0,D4,D0,D4,D4,D0,D4,D0),
                UNIT=3490,CUADD=1,
            UNITADD=((00,16))

STRING2  IODEVICE ADDRESS=(0510,16),
                 CUNUMBER=(002),
        UNIT=3490,
UNITADD=00,STADET=Y
.
.
.
CU15CNTLUNIT CUNUMBR=015,
                PATH=(20,21,30,31,70,71,80,81),
                LINK=(D0,D4,D0,D4,D4,D0,D4,D0),
                UNIT=3490,CUADD=E,
                  UNITADD=((00,16))

STRING15  IODEVICE ADDRESS=(05E0,16),
                 CUNUMBER=(015),
       UNIT=3490,
UNITADD=00,STADET=Y

CU16CNTLUNIT CUNUMBR=016,
                PATH=(20,21,30,31,70,71,80,81),
                LINK=(D0,D4,D0,D4,D4,D0,D4,D0),
                UNIT=3490,CUADD=F,
                  UNITADD=((00,16))

STRING16  IODEVICE ADDRESS=(05F0,16),
              CUNUMBER=(016),
        UNIT=3490,
UNITADD=00,STADET=Y
```

FIGURE D-11 IOCP Example: Single MVS Host Connected to a VSM5 via FICON Directors

Tip – Unlike ESCON, FICON supports multiple active I/Os per channel. If the number of active VTDs is less than the number of channels configured to the VTSS, the I/Os to those VTDs may not be evenly spread across all the channels. As the number of active VTDs increases to be greater than the number of channels configured to the VTSS, the channel subsystem will spread the I/Os across all the channels. If it is desired to spread the I/Os across all of the channels even when only a few VTDs are active, it is necessary to use the preferred path feature to force the channel subsystem to spread the I/Os across the channels. The preferred path feature is specified via the PATH= parameter on the IODEVICE statement. When you specify preferred path on the IODEVICE statement, the channel subsystem always tries the preferred path first. If it is busy or unavailable, the channel subsystem next tries the channel path following the preferred path in the rotation order, and so on.

FIGURE D-11 on page 219 (repeated in FIGURE D-12) shows IODEVICE statements for STRING1 **without** using preferred pathing.

```
STRING1  IODEVICE ADDRESS=(0500,16),
             CUNUMBER=(001),
        UNIT=3490,
UNITADD=00,STADET=Y
```

FIGURE D-12 IODEVICE Statements for STRING 1 without Preferred Pathing

FIGURE D-13 shows IODEVICE statements for STRING1 **using** preferred pathing. If you're using preferred pathing, you need to use these kind of IODEVICE statements for **all** paths, such as STRING2 through STRING16 in FIGURE D-11 on page 219.

```
STRING10 IODEVICE ADDRESS=(0500,2),
CUNUMBER=(001),
UNIT=3490,
UNITADD=00,STADET=Y,
PATH=20

STRING12 IODEVICE ADDRESS=(0502,2),
CUNUMBER=(001),
UNIT=3490,
UNITADD=00,STADET=Y,
PATH=21

STRING14 IODEVICE ADDRESS=(0504,2),
CUNUMBER=(001),
UNIT=3490,
UNITADD=00,STADET=Y,
PATH=30

STRING16 IODEVICE ADDRESS=(0506,2),
CUNUMBER=(001),
UNIT=3490,
UNITADD=00,STADET=Y,
PATH=31

STRING18 IODEVICE ADDRESS=(0508,2),
CUNUMBER=(001),
UNIT=3490,
UNITADD=00,STADET=Y,
PATH=70

STRING1A IODEVICE ADDRESS=(050A,2),
CUNUMBER=(001),
UNIT=3490,
UNITADD=00,STADET=Y,
PATH=71

STRING1C IODEVICE ADDRESS=(050C,2),
CUNUMBER=(001),
UNIT=3490,
UNITADD=00,STADET=Y,
PATH=80

STRING1E IODEVICE ADDRESS=(050E,2),
CUNUMBER=(001),
UNIT=3490,
UNITADD=00,STADET=Y,
PATH=81
```

FIGURE D-13 IODEVICE Statements for STRING 1 Using Preferred Pathing

APPENDIX **E**

VSM5 ESCON/FICON Configurations

TABLE E-1 summarizes the supported VCF (FICON) and ICE (ESCON) card configurations for VSM5.

TABLE E-1 Supported Card Configurations for VSM5 ESCON/FICON

VCF Cards	FICON Ports	ICE Cards	ESCON Ports	Total Ports	Total Logical Paths (16 per ICE Port, 64 per VCF Port)
0	0	8	32	32	512
4	8	4	16	24	768

VSM5 ICE/VCF Card Options

VSM5 supports the following ICE/VCF card options:

- FIGURE E-1 shows a VSM5 with 8 ICE cards.
- FIGURE E-2 on page 225 shows a VSM5 with 4 ICE cards, 4 VCF cards.

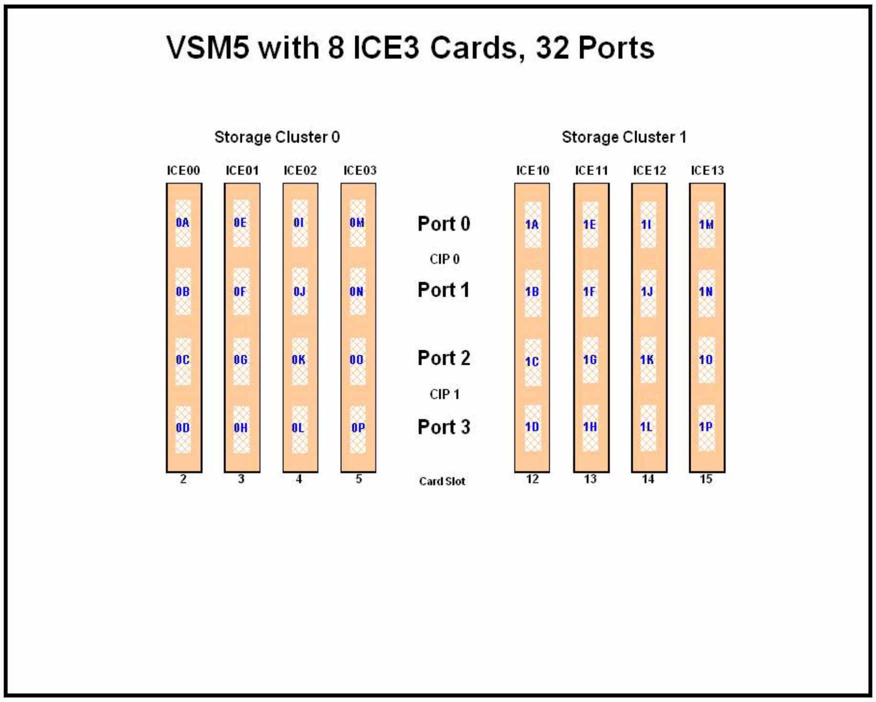


FIGURE E-1 VSM5 with 8 ICE cards

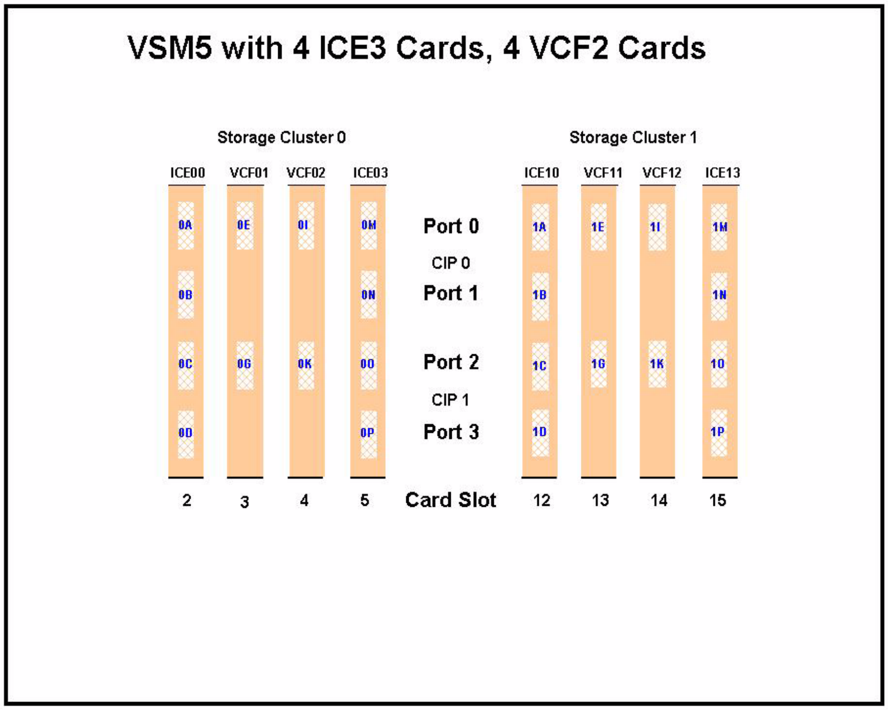


FIGURE E-2 VSM5 with 4 ICE cards, 4 VCF cards

Note –

- In FIGURE E-2 on page 225, the VCF cards must go in: Slots 3, 4, 13, and 14 in a four-VCF card configuration.
- FICON ports are controlled by a FICON Interface processor (FIP), ESCON ports are controlled by a CIP. Regardless of the card configuration, there can be only a total of 14 Nearlink FIPs and/or CIPs.

Note – With microcode level D02.06.00.00 or higher, multiple Nearlink device connections via a FICON or ESCON switch or Director on the same port now allow:

- **Up to a total of 16 simultaneous NearLink I/O transfers**, which can be spread across multiple targets on as many as 14 NearLink ports.
- **Up to a total of 2 simultaneous NearLink I/O transfer**s are allowed per port.

- All FICON ports can be configured as either a Host port or Nearlink (RTD/CLINK origination) port. All ESCON ports continue to be configurable as host or Nearlink ports in pairs on a per CIP basis.
- As shown in FIGURE E-1 on page 224 and FIGURE E-2 on page 225, the ports are shown with their channel interface identifiers. These channel interface identifiers are the values that are required for the CHANIF values that you code for the CONFIG utility. Each value is two characters in length and has a value from 0A to 1O. The first digit is the VTSS cluster ID (valid values are 0 or 1). The second digit is the group or adapter ID (valid values are A to P).

  Each FICON port can attach to four RTDs, or two CLINKs, or an RTD/CLINK combination via a FICON director or supported switch (in FICON mode). **Note that**, as shown in these figures, **for RTDs only**, each FICON port has two CHANIF values **only if** the port is connected to a FICON director which is then connected to two RTDs. Nearlink RTD connections that are paired via a FICON switch or director on the same port dynamically alternate between both RTDs for atomic operations such as mount, migrate VTV, recall VTV, etc.
- **Each ICE card** contains two pairs of ESCON ports. Each pair is controlled by its own Channel Interface Processor (CIP). Each CIP switches between the two ports, so that **only one port** can transfer data at a time, which emulates a FICON port attached to a director attached to RTDs.
- Each host FICON channel supports 64 logical paths (times 16 logical units). However, in HCD:
  - From a single MVS host, you can only define 8 channels (CHPIDs) running to a single control unit (single VSM5).
  - You use the CNTLUNIT statement to define each VSM5 as 16 3490 control unit images.
  - You use the IODEVICE statement to define the 16 VTDs that are associated with each 3490 control unit image.
- For a VSM5, each ESCON CIP or FICON FIP can operate with only *one* of two modes, which is set at the VTSS DOP:
  - *Host Mode*. In Host Mode, ports can connect to the host CPU channels, including via Director(s) or channel extenders. A port in Host Mode can also serve as a CLINK terminator.

Also note that for ESCON ports, you can have two physical paths from the same LPAR to the same CIP, as long as the two physical paths address different (not overlapping) logical control units. For example, a single host LPAR can address logical control units 0-7 on one CIP port, and 8-F on the other CIP port of the same CIP.

- *Nearlink Mode*. In Nearlink Mode, ports can connect to an RTD. A port in Nearlink Mode can also serve as a CLINK originator.

Caution – In bi-directional clustering, each CLINK **must be** attached to **the same Storage Cluster** on each VTSS, **which is a requirement**. Failure to configure in this manner can produce Replicate, Channel, and Communication errors!

In both FICON and ESCON, what are Best Practices for optimizing port operations? See TABLE E-2...

TABLE E-2 Optimizing VSM5 FICON/ESCON Port Operations

Configuration - Two ESCON Ports on a CIP (ICE) or FICON port attached to a FICON Director (VCF)	**Best Practices**
Multiple CLINKs (up to 4)	**Attach a maximum of 2** ....because each port allows two active operations. **Note, however**, that these operations share the bandwidth of the port.
CLINK and RTD combinations	**An advantage** if you attach one CLINK originator/one RTD per director, because both can be active.
Up to 4 RTDs	**An advantage** for the following: **Optimize use of local and remote RTDs**. During busy shifts, use only local RTDs on the FIP. During quiet periods, switch to remote RTDs for deep archive and DR work. Because you can have two active devices, you can also simultaneously run one local and one remote RTD. **Note, however**, that these operations share the bandwidth of the port. **Optimize use of different drive technologies**. As described in the previous bullet, use a T9840 as a local RTD, then switch to a T9940 for deep archive. You can also use this feature to migrate from older drive technology (such as 9490) to newer technology (such as 9840). Use Management and Storage Classes to read in data from older media, then switch to the newer technology drive to place data on new media. This technique effectively gives you greater physical connectivity to different drive technologies without incurring the overhead of full time, real time FICON connections to each drive type. Also as above, because you can have two active devices, you can also simultaneously run two RTDs with different drive technologies. **Note, however**, that these operations share the bandwidth of the port.

VSM5 Configuration Example: 8 ICE Cards, 16 Host Ports, 16 RTD Ports

FIGURE E-3 shows CONFIG channel interface identifiers of 16 for hosts, 16 for RTDs for a VSM5.

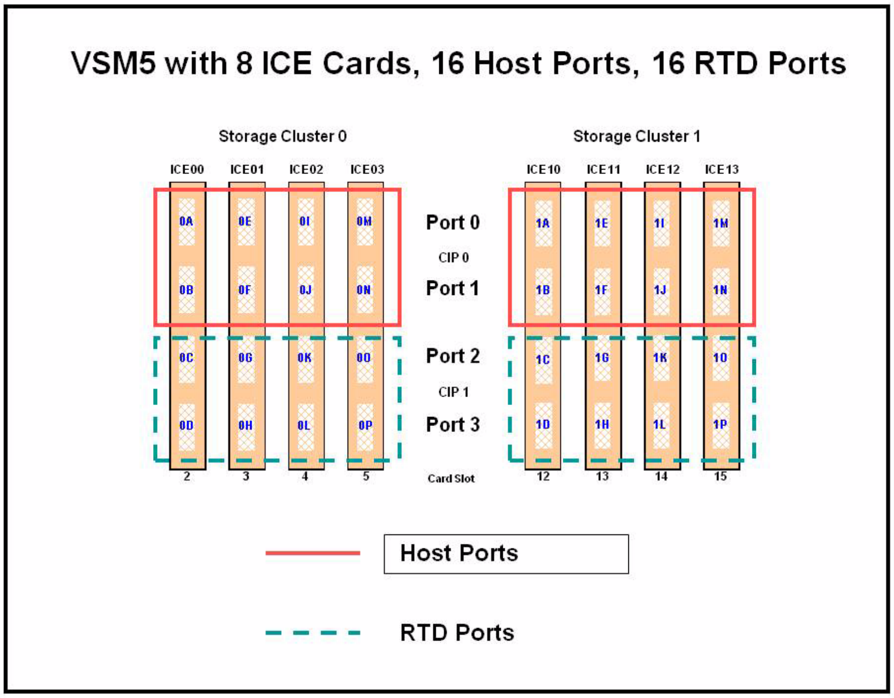


FIGURE E-3 VSM5 with 16 Host Ports, 16 RTD Ports

CONFIG Example for VSM5 with 16 Host Ports, 16 RTD Ports

FIGURE E-4 shows example CONFIG JCL to define the VSM5 configuration shown in FIGURE E-3 on page 229.

```
//CREATECF EXEC PGM=SLUADMIN,PARM='MIXED'
//STEPLIB   DD DSN=hlq.SEALINK,DISP=SHR
//SLSCNTL   DD DSN=hlq.DBASEPRM,DISP=SHR
//SLSCNTL2  DD DSN=hlq.DBASESEC,DISP=SHR
//SLSSTBY   DD DSN=hlq.DBASETBY,DISP=SHR
//SLSPRINT  DD   SYSOUT=*
//SLSIN     DD   *
 CONFIG
 GLOBAL    MAXVTV=32000MVCFREE=40
 RECLAIM   THRESHLD=70MAXMVC=40  START=35
VTSS NAME=VSM501 LOW=70 HIGH=80 MAXMIG=8 RETAIN=5
 RTD  NAME=VSM52A00 DEVNO=2A00 CHANIF=0C
 RTD  NAME=VSM52A01 DEVNO=2A01 CHANIF=0D
 RTD  NAME=VSM52A02 DEVNO=2A02 CHANIF=0G
 RTD  NAME=VSM52A03 DEVNO=2A03 CHANIF=0H
 RTD  NAME=VSM52A04 DEVNO=2A04 CHANIF=0K
 RTD  NAME=VSM52A05 DEVNO=2A05 CHANIF=0L
 RTD  NAME=VSM52A06 DEVNO=2A06 CHANIF=0O
 RTD  NAME=VSM52A07 DEVNO=2A07 CHANIF=0P
 RTD  NAME=VSM52A08 DEVNO=2A08 CHANIF=1C
 RTD  NAME=VSM52A09 DEVNO=2A09 CHANIF=1D
 RTD  NAME=VSM52A0A DEVNO=2A0A CHANIF=1G
 RTD  NAME=VSM52A0B DEVNO=2A0B CHANIF=1H
 RTD  NAME=VSM52A0C DEVNO=2A0C CHANIF=1K
 RTD  NAME=VSM52A0D DEVNO=2A0D CHANIF=1L
 RTD  NAME=VSM52A0E DEVNO=2A0E CHANIF=1O
 RTD  NAME=VSM52A0F DEVNO=2A0F CHANIF=1P
 VTD LOW=9900 HIGH=99FF
```

FIGURE E-4 CONFIG example: VSM5 with 16 Host Ports, 16 RTD Ports

IOCP Example for Single MVS Host Connected to a VSM5 Via ESCON Directors

FIGURE E-5 shows a configuration diagram for a single MVS host connected to a VSM5 via ESCON Directors, and FIGURE E-6 on page 232 shows example IOCP statements for this configuration. **Note that**:

- From MVSA, you define 8 CHPIDs, with each path switched in the ESCON Director, for a total of 8 channels running to the VSM5.
- You code 16 CNTLUNIT statements to define the VSM5 as 16 3490 images.
- You code IODEVICE statement to define the 16 VTDs that are associated with each 3490 image.

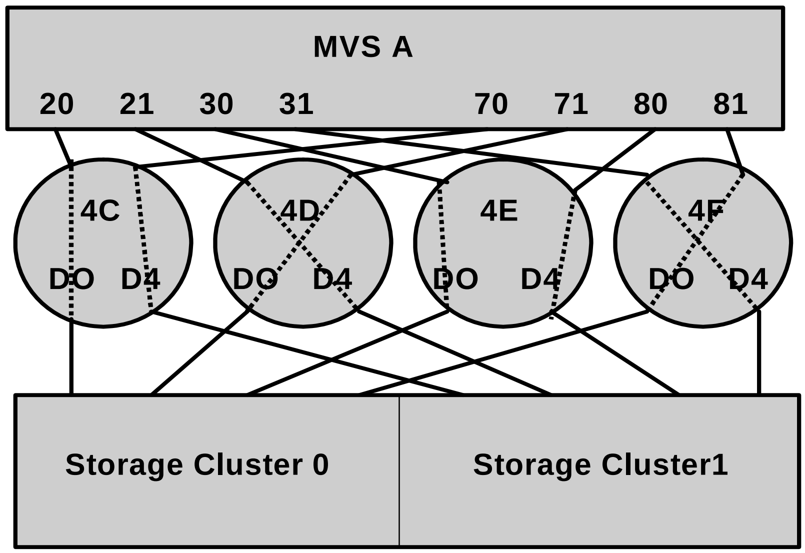


FIGURE E-5 Configuration Diagram: Single MVS Host Connected to a VSM5 via ESCON Directors

```
ESCD4C CHPID PATH=(20,70),TYPE=CNC,SWITCH=4C
ESCD4D CHPID PATH=(21,71),TYPE=CNC,SWITCH=4D
ESCD4E CHPID PATH=(30,80),TYPE=CNC,SWITCH=4E
ESCD4F CHPID PATH=(31,81),TYPE=CNC,SWITCH=4F

CU1    CNTLUNIT CUNUMBR=001,
                 PATH=(20,21,30,31,70,71,80,81),
                 LINK=(D0,D4,D0,D4,D4,D0,D4,D0),
                 UNIT=3490,CUADD=0,
                 UNITADD=((00,16))

STRING1  IODEVICE ADDRESS=(0500,16),
            CUNUMBER=(001),
              UNIT=3490,
           UNITADD=00,STADET=Y

CU2    CNTLUNIT CUNUMBR=002,
                 PATH=(20,21,30,31,70,71,80,81),
                 LINK=(D0,D4,D0,D4,D4,D0,D4,D0),
                 UNIT=3490,CUADD=1,
             UNITADD=((00,16))

STRING2  IODEVICE ADDRESS=(0510,16),
                  CUNUMBER=(002),
                UNIT=3490,
           UNITADD=00,STADET=Y
           .
           .
           .
CU15   CNTLUNIT CUNUMBR=015,
                 PATH=(20,21,30,31,70,71,80,81),
                 LINK=(D0,D4,D0,D4,D4,D0,D4,D0),
                 UNIT=3490,CUADD=E,
                   UNITADD=((00,16))

STRING15  IODEVICE ADDRESS=(05E0,16),
                  CUNUMBER=(015),
               UNIT=3490,
           UNITADD=00,STADET=Y

CU16   CNTLUNIT CUNUMBR=016,
                 PATH=(20,21,30,31,70,71,80,81),
                 LINK=(D0,D4,D0,D4,D4,D0,D4,D0),
                 UNIT=3490,CUADD=F,
                   UNITADD=((00,16))

STRING16  IODEVICE ADDRESS=(05F0,16),
               CUNUMBER=(016),
                UNIT=3490,
           UNITADD=00,STADET=Y
```

FIGURE E-6 IOCP Example: Single MVS Host Connected to a VSM5 via ESCON Directors

APPENDIX **F**

Tapeless VSM

"Tapeless VSM" basically means that you can have a VTSS without any RTDs directly attached to the VTSS; in the `CONFIG` deck, there are no `RTD` statements for the tapeless VTSS. Tapeless VSM applies to VSM4s and VSM5s.

How Does Tapeless VSM Work?

Configuring and managing a Tapeless VSM works as follows:

1. In the CONFIG deck, there are no RTD statements for the Tapeless VTSS.

Note – For clustered VTSS configurations, **all** VTSSs in the cluster must be Tapeless or **all** VTSSs in the cluster must have RTDs attached. You **cannot** mix Tapeless VTSSs and VTSSs with RTDs attached within a cluster.

---

2. The new MGMTCLAS NOMIGRAT parameter specifies that VTVs in the Management Class **are not** candidates for migration, consolidation or export, but **are** candidates to reside on a tapeless VTSS.

   VTSS selection is changed to prefer Tapeless VTSSs for VTVs in Management Classes with NOMIGRAT, and to disallow VTVs without NOMIGRAT from VTSSs with no RTDs.

   NOMIGRAT parameter is mutually exclusive with the ACSLIST, IMMEDMIG, DUPLEX, MIGPOL, ARCHAGE, ARCHPOL, RESTIME, CONSRC and CONTGT parameters.

3. A Management Class can specify DELSCR(YES), which is a proactive method to cause VSM to delete scratched VTVs, which frees VTSS buffer space and (logically) deletes any VTV copies from MVCs so that MVC space can be reclaimed. As an alternative, you can specify DELSCR(NO) and use the DELETSCR utility (which now provides a VTSS parameter to scratch VTVs on a per VTSS basis) to do demand deletes of scratched VTVs.

For a sample configuration, see "Example: Mixed Tapeless VSM" on page 237.

Note – If you have an environment that is completely Tapeless (no RTDs attached to any VTSS system), then in your LIBGEN you need to code a dummy ACS as shown in the example in "LIBGEN Example for Tapeless ACS" on page 241.

Example: Mixed Tapeless VSM

FIGURE F-1 shows two VTSSs (`VTSS1` and `VTSS2`) where `VTSS1` has no RTDs attached.

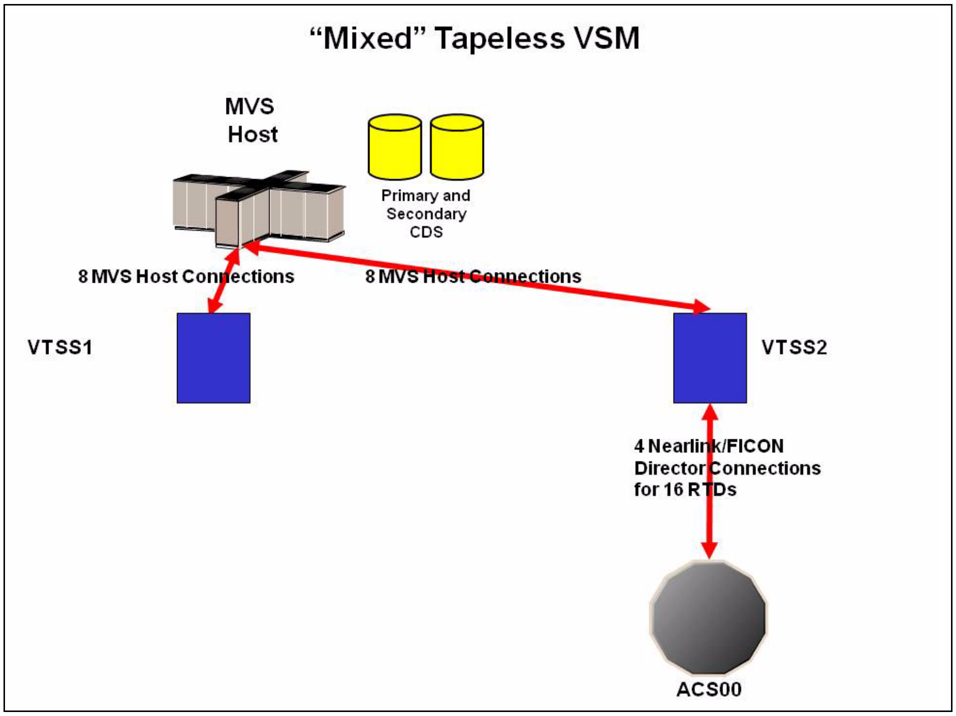


FIGURE F-1 Mixed Tapeless VSM

t Configuring the System

To configure the example system shown in FIGURE F-1 on page 237, do the following:

1. **Code a CONFIG deck as shown in FIGURE F-2.**

   In this figure, **note that** there are no `RTD` statements for `VTSS1` because it is Tapeless.

```
//CREATECF EXEC PGM=SWSADMIN,PARM='MIXED'
//STEPLIB DD DSN=hlq.SLSLINK,DISP=SHR
//SLSCNTL DD DSN=FEDB.VSMLMULT.DBASEPRM,DISP=SHR
//SLSCNTL2 DD DSN=FEDB.VSMLMULT.DBASESEC,DISP=SHR
//SLSSTBY DD DSN=FEDB.VSMLMULT.DBASETBY,DISP=SHR
//SLSPRINT DD   SYSOUT=*
//SLSIN DD   *
 CONFIG RESET CDSLEVEL(V62ABOVE)
GLOBAL MAXVTV=32000 MVCFREE=40 VTVATTR=SCRATCH RECALWER=YES LOCKSTR=VTCS_LOCKS
REPLICAT=CHANGED VTVPAGE=LARGE MAXRTDS=32
RECLAIM THRESHLD=70 MAXMVC=40  START=35
 VTVVOL LOW=905000 HIGH=999999 SCRATCH
 VTVVOL LOW=C00000 HIGH=C25000 SCRATCH
 VTVVOL LOW=RMM000 HIGH=RMM020 SCRATCH
 MVCVOL LOW=N25980 HIGH=N25989
 MVCVOL LOW=N35000 HIGH=N35999
VTSS NAME=VTSS1
VTD LOW=8900 HIGH=89FF
VTSS NAME=VTSS2 LOW=70 HIGH=80 MAXMIG=8 MINMIG=4 RETAIN=5
RTD  NAME=VPR22B00 DEVNO=2B00 CHANIF=0C:0
 RTD  NAME=VPR22B01 DEVNO=2B01 CHANIF=0C:1
 RTD  NAME=VPR22B02 DEVNO=2B02 CHANIF=0C:2
 RTD  NAME=VPR22B03 DEVNO=2B03 CHANIF=0C:3
 RTD  NAME=VPR22B04 DEVNO=2B04 CHANIF=0G:0
 RTD  NAME=VPR22B05 DEVNO=2B05 CHANIF=0G:1
 RTD  NAME=VPR22B06 DEVNO=2B06 CHANIF=0G:2
 RTD  NAME=VPR22B07 DEVNO=2B07 CHANIF=0G:3
 RTD  NAME=VPR22B08 DEVNO=2B08 CHANIF=0K:0
 RTD  NAME=VPR22B09 DEVNO=2B09 CHANIF=0K:1
 RTD  NAME=VPR22B0A DEVNO=2B0A CHANIF=0K:2
 RTD  NAME=VPR22B0B DEVNO=2B0B CHANIF=0K:3
RTD  NAME=VPR23B00 DEVNO=3B00 CHANIF=1C:0
 RTD  NAME=VPR23B01 DEVNO=3B01 CHANIF=1C:1
RTD  NAME=VPR23B02 DEVNO=3B02 CHANIF=1C:2
 RTD  NAME=VPR23B03 DEVNO=3B03 CHANIF=1C:3
 RTD  NAME=VPR23B04 DEVNO=3B04 CHANIF=1G:0
 RTD  NAME=VPR23B05 DEVNO=3B05 CHANIF=1G:1
 RTD  NAME=VPR23B06 DEVNO=3B06 CHANIF=1G:2
 RTD  NAME=VPR23B07 DEVNO=3B07 CHANIF=1G:3
 RTD  NAME=VPR23B08 DEVNO=3B08 CHANIF=1K:0
 RTD  NAME=VPR23B09 DEVNO=3B09 CHANIF=1K:1
 RTD  NAME=VPR23B0A DEVNO=3B0A CHANIF=1K:2
 RTD  NAME=VPR23B0B DEVNO=3B0B CHANIF=1K:3
VTD LOW=9900 HIGH=99FF
```

FIGURE F-2 CONFIG example: Mixed Tapeless VSM

t Defining Policies

To define policies for the example system shown in FIGURE F-1 on page 237, do the following:

1. **Enable the Advanced Management Feature.**
2. **Create the Storage Classes for VTSS2.**

```
STOR NAME(REMOTE1)
STOR NAME(REMOTE2)
```

FIGURE F-3 VTSS2 Storage Classes

3. **Create the Management Classes that point to the Storage Classes in Step 2.**

```
MGMT NAME(REM1)  STOR(REMOTE1)DELSCR(YES)
MGMT NAME(REM2)  STOR(REMOTE2)DELSCR(YES)
MGMT NAME(TAPEL)  NOMIGRAT DELSCR(YES)
```

FIGURE F-4 Management Classes

In FIGURE F-4 we created two Management Classes which point to the corresponding Storage Classes created in Step 2. Note that we also created a “Tapeless” Management Class for VTVs that permanently reside on VSMPA1 or VSMPA2 .

Note – Each Management Class in FIGURE F-4 specifies DELSCR(YES), which is a proactive method to cause VSM to delete scratched VTVs, which frees VTSS buffer space and (logically) deletes any VTV copies from MVCs so that MVC space can be reclaimed. As an alternative, you can specify DELSCR(NO) and use the DELETSCR utility (which now provides a VTSS parameter to scratch VTVs on a per VTSS basis) to do demand deletes of scratched VTVs.

---

4. **Create SMC Policies that specify virtual media and assign the Management Classes created in Step 2.**

```
POLICY NAME(PPAY) MEDIA(VIRTUAL) MGMT(REM1)
POLICY NAME(PTEST) MEDIA(VIRTUAL) MGMT(REM2)
POLICY NAME(PTAPEL) MEDIA(VIRTUAL) MGMT(TAPEL)
```

5. **Create** `TAPEREQ` **statements to route three types of critical data to VSM and assign corresponding Policies to the data.**

```
TAPEREQ DSN(*.PAYROLL.**) POLICY(REM1)
TAPEREQ DSN(*.TEST.**) POLICY(REM2)
TAPEREQ DSN(*.HR.**) POLICY(PTAPEL)
```

FIGURE F-5 TAPEREQ Statement to Route Data, Assign Policies

In FIGURE F-5, the TAPEREQ statement specifies:

- Route data sets with HLQ mask `*.PAYROLL.**` to VSM and assign Policy `PPAY`.
- Route data sets with HLQ mask `*.TEST.**` to VSM and assign Policy `PTEST`.
- Route data sets with HLQ mask `*.HR.**` to VSM and assign Policy `PTEST`.

Note – Although you can use SMC policies to direct your migrations to a specific esoteric, StorageTek recommends using **only** `MGMTCLAS` so that the SMC/VTCS allocation influencing can use any VTSS that supports the `MGMTCLAS` requirements.

LIBGEN Example for Tapeless ACS

FIGURE F-6 and FIGURE F-7 on page 242 show a LIBGEN example for a Tapeless ACS, where the `SLISTATN ADDRESS=(0032)` statement denotes a dummy ACS.

```
*
* LABELS WILL BE GENERATED IN THE OUTPUT LIBGEN AS FOLLOWS:
* ACS      - "ACSXX"    WHERE "XX" IS THE HEX ACS NUMBER 00-FF
*                       STARTING WITH ZERO
* LSM      - "LSMXXYY"  WHERE "XX" IS THE HEX ACS NUMBER OF THIS LSM
*                      AND "YY" IS THE HEX LSM NUMBER (00-FF) IN THAT
*                       ACS, STARTING AT ZERO FOR EACH NEW ACS
* STATION - "STXXH"    WHERE "XX" IS THE HEX ACS NUMBER AND H IS THE
*                       HOST INDEX IN HEX (0-F)
* PANEL   - "PXXYYPP"  WHERE "XX" IS THE HEX ACS NUMBER, YY IS THE HEX
*                      LSM NUMBER, AND PP IS THE DECIMAL PANEL NUMBER
*                       OF THE DRIVE PANEL
* DRIVE   - "DXXYYPPH" WHERE "XX" IS THE HEX ACS NUMBER, YY IS THE HEX
*                     LSM NUMBER, PP IS THE DRIVE PANEL NUMBER IN DEC
*                       IMAL, AND H IS THE HOST INDEX IN HEX
*
LIBGEN   SLIRCVRY TCHNIQE=NONE
*
        SLILIBRY SMF=231,                                             X
              ACSLIST=ACSLIST,                                        X
              HOSTID=(EC20,EC21),                                     X
              MAJNAME=STKALSQN,                                       X
              CLNPRFX=CLN,                                            X
              COMPRFX=!,                                              X
              DRVHOST=,                                               X
               SCRLABL=SL
*
ACSLIST  SLIALIST ACS00
*
ACS00   SLIACS LSM=(LSM0000,LSM0001,LSM0002,LSM0003),                 X
               STATION=(ST000,ST001)
*
ST000    SLISTATN ADDRESS=(0032)
ST001    SLISTATN ADDRESS=(0032)
*
LSM0000  SLILSM PASTHRU=((0,M),(0,M),(0,M)),                          X
             ADJACNT=(LSM0001,LSM0002,LSM0003),                       X
              DRIVE=(1),                                              X
              DRVELST=(P000001),                                      X
              TYPE=8500,                                              X
               DOOR=8500-2
*
```

FIGURE F-6 LIBGEN Example for Tapeless ACS (Part 1)

```
P000001  SLIDLIST HOSTDRV=(D0000010,D0000010)
*
D0000010 SLIDRIVS ADDRESS=(,,,,,,,,,,,,,,,)
*
LSM0001  SLILSM PASTHRU=((0,S),(0,M),(0,M)),                          X
              ADJACNT=(LSM0000,LSM0002,LSM0003),                      X
               DRIVE=(1),                                             X
               DRVELST=(P000101),                                     X
               TYPE=8500,                                             X
                DOOR=8500-2
*
P000101  SLIDLIST HOSTDRV=(D0001010,D0001010)
*
D0001010 SLIDRIVS ADDRESS=(,,,,,,,,,,,,,,,)
*
LSM0002  SLILSM PASTHRU=((0,S),(0,S),(0,M)),                          X
              ADJACNT=(LSM0000,LSM0001,LSM0003),                      X
               DRIVE=(1),                                             X
               DRVELST=(P000201),                                     X
               TYPE=8500,                                             X
                DOOR=8500-2
*
P000201  SLIDLIST HOSTDRV=(D0002010,D0002010)
*
D0002010 SLIDRIVS ADDRESS=(,,,,,,,,,,,,,,,)
*
LSM0003  SLILSM PASTHRU=((0,S),(0,S),(0,S)),                          X
              ADJACNT=(LSM0001,LSM0002,LSM0000),                      X
               DRIVE=(1),                                             X
               DRVELST=(P000301),                                     X
               TYPE=8500,                                             X
                DOOR=8500-2
*
P000301  SLIDLIST HOSTDRV=(D0003010,D0003010)
*
D0003010 SLIDRIVS ADDRESS=(,,,,,,,,,,,,,,,)
*
         SLIENDGN ,
```

FIGURE F-7 LIBGEN Example for Tapeless ACS (Part 2)

Glossary

A

access method A technique for moving data between processor storage and input/output devices.

ACS *See* Automated Cartridge System.

ACSid A method used to identify an ACS. An ACSid is the result of defining the SLIALIST macro during the library generation (LIBGEN) process. The first ACS listed in this macro acquires a hexadecimal identifier of 00, the second ACS listed acquires a hexadecimal identifier of 01, and so forth, until all ACSs are identified.

ACS routine An SMS term, referring to automatic class selection routine. Not to be confused with the HSC term, ACS, referring to automatic cartridge system.

AMT automatic migration threshold.

APF Authorized Program Facility.

APPL VTAM APPLID definition for the HSC.

archiving The storage of backup files and associated journals, usually for a given period of time.

audit A VSM audit (which is not the same as an HSC audit) reconstructs VTV and MVC information.

Automated Cartridge System (ACS) The library subsystem consisting of one or two LMUs, and from 1 to 16 attached LSMs.

automated library *See* library.

automatic mode A relationship between an LSM and all attached hosts. LSMs operating in automatic mode handle cartridges without operator intervention. This is the normal operating mode of an LSM that has been modified online.

automatic migration Migrating VTVs to MVCs that is automatically initiated and controlled by VSM.

automatic migration threshold (AMT) AMT values are percentage values that determine when virtual tape volume migration begins and ends. VTV migration begins when the VTSS buffer reaches the high AMT and ends when the buffer reaches or falls below the low AMT. These thresholds apply to all VTSSs.

automatic recall Recalling VTVs to the VTSS that is automatically initiated and controlled by VSM.

automatic reclaim Reclaiming MVC space that is automatically initiated and controlled by VSM.

B

back-end capacity The capacity of the VTSS disk buffer, in bytes, as defined in disk arrays excluding space for system overhead.

block A collection of contiguous records recorded as a unit. Blocks are separated by interblock gaps, and each block may contain one or more records.

buffer A routine or storage used to compensate for a difference in rate of data flow, or time of occurrence of events, when transferring data from one device to another.

C

CA–1 (TMS) Computer Associates Tape Management System. Third–party software by Computer Associates International, Inc.

CAP *See* Cartridge Access Port.

capacity *See* media capacity.

CAPid A CAPid uniquely defines the location of a CAP by the LSM on which it resides. A CAPid is of the form *AAL:CC* where *AA* is the ACSid, *L* is the LSM number, and *CC* is the CAP number. Some commands and utilities permit an abbreviated CAPid format of *AAL*.

cartridge The plastic housing around the tape. It is approximately 4 inches (100 mm) by 5 inches (125 mm) by 1 inch (25 mm). The tape is threaded automatically when loaded in a transport. A plastic leader block is attached to the tape for automatic threading. The spine of the cartridge contains a Tri–Optic label listing the VOLSER (tape volume identifier).

Cartridge Access Port (CAP) An assembly which allows an operator to enter/eject cartridges during automated operations. The CAP is located on the access door of an LSM. (*see also,* standard CAP, enhanced CAP, WolfCreek CAP, WolfCreek optional CAP.)

Cartridge Scratch Loader An optional feature for the Cartridge Drive. It allows the automatic loading of premounted tape cartridges or the manual loading of single tape cartridges.

cartridge system tape The basic tape cartridge media that is used with 4480, 4490, or 9490 Cartridge Subsystems. They are visually identified by a one–color cartridge case.

CAW *See* Channel Address Word.

CDRM Cross Domain Resource Manager definition (if not using existing CDRMs).

CDRSC Cross Domain Resource definition.

CDS *See* control data set.

CE Channel End.

cell A storage slot in the LSM that is used to store a tape cartridge.

Central Support Remote Center (CSRC) *See* Remote Diagnostics Center.

CFT Customer field test.

channel A device that connects the host and main storage with the input and output control units.

Channel Address Word (CAW) An area in storage that specifies the location in main storage at which a channel program begins.

channel command A command received by a CU from a channel.

Channel Status Word (CSW) An area in storage that provides information about the termination of input/output operations.

check Detection of an error condition.

CI Converter/Interpreter (JES3).

Clink (cluster link). The path between a primary VTSS and secondary VTSS in a cluster. The Clink path is used to copy replicate VTVs from the primary to the secondary.

Cluster. Two VTSSs which are physically cabled together by Clink paths and are defined in CONFIG as a cluster. A cluster consists of a primary and a secondary VTSS. VTVs with the replicate attribute attached will be copied from the primary to the secondary as soon as possible after dismount time.

connected mode A relationship between a host and an ACS. In this mode, the host and an ACS are capable of communicating (at least one station to this ACS is online).

control data set (CDS) The HSC database. In addition to the current information in the CDS, VSM keeps all its persistent data in the CDS as well.

control data set allocation map A CDS subfile that marks individual blocks as used or free.

control data set data blocks CDS blocks that contain information about the library and its configuration or environment.

control data set directory A part of the CDS that maps its subdivision into subfiles.

control data set pointer blocks CDS blocks that contain pointers to map data blocks belonging to a subfile.

control data set recovery area A portion of the CDS reserved for maintaining integrity for updates that affect multiple CDS blocks.

control data set subfile A portion of the CDS consisting of Data Blocks and Pointer Blocks containing related information.

Control Unit (CU) A microprocessor–based unit situated logically between a host channel (or channels) and from two to sixteen tape transports. It functions to translate channel commands into tape transport commands, send transport status to the channel(s), and pass data between the channel(s) and transport(s).

conventional Nearline transport An HSC–controlled transport that is not defined to VSM as an RTD.

cross–host recovery The ability for one host to perform recovery for another host that has failed.

CSE Customer Service Engineer.

CSI Consolidated System Inventory.

CSL Cartridge Scratch Loader.

CSRC Central Support Remote Center (*See* Remote Diagnostics Center)

CSW Channel Status Word.

CU *See* Control Unit.

D

DAE Dump Analysis Elimination.

DASD Direct access storage device.

data Any representations such as characters or analog quantities to which meaning is, or might be, assigned.

data class A collection of allocation and space attributes, defined by the storage administrator, that are used to create a data set.

data compaction An algorithmic data–reduction technique that encodes data from the host and stores it in less space than unencoded data. The original data is recovered by an inverse process call decompaction.

data–compaction ratio The number of host data bytes divided by the number of encoded bytes. It is variable depending on the characteristics of the data being processed. The more random the data stream, the lower the opportunity to achieve compaction.

Data Control Block (DCB) A control block used by access routines in storing and retrieving data.

data set The major unit of data storage and retrieval, consisting of a collection of data in one of several prescribed arrangements and described by control information to which the system has access.

data streaming A continuous stream of data being transmitted in character or binary–digit form, using a specified format.

DBU disk buffer utilization.

DCB Data Control Block.

demand allocation An MVS term meaning that a user has requested a specific unit.

demand migration Migrating VTVs to MVCs that an administrator does with the MIGRATE command or utility.

demand recall Recalling VTVs to the VTSS that an administrator does with the RECALL command or utility.

demand reclaim Reclaiming MVC space that an administrator does with the RECLAIM command or utility.

device number A four–digit hexadecimal number that uniquely identifies a device attached to a processor.

device separation The HSC function which *forces* the MVS device selection process to choose either a nonlibrary transport or a transport in a particular ACS, based on the location of the volume (specific requests) or the given subpool rules in effect (nonspecific request).

DFP Data Facility Product. A program that isolates applications from storage devices, storage management, and storage device hierarchy management.

DFSMS Refers to an environment running MVS/ESA SP and DFSMS/MVS, DFSORT, and RACF. This environment helps automate and centralize the management of storage through a combination of hardware, software, and policies.

DFSMS ACS routine A sequence of instructions for having the system assign data class, storage class, management class, and storage group for a data set.

directed allocation The HSC function of *influencing* MVS's selection of library transports. For a specific request, the HSC influences MVS to choose a transport requiring the fewest number of pass–thrus; for a nonspecific (scratch) request, HSC's influencing is based on the given subpool rules in effect.

disconnected mode A relationship between a host and an ACS. In this mode, the host and an ACS are not capable of communicating (there are no online stations to this ACS).

disk buffer utilization (DBU). The ratio of used to total VTSS buffer capacity.

DOMed Pertaining to a console message that was previously highlighted during execution, but is now at normal intensity.

drain The deletion of data from an MVC. May be accompanied by a "virtual" eject to prevent the MVC from being reused.

drive loaded A condition of a tape drive in which a tape cartridge has been inserted in the drive, and the tape has been threaded to the beginning–of–tape position.

DSI Dynamic System Interchange (JES3).

dual LMU A hardware/u–software feature that provides a redundant LMU capability.

dual LMU HSC release 1.1.0 or later that automates a switchover to the standby LMU in a dual LMU configuration.

dump To write the contents of storage, or of a part of storage, usually from an internal storage to an external medium, for a specific purpose such as to allow other use of storage, as a safeguard against faults or errors, or in connection with debugging.

Dynamic Device Reconfiguration (DDR) A facility that allows a demountable volume to be moved, and repositioned if necessary, without abnormally terminating the job or repeating the initial program load procedure.

E

Ecart Cartridge system tape with a length of 1100 feet that can be used with 4490 cartridge drives. These tapes are visually identified by a two–tone colored case.

EDL *See* eligible device list.

eligible device list A group of tape drives that are available to satisfy an allocation request.

enhanced CAP An enhanced CAP contains two forty–cell magazine–style CAPs and a one–cell priority CAP (PCAP). Each forty–cell CAP holds four removable magazines of ten cells each. An LSM access door with an enhanced CAP contains no cell locations for storing cartridges. An enhanced CAP is ordered as Feature Number CC80. (*see also*, Cartridge Access Port (CAP), standard CAP, WolfCreek CAP, WolfCreek optional CAP.)

Effective Recording Density The number of user bytes per unit of length of the recording medium.

eject The LSM robot places a cartridge in a Cartridge Access Port (CAP) so the operator can remove it from the LSM.

ExPR Expert Performance Reporter.

Expert Performance Reporter Expert Performance Reporter collects performance data and generates reports about StorageTek Nearline ACSs and VTSS status and performance. It has an MVS component and a PC component.

Enhanced Capacity Cartridge System Tape Cartridge system tape with increased capacity that can be used with 4490 and 9490 Cartridge Drives. These tapes are visually identified by a two–tone colored case.

EOT End–of–Tape marker.

EPO Emergency Power Off.

ERDS Error Recording Data Set.

EREP Environmental Recording, Editing, Printing.

ERP Error recovery procedures.

error recovery procedures (ERP) Procedures designed to help isolate and, where possible, to recover from errors in equipment.

ExtendedStore Library One or more LSMs with no cartridge drives (CDs) that are attached by pass–thru ports to other LSMs (with CDs) in an ACS. These LSMs provide archive storage for cartridges containing less active data sets. Cartridges can be entered and ejected directly into and out of this LSM though either a standard CAP or an enhanced CAP.

F

file protected Pertaining to a tape volume from which data can be read only. Data cannot be written on or erased from the tape.

format The arrangement or layout of data on a data medium.

G

GB 1,073,741,824 bytes of storage.

GDG Generation Data Group. An MVS data set naming convention. Sequence numbers are appended to the basic data set name to track the generations created for that data set.

GTF Generalized Trace Facility. An MVS facility used to trace software functions and events.

H

HDA Head/disk assembly.

Host Software Component (HSC) That portion of the Automated Cartridge System which executes on host systems attached to an automated library. This component acts as the interface between the operating system and the rest of the automated library.

host system A data processing system that is used to prepare programs and the operating environments for use on another computer or controller.

HSC Host Software Component.

HSM Hierarchical Storage Manager.

HWS High Watermark Setup. Relates to chains set up for tape transport allocation in JES3.

I

ICRC See Improved Cartridge Recording Capability.

Improved Cartridge Recording Capability (ICRC) An improved data recording mode that, when enabled, can increase the effective cartridge data capacity and the effective data rate when invoked.

ID Identifier or identification.

IDAX Interpreter Dynamic Allocation Exit. This is a subfunction of the DFSMS/MVS subsystem request (SSREQ 55) that the MVS JCL Interpreter and dynamic allocation functions issue for calling DFSMS ACS routines for management of the data set requested.

IML *See* Initial Microprogram Load.

index a function performed by the cartridge loader that moves cartridges down the input or output stack one cartridge position. A loader can perform multiple consecutive indexes.

Initial Microprogram Load (IML) A process that activates a machine reset and loads system programs to prepare a computer system for operation. Processors having diagnostic programs activate these programs at IML execution. Devices running u–software reload the functional u–software usually from a floppy diskette at IML execution.

Initial Program Load (IPL) A process that activates a machine reset and loads system programs to prepare a computer system for operation. Processors having diagnostic programs activate these programs at IPL execution. Devices running u–software reload the functional u–software usually from a floppy diskette at IPL execution.

initial value A value assumed until explicitly changed. It must then be explicitly specified in another command to restore the initial value. An initial value for the HSC is the value in effect when the product is installed.

inline diagnostics Diagnostic routines that test subsystem components while operating on a time–sharing basis with the functional u–software in the subsystem component.

input stack The part of the cartridge loader where cartridges are premounted.

intervention required Manual action is needed.

ips Inches per second.

IVP Installation Verification Programs. A package of programs that is run by a user after the library is installed in order to verify that the library is functioning properly.

J

JCL See Job Control Language.

Job Control Language Problem–oriented language designed to express statements in a job that are used to identify the job or describe its requirements to an operating system.

journal The log associated with journaling. The log (stored in a data set) contains a record of completed work and changes to the control data set since the last backup was created.

journaling A technique for recovery that involves creating a backup control data set and maintaining a log of all changes (transactions) to that data set.

K

KB Kilobyte, thousand bytes, or 1024 bytes.

kb kilobit, or thousand bits (10^3 bits).

keyword parameter In command and utility syntax, operands that include keywords and their related values (*see* "positional parameter"). Values are concatenated to the keyword either by an equal sign, "KEYWORD=value," or by parentheses, "KEYWORD(value)." Keyword parameters can be specified in any order. The HSC accepts (tolerates) multiple occurrences of a keyword. The value assigned to a keyword reflects the last occurrence of a keyword within a command.

L

LAN Local Area Network.

LCU *See* Library Control Unit.

LED *See* Light Emitting Diode.

LIBGEN The process of defining the configuration of the automated library to the host software.

library An installation of one or more ACSs, attached cartridge drives, volumes placed into the ACSs, host software that controls and manages the ACSs and associated volumes, and the library control data set that describes the state of the ACSs.

library control data set *See* control data set.

Library Control Unit (LCU) The portion of the LSM that controls the picking, mounting, dismounting, and replacing of cartridges.

Light Emitting Diode (LED) An electronic device used mainly as an indicator on status panels to show equipment on/off conditions.

LMU Library Management Unit. The portion of the ACS that manages from one to sixteen LSMs and communicates with the host CPU.

loader *See* Cartridge Scratch Loader.

load point The beginning of the recording area on magnetic tape.

Local Area Network (LAN) A computer network in which devices within the network can access each other for data transmission purposes. The LMU and attached LCUs are connected with a local area network.

logical ejection The process of removing a volume from the control data set without physically ejecting it from its LSM location.

LSM Library Storage Module. Provides the storage area for cartridges plus the robot necessary to move the cartridges. The term LSM often means the LCU and LSM combined.

LSMid An LSMid is composed of the ACSid concatenated with the LSM number.

LSM number A method used to identify an LSM. An LSM number is the result of defining the SLIACS macro LSM parameter during a LIBGEN. The first LSM listed in this parameter acquires the LSM number of 0 (hexadecimal), the second LSM listed acquires a hexadecimal number of 1, and so forth, until all LSMs are identified (maximum of sixteen or hexadecimal F).

M

machine initiated maintenance See ServiceTek.

magnetic recording A technique of storing data by selectively magnetizing portions of a magnetizable material.

magnetic tape A tape with a magnetizable surface layer on which data can be stored by magnetic recording.

magnetic tape drive A mechanism for moving magnetic tape and controlling it's movement.

maintenance facility Hardware contained in the CU and LMU that allows a CSE and the RDC to run diagnostics, retrieve status, and communicate with respective units through their control panels.

management class A collection of management attributes, assigned by the storage administrator, that are used to control the allocation and use of space by a data set. Note that SMS Management Classes are different from VSM Management Classes.

manual mode A relationship between an LSM and all attached hosts. LSMs operating in manual mode have been modified offline and require human assistance to perform cartridge operations.

master LMU The LMU currently controlling the functional work of the ACS in a dual LMU configuration.

MDS Main Device Scheduler (JES3).

media capacity The amount of data that can be contained on storage media and expressed in bytes of data.

micro–software *See* υ –software under Symbols.

migration The movement of VTVs from the VTSS to the RTD where the VTVs are stacked onto MVCs. See *automatic migration* and *demand migration*.

MIM Multi–Image Manager. Third–party software by CA Corporation.

mixed configurations Installations containing cartridge drives under ACS control and cartridge drives outside of library control. These configurations cause the Host Software Component to alter allocation to one or the other.

modem Modulator/demodulator. An electronic device that converts computer digital data to analog data for transmission over a telecommunications line (telephone line). At the receiving end, the modem performs the inverse function.

monitor A device that observes, records, and verifies selected system activities to determine significant departure from expected operation.

Multi-Volume Cartridge (MVC) A physical tape cartridge residing in an LSM that either contains migrated virtual tape volumes (VTVs) or is identified as a volume that can be selected for VTV stacking.

MVCPool Statement An HSC control statement that is contained in the definition data set specified by the VT MVCDEF command. An MVCPool statement specifies the MVCs that VTCS uses.

MVCDEF An HSC command that is used to load the definition data set that contains MVCPool statements.

N

O

output stack The part of the cartridge loader that receives and holds processed cartridges.

P

paired–CAP mode The two forty–cell CAPs in an enhanced CAP function in paired–CAP mode as a single eighty–cell CAP.

PARMLIB control statements Parameter library (PARMLIB) control statements allow you statically specify various operation parameters which take effect at HSC initialization. Identifying your system requirements and then specifying the appropriate control statements permits you to customize the HSC to your data center.

Pass–Thru Port (PTP) A mechanism that allows a cartridge to be passed from one LSM to another in a multiple LSM ACS.

physical end of tape A point on the tape beyond which the tape is not permitted to move.

positional parameter In command and utility syntax, operands that are identified by their position in the command string rather than by keywords (*see* “keyword parameter”). Positional parameters must be entered in the order shown in the syntax diagram.

POST *See* Program for Online System Testing.

PowderHorn A high–performance LSM (model number 9310) featuring a high–speed robot. The PowderHorn has a capacity of up to approximately 6000 cartridges.

Primary. One of two VTSSs in a cluster which is designated in CONFIG as the primary. During normal operations the primary services the host workload and copies replicate VTVs to the secondary.

Program for Online System Testing (POST) A program in a host computer that allows it to test an attached subsystem while the subsystem is online.

Program Temporary Fix A unit of corrective maintenance delivered to a customer to repair a defect in a product, or a means of packaging a Small Programming Enhancement (SPE).

Program Update Tape A tape containing a collection of PTFs. PUTs are shipped to customers on a regular basis under the conditions of the customer’s maintenance license.

PTF *See* Program Temporary Fix.

PTP *See* pass–thru port.

PUT *See* Program Update Tape.

R

RACF See Resource Access Control Facility.

Real Tape Drive (RTD) The physical transport attached to the LSM. The transport has a data path to a VTSS and may optionally have a data path to MVS or to another VTSS.

RDC *See* Remote Diagnostic Center.

recall The movement of VTVs from the MVC back to the VTSS. May be automatic or on demand.

reclaim Refers to MVC space reclamation. For automatic and demand reclamation, VTCS uses the amount of fragmented free space on the MVC and the amount of VTV data that would have to be moved to determine if space reclamation is justified.

Reconciliation. An automatic process initiated when a cluster is reestablished after the primary or secondary has been offline. Reconciliation ensures that the contents of the primary and secondary are identical with respect to replicate VTVs.

Recording Density The number of bits in a single linear track measured per unit of length of the recording medium.

Remote Diagnostic Center (RDC) The Remote Diagnostic Center at StorageTek. RDC operators can access and test StorageTek systems and software, through telecommunications lines, from remote customer installations. Also referred to as the Central Support Remote Center (CSRC).

Replication. Copying a replicate VTV from the primary VTSS to the secondary VTSS in a cluster. When replication completes, there are two copies of the VTV, one in the primary and one in the secondary.

Replicate VTV. A VTV which has had the replicate attribute attached to it by a management class statement.

Resource Access Control Facility (RACF) Security software controlling access to data sets.

RTD *See* real tape drive.

S

SCP See System Control Program.

scratch tape subpool A defined subset of all scratch tapes. Subpools are composed of one or more ranges of VOLSERs with similar physical characteristics (type of volume {reel or cartridge}, reel size, length, physical location, etc.). Some installations may also subdivide their scratch pools by other characteristics, such as label type (AL, SL, NSL, NL). The purpose of subpooling is to ensure that certain data sets are built only within particular ranges of volumes (for whatever reason the user desires). If a volume which does not belong to the required subpool is mounted for a particular data set, it is dismounted and the mount reissued.

Secondary. One of two VTSSs in a cluster which is designated in CONFIG as the secondary. During normal operations the secondary receives copies of replicate VTVs, stores them, and makes a migration copy on an MVC as soon as possible.

secondary recording A technique for recovery involving maintaining both a control data set and a copy (secondary) of the control data set.

SER Software Enhancement Request.

ServiceTek (machine initiated maintenance) A unique feature of the ACS in which an expert system monitors conditions and performance of subsystems and requests operator attention before a potential problem impacts operations. Customers can set maintenance threshold levels.

servo A device that uses feedback from a sensing element to control mechanical motion.

Small Programming Enhancement (SPE) A supplement to a released program that can affect several products or components.

SMF System Management Facility. An MVS facility used to record system actions which affect system functionality.

SMP System Modification Program.

SMP/E System Modification Program Extended.

SMS System Managed Storage.

SPE Small Programming Enhancement.

standard CAP A standard CAP has a capacity of twenty–one cartridges (three rows of seven cells each). An LSM access door with a standard CAP contains cell locations for storing cartridges. (*see also,* Cartridge Access Port (CAP), enhanced CAP.)

standard LSM A model 4410 LSM which a has a storage capacity of up to approximately 6000 cartridges.

standby The status of a station that has been varied online but is connected to the standby LMU of a dual LMU ACS.

standby LMU The redundant LMU in a dual LMU configuration that is ready to take over in case of a master LMU failure or when the operator issues the SWitch command.

station A hardware path between the host computer and an LMU over which the HSC and LMU send control information.

storage class A named list of storage attributes that identify performance goals and availability requirements for a data set. Note that SMS Storage Classes are different from VSM Storage Classes.

storage group A collection of storage volumes and attributes defined by the storage administrator. Note that this is an SMS concept, not a VSM concept.

switchover The assumption of master LMU functionality by the standby LMU.

System Control Program The general term to describe a program which controls access to system resources, and allocates those resources among executing tasks.

system–managed storage Storage that is managed by the Storage Management Subsystem, which attempts to deliver required services for availability, performance, space, and security applications.

System Modification Program Extended An IBM–licensed program used to install software and software maintenance.

T

tape cartridge A container holding magnetic tape that can be processed without separating it from the container.

tape drive A device that is used for moving magnetic tape and includes the mechanisms for writing and reading data to and from the tape.

TAPEREQ An HSC control statement that is contained in the definition data set specified by the TREQDEF command. A TAPEREQ statement defines a specific tape request. It is divided into two parts, the input: job name, step name, program name, data set name, expiration date or retention period, and an indication for specific requests or nonspecific (scratch) requests; and the output: media type and recording technique capabilities. You can use TAPEREQ statements to direct data sets to VSM.

tape unit A device that contains tape drives and their associated power supplies and electronics.

Timberwolf (9740) LSM A high performance LSM that provides a storage capacity of up to 494 cartridges. Up to 10 drives (STD, 4490, 9490, 9490EE, 9840, and SD-3) can be configured. Timberwolf LSMs can only attach to other Timberwolfs.

TMS Tape Management System.

TP Tape–to–Print.

transaction A short series of actions with the control data set. These actions are usually related to a specific function (e.g., Mount, ENter).

transport An electromechanical device capable of threading tape from a cartridge, moving the tape across a read/write head, and writing data onto or reading data from the tape.

TREQDEF An HSC command that is used to load the definition data set that contains TAPEREQ control statements.

Tri–Optic label An external label attached to the spine of a cartridge that is both human and machine readable.

TT Tape–to–Tape.

U

UNITATTR An HSC control statement that is contained in the definition data set specified by the UNITDEF command. A UNITATTR statement defines to the HSC the transport's media type and recording technique capabilities. For VSM, the UNITATTR statements define the VTD addresses to VSM as virtual and associate them with a VTSS.

UNITDEF An HSC command that is used to load the definition data set that contains UNITATTR control statements.

utilities Utility programs. The programs that allow an operator to manage the resources of the library and to monitor overall library performance.

V

Virtual Storage Manager (VSM) A storage solution that virtualizes volumes and transports in a VTSS buffer in order to improve media and transport use. The hardware includes VTSS, which is the DASD buffer, and RTDs. The software includes VTCS, an HSC–based host software, and VTSS microcode.

Virtual Tape Control System (VTCS) The primary host code that controls activity and information about VTSSs, VTVs, RTDs, and MVCs.

Virtual Tape Drive (VTD) An emulation of a physical transport in the VTSS that looks like a physical tape transport to MVS. The data written to a VTD is really being written to DASD. The VTSS has 64 VTDs that do virtual mounts of VTVs.

Virtual Tape Storage Subsystem (VTSS) The DASD buffer containing virtual volumes (VTVs) and virtual drives (VTDs). The VTSS is a STK RAID 6 hardware device with microcode that enables transport emulation. The RAID device can read and write “tape” data from/to disk, and can read and write the data from/to an RTD.

Virtual Tape Volume (VTV) A portion of the DASD buffer that appears to the operating system as a real tape volume. Data is written to and read from the VTV, and the VTV can be migrated to and recalled from real tape.

virtual thumbwheel An HSC feature that allows read–only access to a volume that is not physically write–protected.

VOLATTR An HSC control statement that is contained in the definition data set specified by the VOLDEF command. A VOLATTR statement defines to the HSC the media type and recording technique of the specified volumes. For VSM, the VOLATTR statements define the volsers for volumes that will be used as MVCs.

VOLDEF An HSC command that is used to load the definition data set that contains VOLATTR control statements.

VOLSER A six–character alphanumeric label used to identify a tape volume.

volume A data carrier that is mounted or demounted as a unit. (*See* cartridge).

VSM *See* Virtual Storage Manager.

VTCS *See* Virtual Tape Control System.

VTD *See* virtual tape drive.

W

WolfCreek A smaller capacity high–performance LSM. WolfCreek LSMs are available in 500, 750, and 1000 cartridge capacities (model numbers 9360–050, 9360–075, and 9360–100 respectively). WolfCreek LSMs can be connected by pass–thru ports to 4410, 9310, or other WolfCreek LSMs.

WolfCreek CAP The standard WolfCreek CAP contains a 20–cell magazine–style CAP and a priority CAP (PCAP). (*see also,* Cartridge Access Port (CAP), Enhanced CAP, standard CAP, WolfCreek optional CAP.)

WolfCreek optional CAP The WolfCreek optional CAP contains a 30–cell magazine–style CAP which is added to the standard WolfCreek CAP. (*see also,* Cartridge Access Port (CAP), Enhanced CAP, standard CAP, WolfCreek CAP.)

Write Tape Mark (WTM) The operation performed to record a special magnetic mark on tape. The mark identifies a specific location on the tape.

WTM *See* Write Tape Mark.

WTO Write–to–Operator.

WTOR Write–to–Operator with reply.

Symbols

υ **-software.** Microprogram. A sequence of microinstructions used to perform preplanned functions and implement machine instructions.

Numerics

4410 LSM *See* standard LSM.

9310 LSM *See* Powderhorn LSM.

9360 LSM See Wolfcreek *LSM.*

9490 Cartridge Subsystem Cartridge tape transports that provide read/write capability for 36–track recording format and extended capacity tape and provide improved performance over the 4490 Cartridge Subsystem. 9490 transports can also read data recorded in 18–track format. The StorageTek 9490 Cartridge Subsystem offers better performance (faster data transfer rate, faster load/unload) than a 3490E device.

9490EE Cartridge Subsystem A high performance tape transport that provides read/write capability for Extended Enhanced (EEtape) cartridges. It is functionally equivalent to the IBM 3490E device.

9**740 LSM** See Timberwolf *LSM*.

9840 Cartridge Subsystem A high performance tape transport system for Enterprise and Open Systems environments that reads and writes 9840 cartridges. 9840s can be defined in 10-drive and 20-drive panel configurations. The 9840 can perform as a stand-alone subsystem with a cartridge scratch loader installed, or it can be attached to a StorageTek ACS.

Index

A

allocating
- NCS
  - distribution library data sets 94
  - target library data sets 94

AMTs
- policies 53

APF (authorized program list)
- adding MVS/HSC libraries 101

authorized program list (APF)
- adding HSC libraries 101

C

CDS
- formatting 114
- planning for 49

CONFIG utility 12, 19, 21, 23

Configuring VSM
- starting VTCS 164

connecting MVS/CSC clients to VSM 148

D

definition data sets for VSM 11

distribution data sets, allocating for NCS 94

E

esoterics
- configuring 87
- JES2 and JES3 162

esoterics:planning for 14

ExLM
- SYNCVTV function 55

H

HCD facility 13

high AMT
- policies 53

hosts
- enabled for migration 71

HSC
- adding libraries to authorized program list (APF) 101
- defining a security system user ID 86
- reconfiguration
  - creating an MVC pool 23
  - formatting the new CDS 114
  - LIBGEN definitions for RTDs 21
  - overview 111
  - PARMLIB member, updating for VSM 19, 146
  - updating the LIBGEN 112
  - verifying the LIBGEN 113
- SLUADMIN utility 158
- user exits
  - SLSUX02 162
  - SLSUX04 162
  - SLSUX15 158

J

JES2 environment
- user exit SLSUX02 162

JES2 environment:esoterics 14

JES3 environment
- user exit SLSUX04 162

JES3 environment:esoterics 14

L

LIBGEN
- updating 112
- verifying 113

LIBUNIT statement 148

low AMT
- policies 53

M

migration
- policies 67

MIH, setting 88

MVCPool control statement 23

MVCs
- creating an MVC pool 23
- defining and selecting 22
- defining available with CONFIG utility 23
- managing scratch status 24
- maximum retain interval 59, 79, 80

planning for 22
policies
space reclamation 74
protecting 24
RTV access 158
space reclamation
policies 76, 77, 78
tape management system 24
unscratching current scratch cartridges 158

N

NCS
target and distribution library data sets 94

P

PARMLIB member for VSM 146

R

recovery utility
MVC access 158
VTV access 158
routing data sets to VSM 160
RTDs
defining for sharing with MVS 21
LIBGEN definitions 21
planning for 21
policies 59, 79, 80
UNITATTR control statement 21
RTV utility
MVC access 158
VTV access 158

S

scratch subpools
for VTVs 19
security
defining for VSM 157, 158
planning for 24
SLSUX15 user exit
issuing command authorization request to security product 158
SLUADMIN utility
unscratching current scratch cartridges 158
SMC/DFSMS interface 14
SMF record formats 147
SMFPRMxx, producing SMF records 147
space reclamation
MVCs 74, 76, 77, 78
SPNUM statement 148, 155

T

tape management system
planning for 19, 24, 49
updating 156
TAPEREQ control statement 14
TAPEVOL security class 157
target library data sets, allocating 94

U

UNITATTR control statement 12, 21
UNITMAP statement 148

V

VIRTACS statement 148, 155
VOLATTR control statement
examples 115
VSM
candidate data sets 49
configuration
defining VSM security 157, 158
record 165
routing data sets to VSM 160
updating the tape management system 156
values 11
Nearline hardware requirements 7
policies
AMTs 53
hosts enabled for migration 71
maximum concurrent migration tasks 67
maximum VTVs per MVC 70
migration 58, 71
MVC space reclamation 77
RTDs 59, 79, 80
space reclamation 74
security 158
software and hardware prerequisites 5
VSM (Virtual Storage Manager), support 92, 105
VSM, connecting MVS/CSC clients 148
VSM, connecting non-MVS/CSC clients 155
VTCS
configuration values 11
installation
applying the VTCS and HSC base FMIDs 99
installing the HSC base FMID and maintenance 100
modifying the HSC startup procedure 102
overview 89
receiving the VTCS FMID 95
verifying installation materials 92, 105
pre-installation
configuring MVS unit addresses/esoterics 87
defining a security system user ID 86
setting the MIH value 88
SMF record formats 147
VTDs
configuring
esoterics 87
MVS device numbers 87
data chaining a read forward or write command 49
MVS device numbers 13

planning for 13
UNITATTR control statement 12
VTSSs
planning for 12
VTVs
defining to tape management system 19
migration
on dismount 58
policies 58, 67, 71
planning for 19
RTV access 158
scratch subpools 19