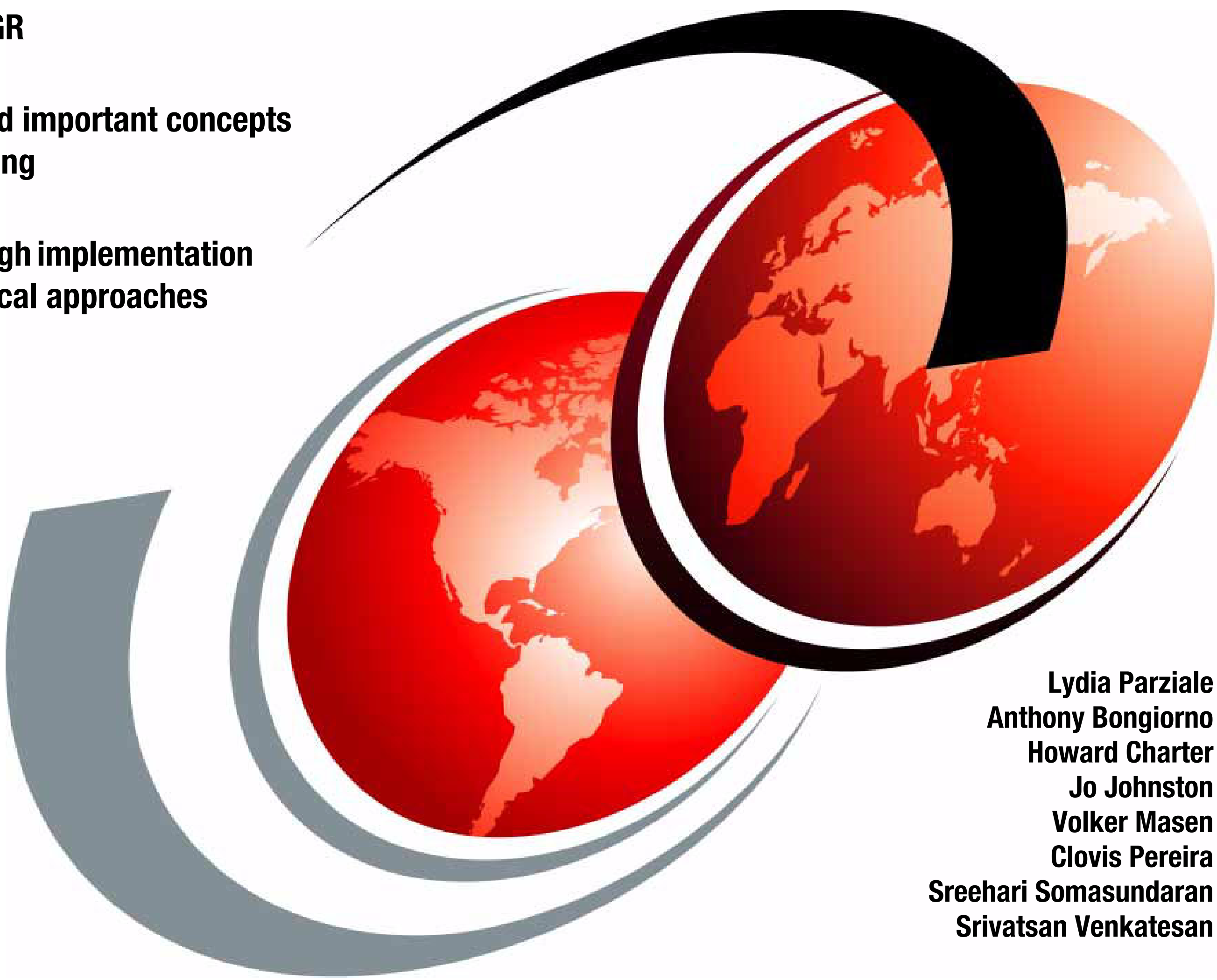
An introduction to z/VM Single System Image (SSI) and Live Guest Relocation (LGR)

Learn about the business benefits of SSI and LGR

Understand important concepts and planning

Step through implementation and practical approaches



Lydia Parziale
Anthony Bongiorno
Howard Charter
Jo Johnston
Volker Masen
Clovis Pereira
Sreehari Somasundaran
Srivatsan Venkatesan

Redbooks

ibm.com/redbooks

International Technical Support Organization

An introduction to z/VM Single System Image (SSI) and Live Guest Relocation (LGR)

April 2012

SG24-8006-00

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

First Edition (April 2012)

This edition applies to Version 6, Release 2, of z/VM.

Contents

Notices

This information was developed for products and services offered in the U.S.A.

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

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:
IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

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

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

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

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

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

COPYRIGHT LICENSE:

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

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at http://www.ibm.com/legal/copytrade.shtml

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

AIX®
CICS®
DB2®
ECKD™
FICON®
GDPS®
IBM®
IMS™
MVS™
RACF®
Rational®
Redbooks®
Redpapers™
Redbooks (logo) ®
System z10®
System z®
Tivoli®
VTAM®
WebSphere®
z/OS®
z/VM®
z/VSE®
z10™

The following terms are trademarks of other companies:

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

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

Preface

IBM® z/VM® 6.2 introduces significant changes to z/VM in the form of multi-system clustering technology allowing up to four z/VM instances in a single system image (SSI) cluster. This technology is important, because it offers clients an attractive alternative to vertical growth by adding new z/VM systems. In the past, this capability required duplicate efforts to install, maintain, and manage each system. With SSI, these duplicate efforts are reduced or eliminated.

Support for live guest relocation (LGR) allows you to move Linux virtual servers without disruption to the business, helping you to avoid planned outages. The z/VM systems are aware of each other and can take advantage of their combined resources. LGR enables clients to avoid loss of service due to planned outages by relocating guests from a system requiring maintenance to a system that remains active during the maintenance period.

Together, the SSI and LGR technologies offer substantial client value, and they are a major departure from past z/VM practices.

This IBM Redbooks® publication gives you a broad understanding of the new SSI architecture and an overview of LGR. In 5.7.2, "Relocating an SAP system in a typical System z setup" on page 79, we show an LGR example that shows a typical SAP user environment. In our example, the SAP Application Server Central Instance resides on a Linux on System z® guest and an IBM DB2® 10 database server runs on z/OS®.

This book is written for IT architects, who design the systems, and IT specialists, who build the systems.

The team who wrote this book

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

Lydia Parziale is a Project Leader for the ITSO team in Poughkeepsie, New York, with domestic and international experience in technology management including software development, project leadership, and strategic planning. Her areas of expertise include web business development and database management technologies. Lydia is a certified PMP and an IBM Certified IT Specialist with an MBA in Technology Management and has been employed by IBM for over 25 years in various technology areas.

Anthony Bongiorno is a Marist College Institute for Data Center Professional graduate and holds a System z Expert Certificate.

Howard Charter is a z Series Automation Specialist in IBM UK. He provides mainframe automation support on internal and outsourced accounts using IBM Tivoli® NetView with System Automation, OPSMVS, and AF/OPERATOR on z/OS, and IBM Operations Manager on z/VM. Howard has nearly 30 years experience with IBM in Systems Management, including automation, VM systems programming, and technical consultancy on security across all IBM platforms. Howard has previously co-authored IBM Redbooks publications on GDPS®.

Jo Johnston is a Certified IT Specialist and Chartered Engineer, who works in the IBM System z Strategic Outsourcing Team in the United Kingdom. She has worked on IBM mainframe systems as a systems programmer supporting z/VM, z/OS, IBM MVS™, IBM CICS®, DB2, and IBM IMS™ for more than 30 years. She joined IBM in 2001 working in Strategic Outsourcing Commercial Platform Support where she provided day-to-day z/OS, CICS, DB2, and IMS support for client systems that had been outsourced to IBM. She then moved to the IBM System z Technical Sales Team specializing in IBM WebSphere® Business Integration products on System z with specific responsibility for WebSphere Application Server, WebSphere Process Server, and WebSphere Service Registry and Repository. She now works in the System z Database team supporting not only DB2 and Adabas on z/OS, but also WebSphere Application Server on z/OS and Tivoli Storage Productivity Center for Replication on z/OS and z/VM. Jo has co-authored two IBM Redpapers™ on WebSphere Process Server on System z.

Volker Masen is a IT specialist in the IBM Development Lab in Boeblingen, Germany. He started his career 20 years ago in the System z environment, supporting the library and build management environment around ISPF/SCLM. After spending several years supporting development environments for the IBM Rational® brand, he moved back into the System z environment three years ago as a system programmer for z/VM in Boeblingen and specialist for other virtualization environments (VMware and KVM).

Clovis Pereira is a System Support Analyst in Brazil. He has about 28 years of experience in supporting IBM operating systems, the last 22 at IBM. His areas of expertise include IBM z/VM, z/OS, and z/VSE®. Today, he works in the Customer Support Center (CAC) in Brazil, helping clients to solve problems related to operating systems (z/VM, z/OS, and z/VSE) and related products. He has written extensively on Chapter 4, “Scenario one: Creating a single system image cluster with four new z/VM members” on page 35 and Appendix A, “Frequently asked questions” on page 87.

Sreehari Somasundaran is an IBM and Open Group Certified IT Specialist working with the Systems and Technology Group. He is a zChampion and the System z Techline Specialist for IBM India/South Asia. He holds an MBA in Marketing and Operations from the Indian Institute of Management (IIMK) and an Engineering Degree in Electronics and Communication from Kerala University. He joined IBM in 1997 and has worked in multiple roles handling System z application development, project management, infrastructure consulting, TCO/ROI Analysis, performance analysis and capacity planning studies, and has managed multiple benchmarks and proofs of concept. His areas of expertise include System z hardware, z/OS, and z/VM. He is a regular speaker at the System z Technical conferences in India.

Srivatsan Venkatesan is an IT Specialist in the Systems and Technology Group in the US. He has one year of experience in the z/VM and Linux on System z fields. He holds a degree in Computer Science from the University of South Carolina. His areas of expertise include Linux and middleware on System z.

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

Bill Bitner of the z/VM Customer Focus and Care team

The z/VM Architecture and Development team:
Emily K Hugenbruch, John Franciscovich, Mark Lorenc, Michael Wilkins, Jim Switzer, Ken Davis, Susan Farrell, and Carol Everitt

Roy Costa and Robert Haimowitz
International Technical Support Organization, Poughkeepsie Center

Oliver Petrik
IBM Boeblingen

Now you can become a published author, too!

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

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

ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

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

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

  ibm.com/redbooks

- Send your comments in an email to:

  redbooks@us.ibm.com

- Mail your comments to:

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

Stay connected to IBM Redbooks

- Find us on Facebook:

  http://www.facebook.com/IBMRedbooks

- Follow us on Twitter:

  http://twitter.com/ibmredbooks

- Look for us on LinkedIn:

  http://www.linkedin.com/groups?home=&gid=2130806

- Explore new Redbooks publications, residencies, and workshops with the IBM Redbooks weekly newsletter:

  https://www.redbooks.ibm.com/Redbooks.nsf/subscribe?OpenForm

- Stay current on recent Redbooks publications with RSS Feeds:

  http://www.redbooks.ibm.com/rss.html

1

Introduction

We are in an era of information technology in which enterprises address the challenge of an ever-increasing demand for IT resources and tighter budgets. IBM smarter computing addresses this challenge through IT transformation.

Virtualization is a key element of this transformation. z/VM Version 6.2 provides virtualization enhancements that help reduce sprawling IT. It helps provide flexibility to meet user demands for security, availability, scalability, and the manageability of IBM System z.

This chapter provides a brief explanation of the business benefits obtained when using IBM z/VM Version 6.2 with single system image (SSI) and live guest relocation (LGR). Additionally, we provide navigation of the z/VM library with a brief abstract of the purpose of each volume, and where these books have additional information about z/VM V6.2.

1.1 Business benefits

This section outlines the business benefits realized with z/VM Version 6.2 using single system image (SSI) with live guest relocation (LGR):

- Simplified z/VM systems management.
- Ability to share all system resources with high levels of resource utilization.
- Software or hardware maintenance and upgrades can be performed without disruption to the business, providing continuous availability.
- Concurrent support for virtual machines running separate operating systems and a secure isolated environment.

1.1.1 Simplified z/VM systems management

When z/VM multiple system virtualization with SSI is used to create clusters, the members can be serviced, managed, and administered as though they are one integrated system. The coordination of members joining and leaving the cluster, the maintenance of a common view of cluster member and resource states, and the negotiated access to shared cluster resources are all done seamlessly.

The z/VM multi-system virtualization helps clients avoid virtual machine sprawl challenges. These challenges include the creation of uncontrolled numbers of virtual machines that IT managers do not know about, whether they are up or down, and whether they are secure.

1.1.2 Share system resources with high resource utilization

Sharing all system resources with high levels of resource utilization is extended with z/VM V6.2. Within SSI, resources that are used by the z/VM hypervisors and the virtual machines are shared. The shared resources are managed as a single resource pool and provide a more manageable infrastructure. Resources include the user directories, minidisks, spool files, network device Media Access Control (MAC) addresses, and security definitions, if implemented.

Sharing resources among members improves the integrity and performance of the system. Through resource sharing, virtual servers have access to the same devices and networks, regardless of which z/VM member they are logged on to within the SSI cluster. Sharing resources allows service to be rolled out to each member of the cluster on individual schedules, avoiding an outage for the entire cluster.

High levels of resource sharing within z/VM include the sharing of Linux program executables and file systems.

1.1.3 Non-disruptive maintenance and upgrades

Live guest relocation (LGR) is a process where a running Linux virtual guest can be relocated from one z/VM member system in an SSI cluster to another member. Guests can be moved to other members that are on the same or separate System z servers without disruption to the business. Virtual Linux guests can even be moved across the System z family between IBM System z10® EC, IBM System z10 BC, IBM z196, and IBM z114. This flexible manual workload balancing allows work to be moved non-disruptively to available system resources. The business benefit is the reduction of the impact of planned z/VM outages when performing the z/VM software or hardware maintenance. This reduced impact delivers the application continuity that clients require and is an important factor contributing to an optimized system.

1.1.4 Concurrent support for disparate environments

z/VM provides concurrent support for virtual machines running separate operating systems in a secure isolated environment. z/VM supports z/OS, z/VSE, z/TPF, and Linux on System z operating environments in addition to supporting the CMS application development platform.

The ability to support multiple machine images and architectures enables z/VM to run multiple production and test versions of the System z operating systems. The versions all run in the same System z server, providing a highly flexible test and production environment. The test environment can be architected to reflect the server production environment. The test environment can help simplify migration from one release to another and facilitate the transition to newer applications, providing a test system whenever one is needed. The efficient use of shared resources results in z/VM utilization of nearly 100% of available system resources nearly 100% of the time.

1.2 Navigating the z/VM library

Because of the many changes brought about with SSI and LGR, many of the z/VM books are updated. This section provides an abstract for each book in the z/VM library that is modified to include SSI and LGR.

1.2.1 z/VM V6R2.0 General Information

Information in *z/VM V6R2.0 General Information*, GC24-6193-02, is intended for anyone who wants a general overview of z/VM. It is also useful for individuals who need to evaluate the capabilities of z/VM and determine the necessary resources to install and run it.

The following information is in this manual:

- How z/VM can help you
- z/VM overview
- What is new or changed in z/VM V6.2
- z/VM hardware and software requirements, including device and storage support and requirements, performance considerations for operating systems supported as guests, and other programs supported on z/VM

A description of z/VM SSI clusters and LGR, new in z/VM V6.2, is in Chapter 3 of this manual. A description of the enhancement to the directory maintenance facility to support z/VM SSI clusters is also included in this chapter.

Technical information in Chapter 4 of this manual includes SSI cluster hardware requirements and SSI cluster program requirements.

You can access this document at this website:

http://publibz.boulder.ibm.com/epubs/pdf/hcsf8c10.pdf

1.2.2 z/VM V6R2.0 Installation Guide

Information in the *z/VM V6R2.0 Installation Guide*, GC24-6246, guides you through the installation of Version 6 Release 2 of IBM z/VM through installation procedures. The procedures cover the installation of a z/VM system first-level (in a processor's logical partition) or second level (as a guest operating system hosted by z/VM) from tape or DVD media.

A number of enhancements were made to the z/VM installation process. The process supports the installation of either a non-SSI (traditional) z/VM system or an SSI cluster consisting of from one to four members. The installation procedure is restructured so that all planning information is gathered at one time and installation is initiated with a single command. This change minimizes the chance of errors, because the planning information is validated before the actual installation begins. Additionally, clients are now able to specify labels for all DASD volumes, including the system residence volume.

A description for the SSI tape installation method is in Chapter 4, where a procedure is outlined to continue to install the first-level SSI in a new system environment from the z/VM system installation tape. Or, you can install the SSI second level on an existing z/VM system from the z/VM system installation tape. A description for the SSI DVD installation method is in Chapter 9 where you find a procedure to install a one-member or multi-member z/VM SSI cluster.

You can access this document at this website:

http://publibz.boulder.ibm.com/epubs/pdf/hcsk2c10.pdf

1.2.3 z/VM V6R2.0 CP Planning and Administration

Information in *z/VM V6R2.0 CP Planning and Administration*, SC24-6178-02, is for anyone responsible for planning, installing, and updating a z/VM system.

This manual has six parts. Part 5 was added and others changed to include z/VM SSI clusters and LGR. Part 1 of this book has a new section for planning for multisystem environments. Part 2 has new SYSTEM CONFIG file statements for SSI/LGR in Chapter 6. Part 3 has new user directory statements for SSI/LGR in Chapter 17. Part 4 has a new Chapter 21 about how the control program (CP)-owned list is process in the SSI/LGR environment.

Part 5 has information for setting up z/VM single system image clusters in Chapter 25, cross-system spool in a z/VM SSI cluster in Chapter 26 and preparing for guest relocations in a z/VM SSI cluster in Chapter 27. Chapter 28 - Chapter 33 contain detailed scenarios for migrating to an SSI environment.

You can access this document at this website:

http://publibz.boulder.ibm.com/epubs/pdf/hcsg0c10.pdf

1.2.4 z/VM V6R2.0 Getting Started with Linux on System z

z/VM V6R2.0 Getting Started with Linux on System z, SC24-6194, is not only about Linux. It has a good description of z/VM, and detailed instructions about how to customize z/VM after installation, including the Linux parts. Information in this manual is designed to help anyone who is a system programmer, administrator, or operator, but has limited knowledge of z/VM and wants to get started deploying Linux servers on z/VM.

Information in this manual describes how to configure and use z/VM functions and facilities for Linux servers running on the IBM System z platform. The document provides requirements and guidelines to implement during z/VM installation, but primarily assumes that you have installed z/VM and are ready to deploy Linux in virtual machines.

Early sections acquaint you with z/VM and take you from the point after z/VM installation to the point where you are ready to install your first Linux server. You must turn to the installation documentation that is provided by your Linux distributor. Following the deployment of your first Linux server, you can replicate or clone additional servers. When you finish the early sections, you have two or more Linux servers running on z/VM with TCP/IP connections to the network. Subsequently, you can turn to vendor-supplied documentation to install applications on Linux. Later sections cover operations, administration, performance, guest relocation, and other day-to-day bare essentials.

See Section Planning for SSI clusters and LGR in Chapter 2 for information about additional planning guidelines for an SSI cluster and LGR.

You can access this document at this website:

http://publibz.boulder.ibm.com/epubs/pdf/hcsx0c10.pdf

1.2.5 z/VM V6R2.0 VMSES/E Introduction and Reference

Information in *z/VM V6R2.0 VMSES/E Introduction and Reference*, GC24-6243-01, is intended for anyone responsible for installing, migrating, building, deleting, or servicing products on z/VM and individuals managing the z/VM software inventory.

VMSES/E is enhanced to support SSI clusters. Product service disks and inventory are shared by all member systems in an SSI cluster. Each member of the cluster has its own production disks, allowing flexibility for placing new service into production in a staged fashion. With the SSI feature, Linux guests can be moved from one SSI member to another ahead of most planned outages, such as outages required for service and hardware upgrades. Without the relocation capability, an outage for the guest is necessary.

VMSES/E is enhanced to record the serviced objects and copy only those serviced objects to the appropriate production disks. Previously, entire disks were copied. This change is less disruptive to the running system. A new release-specific maintenance user ID is supplied for the cluster. This user ID owns the product service disks, including the test build disks.

To support SSI clusters, many functions described in this document are updated and new functions are added. See “Support for z/VM Single System Image Clusters” in the Summary of Changes for more information.

You can access this document at this website:

http://publibz.boulder.ibm.com/epubs/pdf/hcsc6c10.pdf

1.2.6 z/VM V6R2.0 Directory Maintenance Facility Commands Reference

Information in *z/VM V6R2.0 Directory Maintenance Facility Commands Reference,* SC24-6188-02, is intended for those persons responsible for creating and maintaining the VM directory. Certain functions in this document are also available to general users, allowing them to implement limited changes to their own directory entries.

The Directory Maintenance Facility for z/VM is a priced feature of z/VM. The Directory Maintenance Facility for z/VM, function level 620, is enhanced to support SSI clusters. New capabilities include support for the enhanced directory syntax for SSI, conversion of directory contents to aid transition to an SSI cluster, and assistance in adjusting the directory when adding a member to the cluster.

To support SSI clusters, several DIRMAINT commands are updated and three new operands are added:

- SSI operand of the DIRMAINT command to prepare a source directory to use on a node in an SSI cluster.
- UNDOSSI operand of the DIRMAINT command to roll back BUILD statement changes made by the SSI operand and to remove the SSI operand from the DIRECTORY statement.
- VMRELOCATE operand to query, update, or delete the relocation capability that is associated with a user or profile entry.
- ADD operand is updated for cloning SUBCONFIG entries. A new SSI_VOLUMES section is added to the EXTENT CONTROL file to support the ADD operand.

You can access this document at this website:

http://publibz.boulder.ibm.com/epubs/pdf/hcsl4c10.pdf

1.2.7 z/VM V6R2.0 Connectivity

Information in *z/VM V6R2.0 Connectivity*, SC24-6174-02, is intended for anyone responsible for planning, setting up, running, and maintaining z/VM connectivity facilities. Information in this manual provides an introduction to the facilities of IBM z/VM that allow z/VM systems, application programs, and guests to establish logical connections to other systems, application programs, and guests. It describes the following z/VM connectivity facilities:

- Transmission Control Protocol/Internet Protocol (TCP/IP) for z/VM
- z/VM virtual networks (guest LANs and virtual switches)
- Advanced Program-to-Program Communications (APPC)
- Transparent Services Access Facility (TSAF)
- APPC/VM VTAM® Support (AVS)
- Inter-System Facility for Communications (ISFC)

To support SSI clusters, many functions that are described in this document are updated and new functions have been added. One example is virtual networking support for an SSI cluster. This feature extends the z/VM Ethernet virtual switch logic to coordinate its automatic MAC address assignment with all active members of an IBM z/VM SSI feature.

The following sections are new for SSI support:

- Chapter 4, “Single System Image MAC Address Considerations,” on page 50
- Chapter 6, “Live Guest Relocation Networking Considerations,” on page 87

You can access this document at this website:

http://publibz.boulder.ibm.com/epubs/pdf/hcsc9c10.pdf

1.2.8 z/VM V6R2.0 CP Commands and Utilities Reference

z/VM V6R2.0 CP Commands and Utilities Reference, SC24-6175-02, provides detailed reference information about Control Program (CP) commands and system utilities for users of every privilege class. System utilities perform CP functions but operate only in the CMS environment.

Information is added about new and changed CP commands and utilities in support of SSI cluster configuration and management. The following CP commands are added or changed for this version and release (this list is not exhaustive):

- AT
- DEFINE RELODOMAIN
- MSG has a new AT operand.
- QUERY CHPIDV
- QUERY NAMES has a new AT operand.
- QUERY RELODOMAIN
- QUERY SSI
- QUERY VIRTUAL CHPID
- QUERY VMRELOCATE
- SET VMRELOCATE
- SET SSI
- SET CPTRACE changed. New trace codes for SSI cluster operations are added.
- VMRELOCATE
- WNG has a new AT operand.

The following CP utilities are added or changed for this version and release:

- FORMSSI is new.
- CPSYNTAX is changed. A logical partition (LPAR) operand is added, and enhancements are added to support the new SYSTEM CONFIG file statements SSI, BEGIN, and END.

See "Support for z/VM Single System Image Clusters" in the Summary of Changes of this manual for information about commands and utilities:

- Cross-system SPOOL and SCIF
- ISFC infrastructure enhancements
- LGR
- Multiple access ports per guest
- Real device mapping
- Shared disk enhancements
- SSI user identity and configuration
- Virtual networking support for an SSI cluster

You can access this document at this website:

http://publibz.boulder.ibm.com/epubs/pdf/hcse4c10.pdf

1.2.9 z/VM V6R2.0 Migration Guide

z/VM V6R2.0 Migration Guide, GC24-6201, describes the addition of the IBM z/VM Single System Image Feature (VMSSI). This priced feature of z/VM V6.2 enables the creation of z/VM SSI clusters. It provides three types of information:

- It describes enhancements and changes to the z/VM system that you must be aware of before migrating.
- It identifies specific external interfaces that have changed and provides an assessment of the compatibility of each change, upwardly compatible or incompatible.

- It provides guidance and procedures for certain migration tasks that you might need to perform.

Topics on SSI are included in this book:

- System changes: The chapter is redesigned. It now lists subchapters preceded by their z/VM version and release level. It is much easier to look up a task and then isolate your specific needs by version level:
  - Installation, migration, and service

    Service changes to support the SSI environment
  - Support and exploitation of hardware and architectures

    Information on SSI cluster toleration within an ensemble
  - Connectivity and networking

    Virtual networking support for an SSI cluster
  - System administration and operation:
    - SSI cluster configuration and management: A list of added or updated functions
    - SSI cluster user identity and configuration: Descriptions of a new type of virtual machine definition, changes to the layout of the system minidisks, and changes to the MAINT user ID
    - A description of the changes implemented to provide LGR in an SSI cluster
- An important appendix (Appendix C) that explains the importance of upgrading a CSE system to z/VM V6.2

You can access this document at this website:

http://publibz.boulder.ibm.com/epubs/pdf/hcsf9c10.pdf

2

Single system image (SSI) overview

This chapter provides an overview of multi-system virtualization with z/VM Single System Image (VMSSI) features and operations. It describes the difference between a z/VM SSI cluster and a stand-alone non-SSI z/VM system.

2.1 Introduction to the z/VM single system image (SSI) clusters

The z/VM Single System Image feature (VMSSI) is an optional priced feature that is new with z/VM Version 6.2. It enables up to four z/VM systems to be configured as members of an SSI cluster, sharing the following resources:

- User directory
- DASD volumes
- User minidisks
- Spool files
- Network devices

VMSSI cluster members are not restricted to only one physical frame (CPC) or logical partition (LPAR). The cluster can span multiple LPARs across multiple CPCs. Clusters are however restricted to contain only homogeneous members. This means z/VM systems installed as second-level may not be clustered with any installed as first-level. Each cluster must consist of all first-level or all second-level.

Important: The z/VM Single System Image feature enables sharing of configuration, parameters, and directory data over Channel-to-channel adapter (CTCA) connections. During the z/VM installation process, when you choose to install a system as second-level, there are customized modifications made to the generated system configuration and user directory parameters. Due to these differences, you must not create a cluster which contains a mix of first-level and second-level z/VM members. Attempting to do so will result in unpredictable or catastrophic results. Additional details are covered in z/VM CP Planning and Administration (Version 6 Release 3, SC24-6178-05)

SSI also introduces the concept of live guest relocation (LGR) where a running Linux guest operating system can be relocated from one member in an SSI cluster to another without the need to stop the running Linux guest. This capability provides continuous availability during planned outages for software or hardware maintenance and allows system workload balancing. It also restarts your z/VM system without causing disruption to the Linux guests within the cluster. We describe LGR more fully in Chapter 3, “Live guest relocation (LGR) overview” on page 23.

2.2 Major attributes of the z/VM SSI cluster

This section describes the major characteristics and features of an SSI cluster and changes that have been made to z/VM to accommodate this SSI cluster.

2.2.1 z/VM installation

An SSI cluster can be created with a single z/VM installation. That is, up to four members can be created from the initial installation media at the same time, as shown in Chapter 4, “Scenario one: Creating a single system image cluster with four new z/VM members” on page 35.

A traditional stand-alone non-SSI z/VM system can also be created from the installation media, which we initially did in Chapter 5, “Scenario two: Converting two existing stand-alone z/VM systems to a single system image cluster” on page 61 prior to converting them to an SSI cluster.

2.2.2 Changes to applying service

Planned software or hardware maintenance can be applied to a single member without affecting the other members of the cluster or disrupting running Linux guest operating systems. The Linux guests can be relocated to another member in the cluster prior to maintenance being applied to the original member that hosted the Linux guest.

Systems Management is simplified in an SSI cluster. Service can be applied to z/VM from any member within the cluster, but it needs to be put into production on each member. In addition, members can coexist with other members running a different service level of z/VM 6.2. Service updates can be planned over a period of time and can be applied to a single member of the cluster, and then rolled out to other members of the cluster later.

Figure 2-1 shows how service is applied to an SSI cluster.

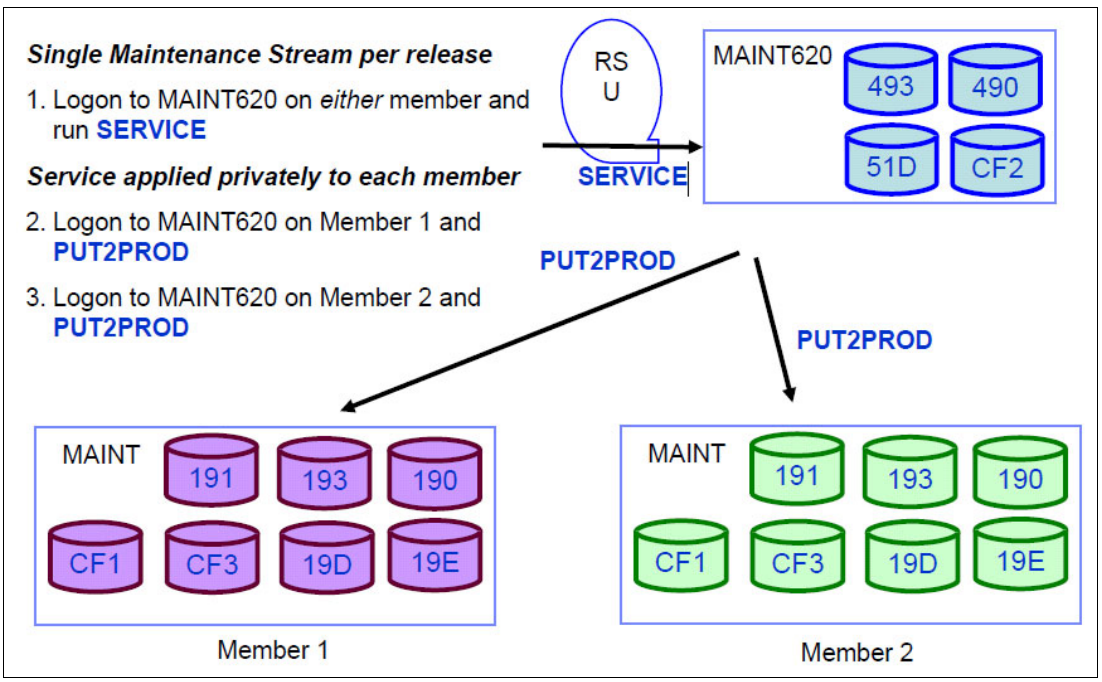


Figure 2-1 Applying service

2.2.3 Workload balancing

SSI facilitates workload balancing, because Linux guests can be manually relocated to members that are under-utilized.

2.2.4 Single configuration users and multiconfiguration users

Single configuration users are z/VM users in the traditional sense They are defined in the user directory with USER entries and can only log on to one member in the cluster at a time. They have the same definitions and access to the same resources on all members in the SSI cluster. Typically, USERs are guest operating systems or service virtual machines that require only one logon in the cluster.

Multiconfiguration users are IDENTITY entries, which are new to SSI clusters, in the user directory. They can log on to multiple members in the cluster at the same time. Like USERs, they have definitions and resources, which are common to all members. In addition, they have definitions and resources, which are unique to the particular member they are logged on to. These definitions and resources are the SUBCONFIG statements. Typically, IDENTITY directory entries are for system support and service virtual machines.

Figure 2-2 on page 12 shows the types of users in the directory.

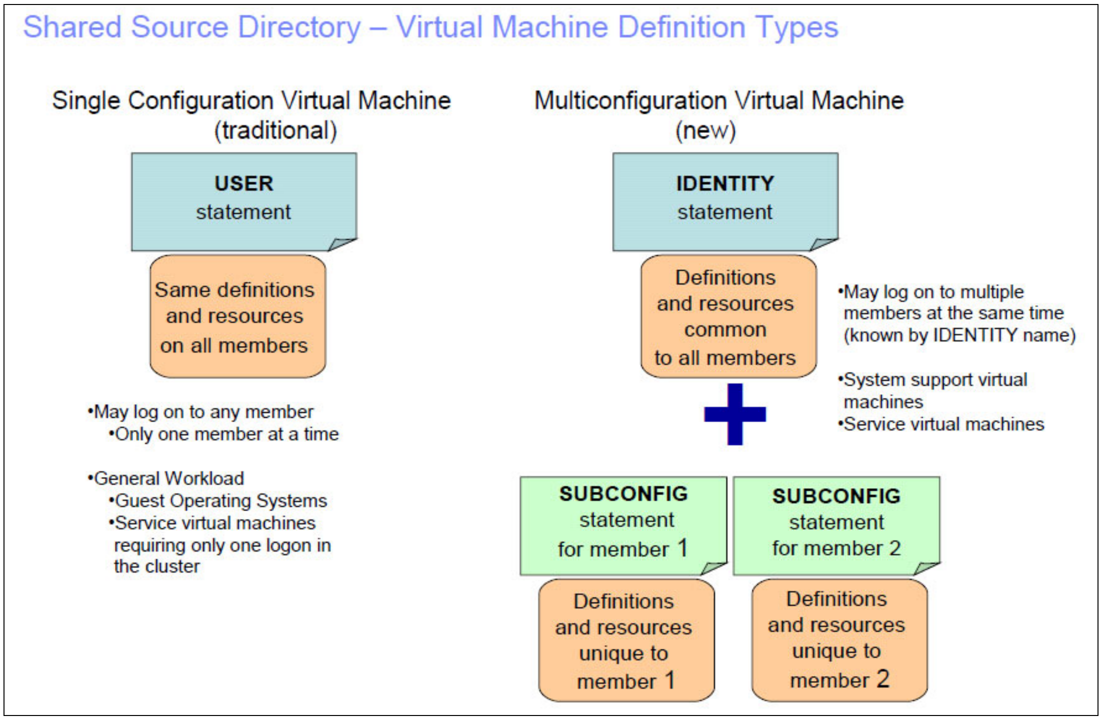


Figure 2-2 Virtual Machine Definitions

2.2.5 DASD requirements

For an SSI cluster, common disks are shared by all members of the cluster:

- Common volumes: Contain the SYSTEM CONFIG file, VMSES/E, and common user directory
- Release volumes: Contain one set of disks per z/VM release per cluster
- Spool volumes: Require at least one spool volume for each member of the cluster and the spool volumes are owned by that member

Spool volumes: A member can only create spool files on volumes that it owns, but all members can view and update files on all spool volumes.

Each member also has its own volumes:

- Sysres volume
- Paging volume
- Private user volumes

Figure 2-3 on page 13 shows the shared and non-shared system volumes.

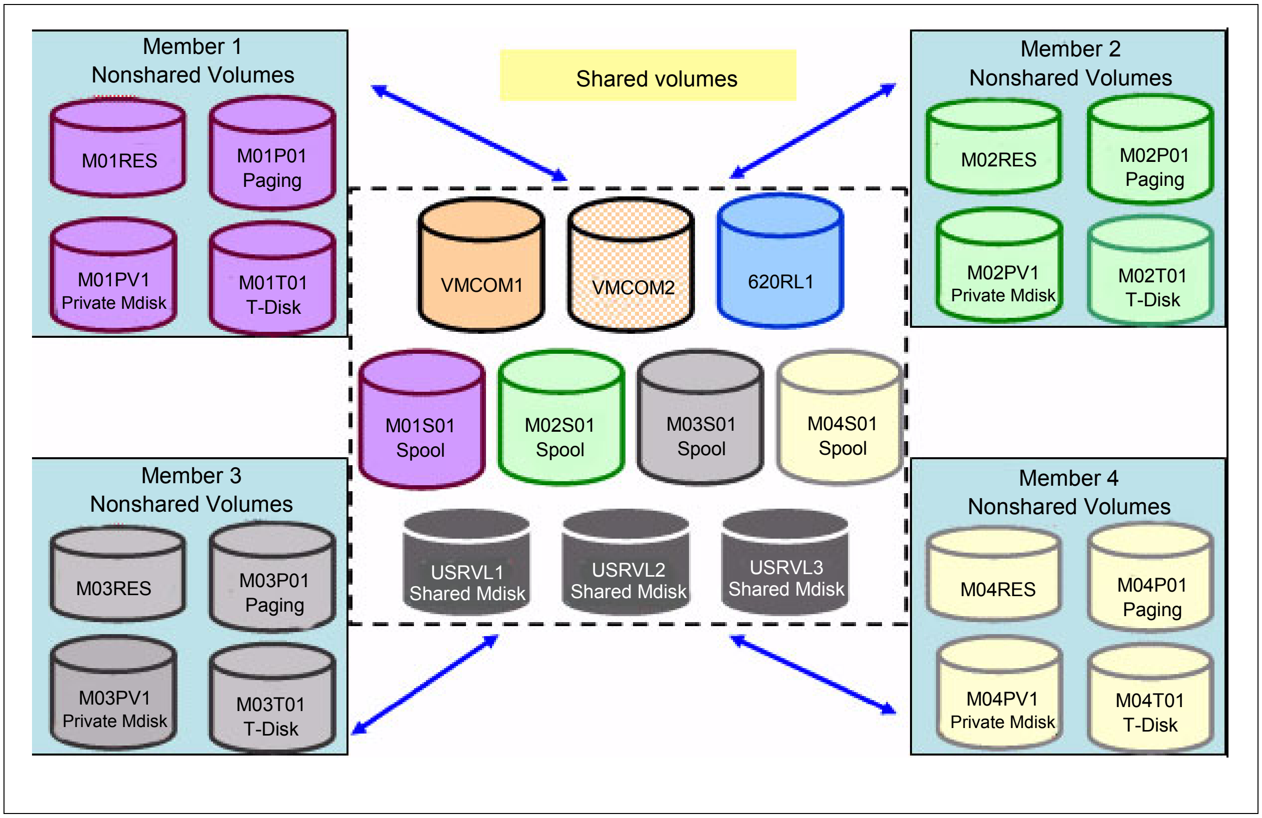


Figure 2-3 DASD planning: Non-shared and shared system volumes

To prevent members from allocating control program (CP) data on a volume owned by another member, CPFMTXA has been modified to mark CP-owned volumes with ownership information, SSI cluster, and owning SSI member. Example 2-1 shows the ownership information of a non-shared CP-owned volume SSI5RS on our ITSOSSI5 system.

Example 2-1 Non-shared CP-owned volume

```
cpfmtxa 123 ssi5rs alloc
HCPCCF6209I INVOKING ICKDSF.
ICK030E DEFINE INPUT  DEVICE: FN FT FM, "CONSOLE", OR "READER"
CONSOLE
.
.
.
ICK03000I CPVOL REPORT FOR 0123 FOLLOWS:
ICK03021I 0123 IS FORMATTED FOR VM/ESA MODE

          CYLINDER ALLOCATION CURRENTLY IS AS FOLLOWS:
          TYPE     START      END        TOTAL
          ----     -----      ---        -----
          PERM     0          0          1
          DRCT     1          20         20
          PERM     21         38         18
          PARM     39         158        120
          PARM     159        159        1
```

```
         PARM     160        279        120
         PERM     280        10016      9737
ICK03092I VOLUME VM SSI OWNER = ITSOSSIB
ICK03093I VOLUME SYSTEM OWNER = ITSOSSI5
ICK00001I FUNCTION COMPLETED, HIGHEST CONDITION CODE WAS 0
```

Most CP-owned volumes that are shared by the cluster have ownership of the SSI cluster and NOSYS. However, this ownership does not apply to all CP-owned volumes. For example, spool volumes have ownership of the SSI cluster and the member. Example 2-2 shows the ownership information of the common volume SSIBC1 on our ITSOSSIB cluster.

Example 2-2 CP-owned volume shared by the cluster

```
cpfmtxa 141 ssibc1 alloc
HCPCCF6209I INVOKING ICKDSF.
ICK030E DEFINE INPUT  DEVICE: FN FT FM, "CONSOLE", OR "READER"
CONSOLE
.
.
.
ICK03000I CPVOL REPORT FOR 0141 FOLLOWS:
ICK03021I 0141 IS FORMATTED FOR VM/ESA MODE

         CYLINDER ALLOCATION CURRENTLY IS AS FOLLOWS:
         TYPE     START      END        TOTAL
         ----     -----      ---        -----
         PERM     0          0          1
         PARM     1          120        120
         PERM     121        10016      9896
ICK03092I VOLUME VM SSI OWNER = ITSOSSIB
ICK03093I VOLUME SYSTEM OWNER = (NO OWNER ASSIGNED)
ICK00001I FUNCTION COMPLETED, HIGHEST CONDITION CODE WAS 0
```

2.2.6 Network requirements

Network planning is a key requirement to setting up the cluster. The channel-to-channel (CTC) connections provide the means of communications between the members of the cluster. Only two CTCs can be defined for each member at installation, but more can be added later. Figure 2-4 on page 15 shows the CTC connections for an SSI cluster. It shows that every member of the cluster uses CTC connections to communicate with every other member of the cluster.

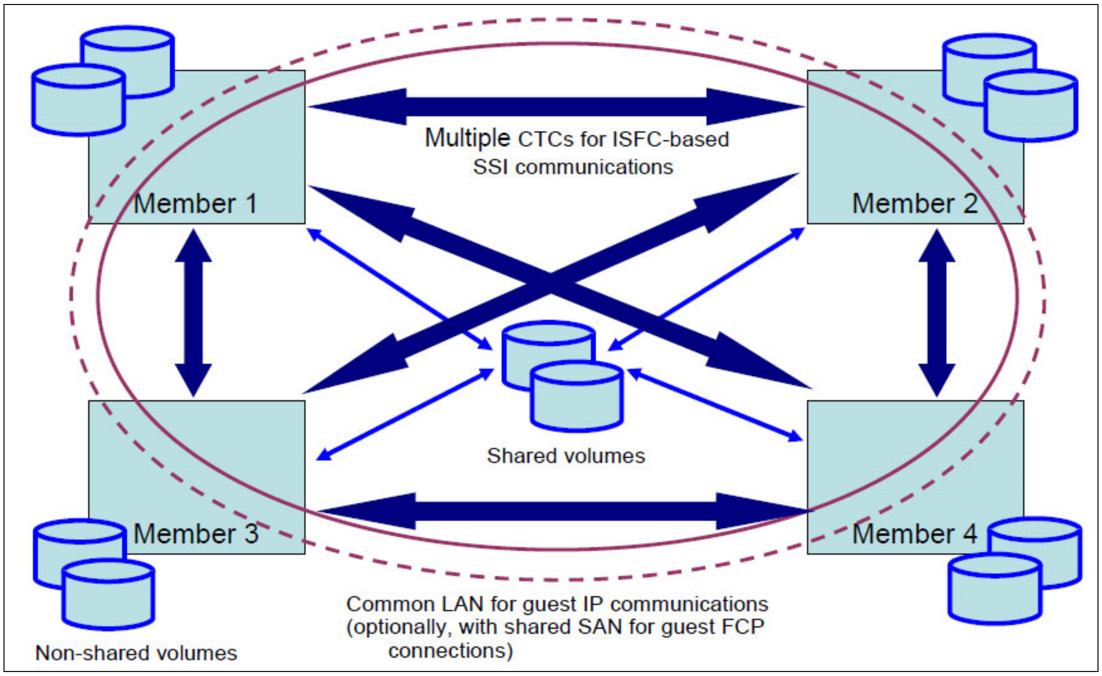


Figure 2-4 SSI cluster connections

2.2.7 Equivalency identifiers

Similar to the considerations for DASD, special considerations exist for networking, too. A mechanism needs to exist to uniquely identify real devices within an SSI cluster. You need to carefully plan and set up CTC connections, Fibre Channel Protocol (FCP), Hipersockets, and Open Systems Adapter (OSA) devices. Each member of the SSI cluster must have identical network connectivity in terms of the virtual switch name, having one or more physical OSA ports connected to the same physical LAN segment.

z/VM 6.2 uses equivalency identifiers (EQIDs) so that real devices can be uniquely identified within an SSI cluster. Also, z/VM 6.2 uses EQIDs to ensure that all members of the cluster use the same physical devices and devices attached over Fibre Channel connection (FICON®).

All physical DASD and tape drives have an EQID that is automatically generated by CP. A physical device has the same EQID across all members of the cluster.

Network adapters, FCP, Hipersocket, and OSA devices also have EQIDs, but these EQIDs must be defined by the system administrator. This way, each member of the SSI cluster has the same network connectivity and the same VSWITCH definitions with the same names.

Example 2-3 shows the OSA1 and VSWITCH definitions in the SYSTEM CONFIG file for z/VM members ITSOSSI5 and ITSOSSI6. The same named virtual switches **1** on different members must have physical OSA ports connected to the same physical LAN segment **2**.

Example 2-3 SYSTEM CONFIG file definitions for ITSOSSI5 and ITSOSSI6

```
ITSOSSI5: Begin
    RDEVICE 3080-308F  EQID OSA1 TYPE OSA  2
    DEFINE VSWITCH VSWITCH1    RDEV 3083   1
    DEFINE VSWITCH VSW999 RDEV 3086 IP
    DEFINE VSWITCH VSW199 RDEV 3089 IP
```

```
ITSOSSI5: End


ITSOSSI6: Begin
   RDEVICE 2100-210F  EQID OSA1 TYPE OSA   2
   DEFINE VSWITCH VSWITCH1    RDEV 2103    1
   DEFINE VSWITCH VSW999 RDEV 2106 IP
   DEFINE VSWITCH VSW199 RDEV 2109 IP
ITSOSSI6: End
```

The EQID concept also helps determine which devices are eligible for relocation to ensure data integrity.

2.2.8 MAC addresses

Virtual networking management is another key planning item. The assignment of MAC addresses to network interface cards (NICs) is coordinated across the SSI cluster. Even if a Linux guest relocates across the cluster, it is accessible without any operational disruption. SSI does not allow the members of the SSI cluster to have a MAC address that is already in use by another Linux guest within the cluster.

The MAC addresses are combined from the following attributes:

- MACPREFIX

  MACPREFIX specifies the 3-byte prefix (manufacturer ID) that is used when CP automatically generates locally administered MAC addresses on the system. It must be six hexadecimal digits in the range 020000 - 02FFFF. In combination with the MAC ID that is automatically assigned by CP, the MACPREFIX allows unique identification of virtual adapters within a network:

  - For a non-SSI system: If MACPREFIX is not specified, the default is 020000 (02-00-00).
  - For an SSI member:
    - The MACPREFIX value must differ for each member system.
    - The default MACPREFIX for a cluster member is 02*xxxx* (02-*xx*-*xx*), where *xxxx* is the slot number assigned to the system in the SSI member list on the SSI configuration statement.

- USERPREFIX

  USERPREFIX specifies the 3-byte prefix that is used when user-defined locally administered MAC addresses are generated. It must be six hexadecimal digits in the range 020000 - 02FFFF.

  The USERPREFIX value must be identical for all members of the cluster, so the same MAC address is created when the virtual machine is logged on to any member.

  The USERPREFIX value for the cluster cannot be the same as the MACPREFIX value for any member. The default USERPREFIX value in an SSI cluster is 020000.

- MACID

  This MACID (three bytes) is appended to the system MACPREFIX or USERPREFIX (three bytes) to form a unique MAC address for the virtual switch. If no MACID is set for the virtual switch, CP generates a unique identifier for this virtual switch.

In many cases, you can use the default values for the attributes MACID, MACPREFIX, and USERPREFIX, because z/VM ensures that the MAC addresses are unique. If you require a predictable MAC address assignment for virtual NICs on your system, you must declare MACPREFIX and USERPREFIX statements in the SYSTEM CONFIG. Nevertheless, we suggest that you define at least a unique MACPREFIX for each system to ensure that you have unique MAC addresses across your environment. See Example 2-4 for an example of setting these attributes.

Example 2-4 Define MACPREFIX and USERPREFIX in SYSTEM CONFIG

```
ITSOSSI5: VMLAN MACPREFIX 025555 USERPREFIX 02AAAA
ITSOSSI6: VMLAN MACPREFIX 026666 USERPREFIX 02AAAA
```

For more information about MAC address definition, see *z/VM V6R2.0 CP Planning and Administration*, SC24-6178-02.

2.2.9 New user IDs for z/VM 6.2

In z/VM 6.2, there are now three MAINT user IDs, product-specific user IDs, and a *shared file system* (SFS) manager user ID:

- MAINT: MAINT is a multiconfiguration virtual machine (IDENTITY) that is used for working on a a particular member of the SSI cluster. It can be used to attach devices or relocate guests. MAINT owns the CF1 and CF3 parm disks apart from the 190, 193,19D, 19E, 401, 402, and 990 CMS disks. It is no longer used for applying service to the z/VM software.
- PMAINT: PMAINT is a single configuration virtual machine (USER) that is used for updating the SYSTEM CONFIG file or for defining *relocation domains* for the SSI cluster. PMAINT includes the following cluster-wide minidisks, which are part of the common volume (default label VMCOM1):
  - PMAINT CF0: Parm disk that contains the common SYSTEM CONFIG file
  - PMAINT 2CC: Common source directory
  - PMAINT 41D: VMSES/E production inventory disk
  - PMAINT 551: PMAINT 551 contains common utilities, such as CPFMTXA, DIRECTXA, DIRMAP, and DISKMAP
- MAINT620: Service for release z/VM 6.2.0 is loaded on one or two release-specific volumes (when installing on 3390 Model 9 DASDs as we did, the default label for the single release-specific volume is 620RL1). MAINT620 is a single configuration virtual machine (USER) that is used for applying z/VM 6.2.0 service. It includes the following service disks:
  - MAINT620 490:Test CMS system disk
  - MAINT620 493:Test system tools disk
  - MAINT620 51D:VMSES/E software inventory disk
  - MAINT620 CF2: Test parm disk
- VMSERVP: VMSERP is a single configuration virtual machine (USER) that is used to manage the *new shared file system*. It is described in detail in *z/VM V6R2.0 CP Planning and Administration*, SC24-6178-02.

Figure 2-5 on page 18 shows the DASD allocation for an SSI cluster.

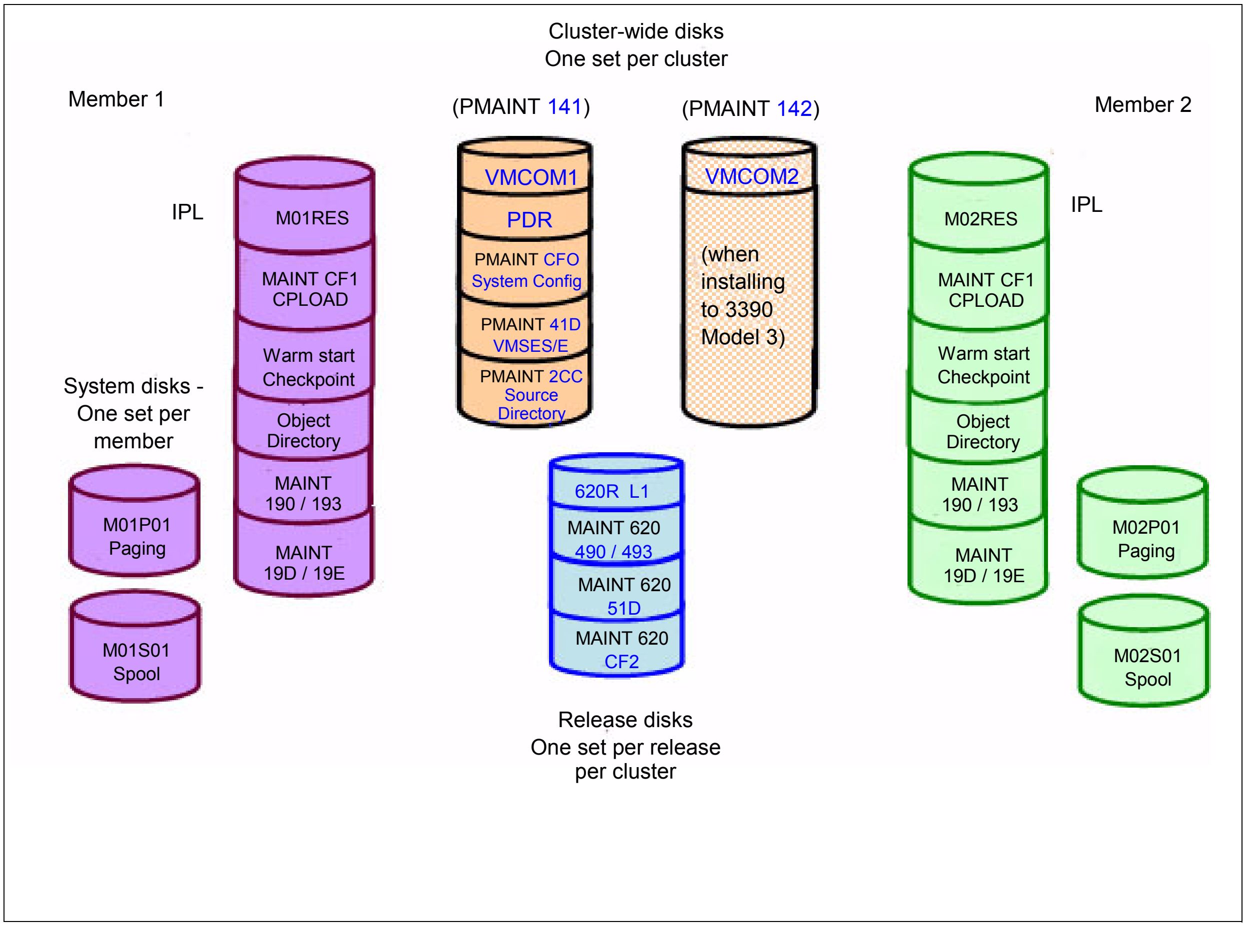


Figure 2-5 DASD volumes and minidisks in an SSI cluster

2.2.10 Persistent data record

The *persistent data record* (PDR) is information that must reside on a shared 3390, usually the common volume. It is used to provide a cross-system serialization point on disk. It is created and viewed with new FORMSSI utility.

Example 2-5 shows the output from the FORMSSI display command.

Example 2-5 FORMSSI display command

```
formssi display 141
HCPPDF6618I Persistent Data Record on device 0141 (label SSIBC1) is for ITSOSSIB
HCPPDF6619I PDR                  state: Unlocked
HCPPDF6619I                 time stamp: 11/14/11 10:04:37
HCPPDF6619I     cross-system timeouts: Enabled
HCPPDF6619I PDR slot  1         system: ITSOSSI5
HCPPDF6619I                      state: Joined
HCPPDF6619I                 time stamp: 11/14/11 10:04:37
HCPPDF6619I                last change: ITSOSSI5
HCPPDF6619I PDR slot  2         system: ITSOSSI6
HCPPDF6619I                      state: Joined
HCPPDF6619I                 time stamp: 11/14/11 10:04:24
HCPPDF6619I                last change: ITSOSSI6
```

It contains information about member status and the heartbeat data. It is also used for health-checking and ensures that a stalled or stopped member can be detected.

2.2.11 VMSSI considerations

The following important considerations relate to VMSSI:

- VMSSI is not compatible with Cross System Extensions (CSE).
- z/VM 6.2 has to run on a z10 Enterprise Class (EC) or z10 Business Class (BC) machine or higher.
- Physical CECs must be close enough to allow FICON CTC connections, shared DASD, and common network resources, because direct logical links must exist between all members in the SSI cluster.
- Installation on Small Computer System Interface (SCSI) devices is not supported, although guests can use SCSI devices.
- Automation for tasks, such as workload balancing, is not included with VMSSI. However, you can achieve automation through the implementation of additional products, such as IBM Operations Manager for z/VM.
- LGR is only officially supported for Linux on System z guests.

2.3 Planning considerations

The setup of multi-system virtualization with VMSSI requires both shared and non-shared DASD and I/O. It is important to plan the hardware configuration before starting the creation of an SSI cluster. Sample worksheets are included in Appendix C, “Additional material” on page 113.

2.3.1 Logical partition (LPAR) planning considerations

When planning the LPAR configuration, consider the following information:

- Up to 16 FICON CTC devices can exist between the LPARs, which are configured to provide direct Inter-System Facility for Communications (ISFC) links that connect every member of the SSI cluster to every other member. These channels can be switched or non-switched.
- FICON channels must be used to connect to the shared DASD. The shared DASD holds the common volumes, release volumes, persistent data record, user directory, and system configuration files.
- If Linux guests, which are being relocated using LGR, have OSAs, the OSAs must be on the same LAN segments.
- DASD that is accessed by Linux guests, which are to be relocated using LGR, must be shared across the SSI cluster.

2.3.2 DASD planning

It is important to perform detailed planning for the DASD that is shared and non-shared. Unlike previous versions of z/VM where only SYSRES, SPOOL, PAGE, and WORK volumes are considered CP system volumes, z/VM 6.2 uses additional CP system volumes that are shared. Common and release volumes are shared across the cluster. Although spool volumes are allocated to each member of the cluster, the spool volumes are shared by all members in the SSI cluster. See Figure 2-3 on page 13.

2.3.3 ISFC

All members of the SSI cluster are part of the same ISFC collection. Every member of the SSI cluster must have direct ISFC connections to every other member of the SSI cluster.

The number of required CTC adapters depends on the amount of data that needs to be accessed on a continuous basis, the size of the Linux guests being relocated, and the time available to relocate a Linux guest. You can improve the relocation time and quiesce time by adding additional channel paths. The preferred practice is to have multiple CTCs distributed on multiple FICON channel paths between each pair of members. This practice avoids write collisions and provides connectivity redundancy in the event of a failure. The preferred practice is to use the same real device number for the same CTC on each member.

It is possible to have multiple virtual switches (VSwitches), but the definitions need to be duplicated across all the members participating in the SSI cluster.

All members in the SSI cluster must have identical network connectivity and connect to the same physical LAN segments and the same SAN fabric. The VSwitch name, type, and virtual LAN (VLAN) settings must also be identical across all members of the SSI cluster if Linux guests are to be relocated successfully. The MAC address assignments must also be coordinated across the SSI cluster.

2.3.4 Non-SSI z/VM installation planning considerations

A stand-alone non-SSI z/VM Version 6.2 system also requires careful planning. You install this type of system in a manner that makes it easy to convert it into an SSI cluster at a later date. See Scenario two in Chapter 5, “Scenario two: Converting two existing stand-alone z/VM systems to a single system image cluster” on page 61.

Planning the DASD layout, user directory, and system configuration files is the same for a non-SSI installation as for an SSI installation, even if you do not plan to migrate the system to an SSI cluster.

2.4 z/VM SSI cluster operation

A z/VM system that is configured as a member of an SSI cluster joins the cluster when you IPL the system. A member leaves the cluster when it shuts down.

In the SSI cluster, the members can be in various states. The overall mode of the cluster depends on the combined states of the individual members. The state of each member and the cluster mode determine the degree of communication, resource sharing, and data sharing among the members.

New commands can be used to query the state of the cluster. We describe a subset of these commands in Appendix B, “New and updated commands” on page 105. Example 2-6 on page 21 shows the output from issuing the Q SSI command on our ITSOSSIB cluster.

Example 2-6 Q SSI Command output

```
q ssi
SSI Name: ITSOSSIB
SSI Mode: Stable
Cross-System Timeouts: Enabled
SSI Persistent Data Record (PDR) device: SSIBC1 on 9D20
SLOT SYSTEMID STATE     PDR HEARTBEAT       RECEIVED HEARTBEAT
   1 ITSOSSI5 Joined    11/11/11   11:10:37 11/11/11   11:10:37
   2 ITSOSSI6 Joined    11/11/11   11:10:39 11/11/11   11:10:39
   3 -------- Available
   4 -------- Available
```

2.4.1 SSI member states

The SSI modes and member states are described fully in *z/VM V6R2.0 CP Planning and Administration*, SC24-6178-02.

The information about the state of each member of the SSI cluster is held in the persistent data record (PDR), which is located on the shared common disk. This record contains a heartbeat from all the members of the client so that stalled or stopped members can be identified quickly.

The following states are normal member states:

- Down: The member is not initialized as a member of the SSI cluster. Or, the member left the cluster due to being shut down or as result of an error or has not attempted to join the cluster during system initialization.
- Joined: The member has successfully joined the cluster and is participating in cluster operations.
- Joining: The member is in the process of joining a cluster that already has one or more joined members.
- Leaving: The member is shutting down.

The following states are error member states:

- Suspended: The member cannot communicate with another member that is in a state other than Down or Isolated.
- Isolated: The member cannot join the cluster due to a failure either in the enablement of the cluster operations or when it attempts to join the cluster.
- Unknown: This is not a real state, but is displayed when another member's state cannot be determined because it has stopped communicating.

Example 2-7 shows the state of two members in our ITSOSSIB cluster. ITSOSSI6 is an active member of the cluster and ITSOSSI5 is shut down.

Example 2-7 State of the members of the SSI cluster ITSOSSIB

```
q ssi
SSI Name: ITSOSSIB
SSI Mode: Stable
Cross-System Timeouts: Enabled
SSI Persistent Data Record (PDR) device: SSIBC1 on 9D20
SSI Persistent Data Record (PDR) device: SSIBC1 on 9D20
SLOT SYSTEMID STATE     PDR HEARTBEAT       RECEIVED HEARTBEAT
```

```
 1 ITSOSSI5 Down (shut down successfully)
 2 ITSOSSI6 Joined    11/11/11   11:45:09 11/11/11   11:45:09
```

For more information about member states, see *z/VM V6R2.0 CP Planning and Administration*, SC24-6178-02.

2.4.2 SSI cluster modes

The cluster has a number of modes depending on the state of the individual members of the cluster.

The following modes are the SSI cluster modes:

- Stable: SSI cluster is fully operational.
- Influx: Members are in the process of joining or leaving the cluster. Cross-system functions are temporarily suspended in this mode, and negotiations for shared resources are deferred. Any existing accesses are unaffected.
- Safe: This mode occurs when a remote member's state cannot be determined, or any member is in a suspended state. It results in a failure to access any shared resources. Any existing accesses are unaffected.

Example 2-6 on page 21 shows our ITSOSSIB cluster in stable mode.

For more information about cluster modes, see *z/VM V6R2.0 CP Planning and Administration*, SC24-6178-02.

3

Live guest relocation (LGR) overview

Several reasons exist why it might be necessary to move a running Linux guest virtual machine from one single system image (SSI) member to another. To prepare for live guest relocation (LGR), ensure that the Linux guest has a relocatable configuration and that a matching configuration can be set up on the destination member. Certain configuration attributes can prevent the Linux guest from being relocated.

In this chapter, we provide a brief introduction to live guest relocation (LGR). We also describe several major attributes of LGR and factors that affect relocation. We identify the supported configurations for relocation. Also, we describe considerations about memory and paging that support LGR.

Additionally, we define relocation domains and business reasons for creating relocation domains. We share the characteristics of relocation domains. We also explain the steps for defining relocation domains and for placing virtual servers into relocation domains.

Finally, we outline the required steps to relocate and we conclude with conditions that potentially prevent a successful relocation.

3.1 Introducing live guest relocation (LGR)

LGR provides virtual server mobility, which means a nondisruptive move of virtual Linux guests from one member of a cluster to another member. Possible reasons for the move are load balancing or hardware or software maintenance of the z/VM member.

The major principle during LGR is that the relocating guests continue to run on the source member until the destination is ready to start running the guests. If any reasons exist why the target member cannot run the Linux guests, the guests stay on the source member.

As a result of the way that LGR is implemented, the time that the Linux guest is inoperable during the relocation is brief. The user or the application does not notice this short interruption.

Live guest relocation can be used for following reasons:

- To load balance.
- To perform maintenance of a z/VM member by moving the Linux guests nondisruptively to another member in the cluster.

Live guest relocations are initiated by a manual VMRELOCATE command. *The command is not automatically issued for any guest*. Therefore, LGR is not a high availability solution, and it is not a disaster recovery solution.

Think of LGR as a continuous availability solution for the virtual Linux guest.

3.2 Major attributes of live guest relocation

Chapter 2, "Single system image (SSI) overview" on page 9 describes the characteristics of an SSI cluster as a prerequisite for LGR. When a Linux guest is relocated, its direct access storage devices (DASD) are not moved, its virtual memory only is moved.

The z/VM control program (CP) attempts to relocate all the memory of the Linux guest in a series of passes, on each pass only moving the memory that changed since the last pass. This process continues until an internal algorithm determines that the guest needs to be quiesced to move the last memory pages. CP quiesces the Linux guest and relocates the final guest state, I/O information, and changed pages.

3.2.1 Factors affecting relocation

A number of factors can affect relocation:

- Virtual memory size

  The more memory a Linux guest has, the longer the relocation and quiesce time.

- Virtual machine page change rate

  The rate at which a Linux guest changes its pages in memory directly affects the total relocation time and, possibly, quiesce time. A Linux guest changing pages rapidly has more pages to relocate in each memory pass and so the memory move stage lasts longer.

- Inter-System Facility for Communications (ISFC) setup

  Faster channel-to-channel (CTC) speeds and the number of defined CTCs increase throughput and result in shorter relocation and quiesce time.

- Relocation options

  Relocation options can influence the relocation and quiesce time for relocation. See section 3.6.1, “VMRELOCATE command” on page 31 for a short description of these options.
- Other non-relocation activity

  Other non-relocation activity on source and destination systems might make relocation or quiesce time longer.

3.3 Supported configurations for relocation

In this section, we list the supported configurations that we studied in the ITSO lab environment. For more details, see *z/VM V6R2.0 CP Planning and Administration*, SC24-6178-02.

> **Reminder:** Linux on System z is currently the only guest environment that is supported for relocation.

We studied the following supported configurations:

- A Linux guest running in an ESA or XA virtual machine in ESA/390 or IBM zArchitecture mode
- A Linux guest running with a disconnected virtual console
- A Linux guest that was defined as a single configuration virtual machine, that is, the virtual machine definition in the directory begins with a USER statement
- A Linux guest with virtual processor types of all CP or all Integrated Facility for Linux (IFL)
- A Linux guest that we IPLed from a device or IPLed from a named saved system (NSS) that meets the NSS criteria identified in the next item of this list
- A Linux guest with a discontiguous saved segment (DCSS) or NSS as long as the DCSS or NSS contains no shared write (SW), shared with CP (SC), or shared with no data (SN) page ranges, and the identical DCSS or NSS is available on the destination system
- A Linux guest with supported devices that are available on the destination member:
  - Dedicated devices, including Open Systems Adapter (OSA) and Fibre Channel Protocol (FCP) devices, provided that same device address is available on the destination system.
  - Private virtual disks in storage (created with the DEFINE VFB-512 command).
  - Virtual unit record devices with no open spool files other than open console files. If the guest has an open console file, the relocation process closes the file on the source system and opens a new file on the destination system.
  - Virtual network interface cards (NICs) coupled to active virtual switches.
- The following facilities are supported:
  - Cryptographic adapter when enabled by CRYPTO APVIRTUAL statement and AP type must be the same on the source and the destination
  - IUCV connections to *MSG and *MSGALL CP system services
  - Application monitor record (APPLDATA) collection if guest buffer is not in a shared DCSS
  - Single console image facility

– Collaborative memory management assist (CMMA)

3.4 Memory and paging considerations for live guest relocation

The LGR process must ensure that memory and paging requirements are checked on the destination member before a relocation of a Linux guest is executed. Therefore, the z/VM development team spent significant effort establishing checks regarding available memory and paging space on the destination member.

One memory size check must be finished in any case with a positive result before the relocation can be executed. The current memory size of the guest *must* fit in the available space on the destination member (see Figure 3-1).

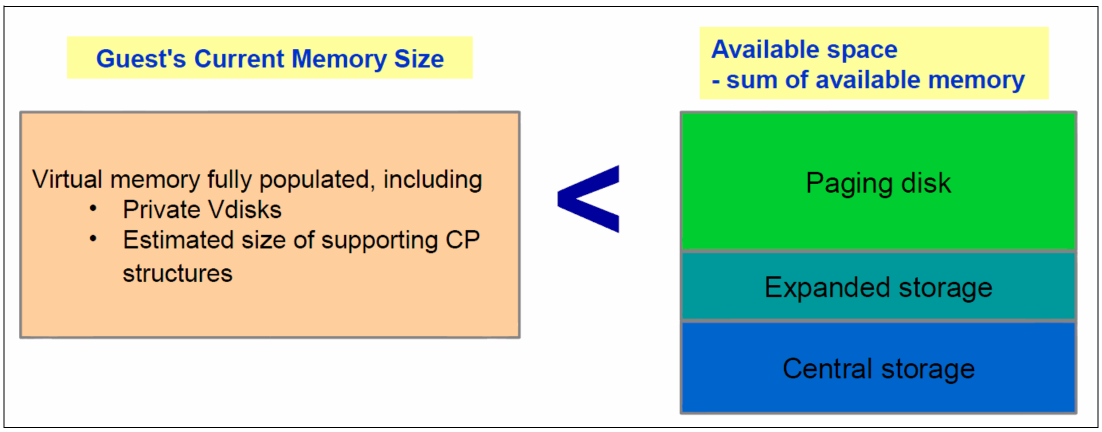


Figure 3-1 Guest storage must fit into the available space on the destination member

You must perform additional checks before the relocation starts. Nevertheless, the following checks can be overruled by using the FORCE STORAGE operand with the VMRELOCATE command. But, use the option “FORCE STORAGE” with caution. See 3.6.1, “VMRELOCATE command” on page 31.

The following checks are additional:

- The guest storage size must be less than the auxiliary paging capacity on the destination. See Figure 3-2.

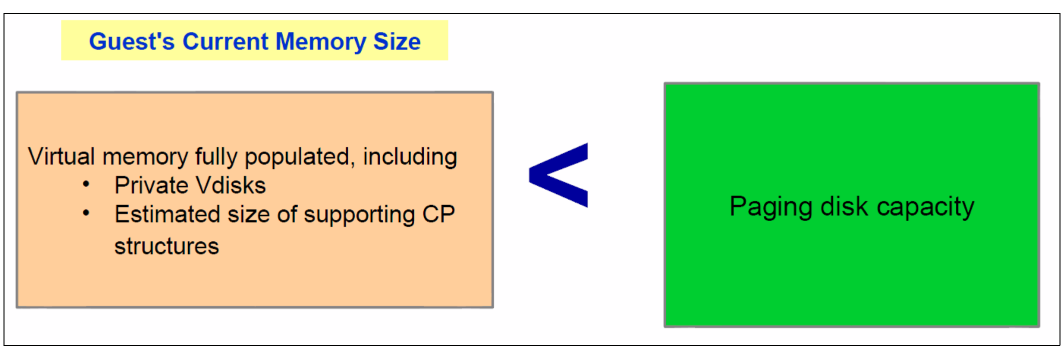


Figure 3-2 Guest storage size must be less than the auxiliary paging capacity on the destination

- z/VM 5.4 implemented a functionality that allowed you to configure reserved and standby storage before you IPL a guest operating system in the virtual machine. This functionality allows the guest to exploit dynamic storage reconfiguration for specific instructions. This functionality is considered in the following checks. Figure 3-3 shows that the guest maximum storage size must be less than all storage and paging capacity available on the destination. Figure 3-4 shows that the guest maximum storage size must be less than the paging capacity on the destination.

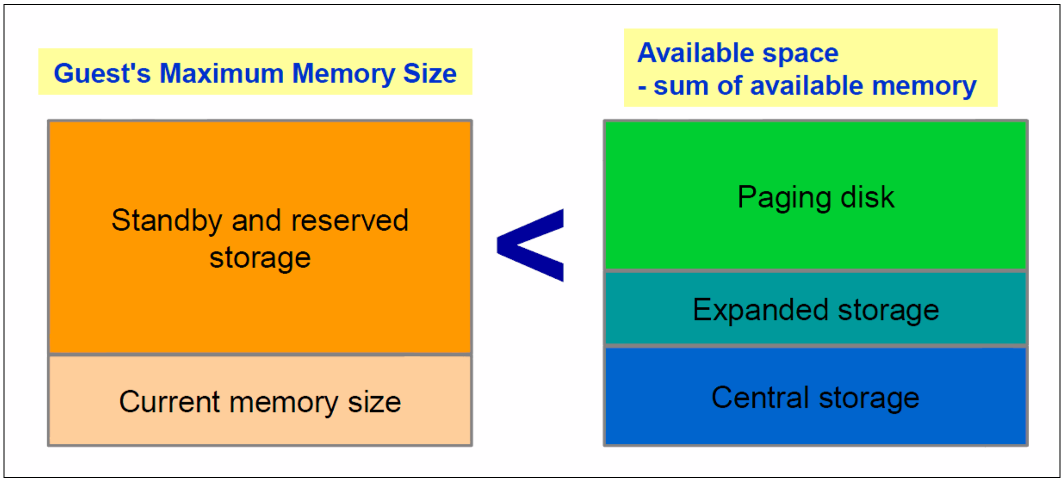


Figure 3-3 Guest maximum storage size is less than destination available storage and paging capacity

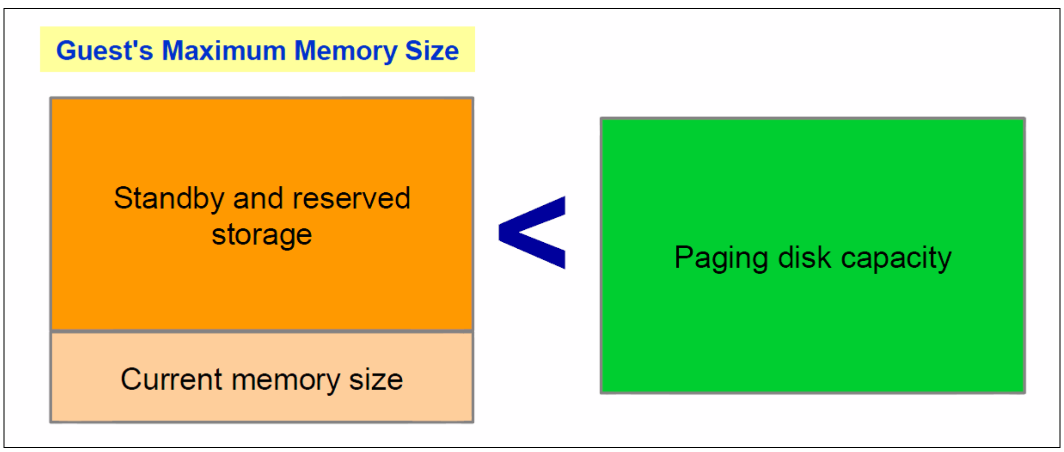


Figure 3-4 Guest maximum storage size must be less than the paging capacity on the destination

Paging requirements: For certain usage scenarios of LGR, such as moving guests to another z/VM member for maintenance reasons, LGR implies increased paging requirements. Therefore, available paging space must be at least two times the total virtual memory of all guests, including guests to be relocated to this member.

Avoid allocating more than 50% of available paging space. If the size of guests to be relocated increases the in-use amount of space to greater that 50%, system performance can be affected.

3.5 Relocation domains

A *relocation domain* defines a set of members of an SSI cluster among which Linux guests can be relocated. A relocation domain can be used to define a subset of members of an SSI cluster to which a particular Linux guest can be relocated. These domains can be defined for business or technical reasons.

Business reasons for relocation domains might be that your cluster contains members for a test environment and members for the production environment and you want to ensure that Linux guests are not moved from the test to the production environment accidentally. Nevertheless, you can enforce an out-of-domain relocation with the VMRELOCATE command and the option "FORCE DOMAIN". See 3.6.1, "VMRELOCATE command" on page 31. Also, a Linux guest can be licensed to run on a specific number of processors. A domain can be defined to ensure that the Linux guest does not run on more processors than defined in its license.

Technical reasons for relocation domains might be that only certain members of an SSI cluster support specific architectural features of a Linux guest. Therefore, a relocation of such a Linux guest is restricted to those members supporting those features. For example, only certain members have crypto features or only certain members support the Floating-point extensions facility. Nevertheless, in this case, you can enforce a relocation with the VMRELOCATE command and the "FORCE ARCHITECTURE" option, if needed. See 3.6.1, "VMRELOCATE command" on page 31.

3.5.1 Characteristics of relocation domains

Relocation domains have the following characteristics:

- Several default domains are automatically defined by CP as part of the configuration of the SSI cluster:
  - A common domain, which is named SSI and includes all members, that supports the facilities that are common to all members
  - Single member domains for each member in the SSI cluster, which are named the same as the member name
- Virtual machines are assigned to a default domain automatically:
  - When no relocation domain is defined in the user directory of a single configuration virtual machine (USER), the guest is assigned to the common SSI domain.
  - When no relocation domain is defined in the user directory of a multiple configuration virtual machine (IDENTITY), then it is assigned to the single member domain.
  - Regardless of differences in the features of the individual members, a domain has a common architectural level, containing the common features. This domain uses software fencing and hardware blocking facilities, which are in place to ensure that the Linux guest can use these features only during the logon of the Linux guest.
  - Linux guests can be relocated only between members that belong to the same relocation domain.

Figure 3-5 on page 29 is an example of a more complex scenario. It shows an SSI cluster that includes members VMSYS01, VMSYS02, VMSYS03, and VMSYS04. The members are on separate servers, which provide separate sets of architectural features. In addition to the implicitly defined cluster-wide relocation domain (SSI) and single-member relocation domains (VMSYS01, VMSYS02, VMSYS03, and VMSYS04), three relocation domains are explicitly defined. Domain X includes members VMSYS02 and VMSYS03. Domain Y includes members VMSYS02 and VMSYS04. And, domain Z includes members VMSYS03 and VMSYS04.

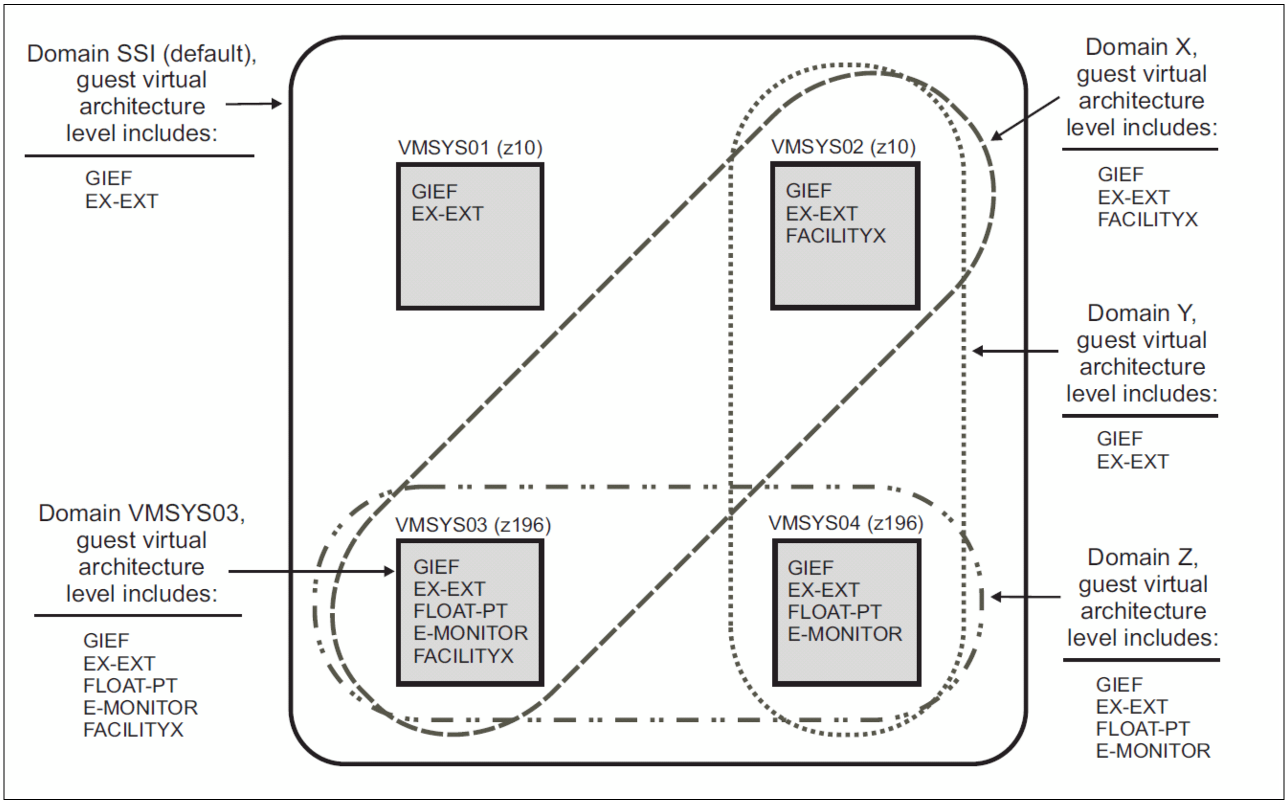


Figure 3-5 Relocation domains

The following architectural features are used in this example:

GIEF	General-instructions-extension facility
EX-EXT	Execute-extensions facility
FLOAT-PT	Floating-point extensions facility
E-MONITOR	Enhanced-monitor facility
FACILITY X	Possible future facility

Remember: The features that we illustrate are examples only. You do not need to understand these features. These features are used to explain the concepts of domains.

Figure 3-5 shows the following characteristics:

- The default domain for single configuration virtual machines (USER) is the cluster-wide relocation domain SSI, which contains all four members and has an architecture level containing the common architectural features.
- The default domain for multiconfiguration virtual machines (IDENTITY) is the single member domains, for example, VMSYS3 and contains the architectural level of that member. Dependent on the cluster member, where the guest is logged on, it gets those architectural features. Such multiconfiguration virtual machines cannot be relocated.
- Linux guests (defined as USERs), which belong to the default domain SSI, can use the common architectural features of the SSI domain (GIEF and EX-EXT). Linux guests can be relocated between all members of the cluster. Software fencing and hardware blocking facilities enforce that only these features are available for the Linux guests in that domain.

- As soon as this guest has to use more architectural features than provided in the default domain SSI, for example, the Linux guest has to use the additional architecture feature ‘FACILITY X’, the guest has to be moved to another relocation domain. In our example, the guest has to be moved to domain X. That means, now the guest only can be relocated between the systems VMSYS02 and VMSYS03 of domain X. The way to move guests to another relocation domain is shown in Example 3-4 on page 31.
- If a single configuration virtual machine is assigned to relocation domain X and logs on to member VMSYS04, which is not included in domain X, CP assigns a virtual architecture level. This virtual architectural level includes the features that are common to domain X and member VMSYS04, which are GIEF and EX-EXT.

For further details, see Chapter 27 of *z/VM V6R2.0 CP Planning and Administration*, SC24-6178-02.

3.5.2 Steps for defining relocation domains

Use the DEFINE RELODOMAIN command to define relocation domains dynamically. The command must be issued on only one member in the SSI cluster. Example 3-1shows the **define** command for setting up domainA on our cluster. The cluster members ITSOSSI1 and ITSOSSI2 are part of this domain.

Example 3-1 Define domain command

```
define relodomain domainA member itsossi1 itsossi2
Ready; T=0.01/0.01 17:27:45
```

For a permanent definition, place the RELOCATION_DOMAIN statement in your SYSTEM CONFIG file. Example 3-2 shows the RELOCATION_DOMAIN statement for domainA in our cluster.

Example 3-2 Relocation_domain statement

```
RELOCATION_DOMAIN domainA members itsossi1 itsossi2
```

Your existing relocation domains can be checked with the QUERY RELODOMAIN command. Example 3-3 shows the output from the **query relodomain** command on our cluster.

Example 3-3 Query Relodomain command

```
q relodomain all
DOMAIN DOMAINA MEMBERS: ITSOSSI1 ITSOSSI2
DOMAIN ITSOSSI1 MEMBERS: ITSOSSI1
DOMAIN ITSOSSI2 MEMBERS: ITSOSSI2
DOMAIN ITSOSSI3 MEMBERS: ITSOSSI3
DOMAIN ITSOSSI4 MEMBERS: ITSOSSI4
DOMAIN SSI      MEMBERS: ITSOSSI1 ITSOSSI2 ITSOSSI3 ITSOSSI4
Ready; T=0.01/0.01 17:28:35
```

3.5.3 Steps for placing virtual servers into relocation domains

Use the SET VMRELOCATE command to set the relocation domain for a virtual server dynamically. See Example 3-4.

Example 3-4 Set relocation domain for a Linux guest

```
set vmrelocate itsolnx1 on domain domainA
Running on member ITSOSSI1
Relocation enabled in Domain DOMAINA
Ready; T=0.01/0.01 17:29:57
```

It is best to assign the virtual server to its appropriate relocation domain permanently in its directory entry so that its virtual architecture level is determined at logon time. Example 3-5 shows the related **dirmaint** command. If the logon is allowed to default its domain, because no relationship is indicated in the directory entry, and then changed dynamically at a later time, the virtual architecture level of the server can be changed during server operations. The server might not handle such a dynamic change gracefully.

Example 3-5 Permanent setting of a relocation domain for a Linux guest

```
dirmaint for itsolnx1 vmrelocate on domain domainA
```

3.6 Performing a relocation

This section describes the steps to perform the relocation.

3.6.1 VMRELOCATE command

VMRELOCATE is the primary command for relocation. It has many operands. Therefore, we only describe the parts of the command that are essential to our scenarios. For a complete description of the command, see *z/VM V6R2.0 CP Commands and Utilities Reference*, SC24-6175-02. The VMRELOCATE command must be issued on the member where the Linux guest is running. We used the following operands with VMRELOCATE:

- VMRELOCATE TEST <*guestname*> <*destination cluster member*> checks the eligibility of the specified Linux guest for LGR.
- VMRELOCATE MOVE <*guestname*> <*destination cluster member*> moves the Linux guest non-disruptively to the destination cluster member.
- VMRELOCATE STATUS <*guestname*> displays information about relocations that are currently in progress. The command can be used for troubleshooting reasons.

The following additional options are available for VMRELOCATE MOVE:

- SYNCHRONOUS/ASYNCHRONOUS option:
  - SYNCHRONOUS, the default, causes the VMRELOCATE MOVE command to complete only after the relocation is completed. This restriction is designed to help you serialize relocations. Serializing relocations is the preferred practice.
  - ASYNCHRONOUS causes the VMRELOCATE MOVE command to return as soon as the initial eligibility check is done. You might use the ASYNC option on the command so that you can issue other commands during the relocation, such as a VMRELOCATE STATUS to see how the relocation is progressing.

- Execute relocations one at a time (SYNCHRONOUS) so that they do not compete with each other for system resources. This competition might increase relocation times and quiesce times.

- IMMEDIATE option

  The IMMEDIATE option causes the quiesce to occur after only one pass through memory. This option usually results in shorter overall relocation times, but longer quiesce times. It might be necessary to move more pages of memory during the quiesce phase depending on the activity on the Linux guest.

- MAXTOTAL/MAXQUIESCE option:
  - The MAXTOTAL time is the maximum total time (in seconds) that the command issuer is willing to wait for the entire relocation to complete. If a relocation needs more time, the relocation is canceled. The default MAXTOTAL time is NOLIMIT.
  - The MAXQUIESCE time is the amount of time (in seconds) that a Linux guest can be stopped during a relocation attempt. If a relocation needs more quiesce time, the relocation is canceled. The default MAXQUIESCE time is 10 seconds. If the option IMMEDIATE is specified, the default MAXQUIESCE time is NOLIMIT.
- FORCE options: *Use these options with caution*:
  - FORCE ARCHITECTURE: This option forces the relocation even if the destination cluster member does not support all architectural features of the source cluster member. The option can be used to allow out-of-domain relocation to a cluster member that does not support all the architectural features of the source cluster member.
  - FORCE DOMAIN: This option forces the relocation even if the destination cluster member is an out-of-domain member. This option does not permit a relocation outside of the domain if the move causes any loss of architectural features. In this case, you have to use the FORCE ARCHITECTURE DOMAIN option.
  - FORCE STORAGE: In 3.4, “Memory and paging considerations for live guest relocation” on page 26, we described the checks that were run during the eligibility checks of the relocation. The checks, which are listed as “additional” can be overruled with the FORCE STORAGE option. See Figure 3-2 on page 26, Figure 3-3 on page 27, and Figure 3-4 on page 27.

3.6.2 Performing the relocation

In this section, we list the necessary steps to perform a relocation. Consider the following assumptions:

- The relocation domains are defined, and the relocatable virtual servers are assigned to their proper domains.
- Equivalency identifiers (EQIDs) are assigned to all necessary devices that are connected to the devices that are connected to the relocatable virtual servers, for example, OSAs, FCPs, and Hipersockets. See 2.2.7, “Equivalency identifiers” on page 15.
- If the virtual server connects to a VSWITCH, the same VSWITCH must be defined on the source and target member.
- The devices are equivalent:
  - OSAs connect to the same LAN segment.
  - FCPs have access to the same SAN fabric.
- In the directory entry of the virtual server, specify OPTION CHPIDVIRTUALIZATION ONE.
- If your virtual server uses FCP devices that you specified within the Linux guest, that multipathing support is enabled through the “queue_if_no_path” option.

Relocation steps

Perform the following steps for the relocation:

1. Choose one class A user ID from which to issue all your relocation commands. Issuing the VMRELOCATE command from one user ID helps keep your relocations organized and helps you avoid issuing multiple relocations at one time.
2. For each virtual server that you want to relocate, determine whether there is a maximum amount of time that the server can be quiesced. Look at the applications running on the virtual server to determine whether any timeout values exist for those applications. For instance, a web server might need to reply to a request to serve a page within 5 seconds.
3. Issue the VMRELOCATE TEST command. The command tells you any reasons why the Linux guest is ineligible to relocate. If there are any issues, address them and then retry the VMRELOCATE TEST command. Example 3-6 and Example 3-7 show two examples of the VMRELOCATE TEST command.

   Example 3-6 VMRELOCATE TEST command with a successful check for eligibility

   ```
   vmrelocate test itsolnx1 itsossi2
   User ITSOLNX1 is eligible for relocation to ITSOSSI2
   Ready; T=0.01/0.01 17:31:33
   ```

   Example 3-7 VMRELOCATE TEST command for a virtual server that is ineligible for relocation

   ```
   vmrelocate test itsolnx1 itsossi2
   HCPRLH1940E ITSOLNX1 is not relocatable for the following reason(s):
   HCPRLE1950I ITSOLNX1: Virtual machine is not running disconnected
   HCPRLI1954I ITSOLNX1: Virtual machine device 0009 is a logical device
   HCPRLL1813I ITSOLNX1: Maximum pageable storage use (20640M) exceeds available
   auxiliary paging space on destination (14423036K) by 6712324K
   Ready(01940); T=0.01/0.01 17:40:48
   ```

4. Issue the VMRELOCATE MOVE command with the MAXTOTAL and MAXQUIESCE times that you determined are required. The default value for the MAXTOTAL time is NOLIMIT. The default value for the MAXQUIESCE time is 10 seconds. By default, the command is issued synchronously, meaning you cannot issue any other commands while the VMRELOCATE MOVE is ongoing. Example 3-8 on page 33 shows an example of the VMRELOCATE MOVE command.

   Example 3-8 VMRELOCATE MOVE command

   ```
   vmrelocate move itsolnx1 itsossi2 maxtotal 10 maxquiesce 5
   Relocation of ITSOLNX1 from ITSOSSI1 to ITSOSSI2 started
   User ITSOLNX1 has been relocated from ITSOSSI1 to ITSOSSI2
   Ready; T=0.01/0.01 17:32:43
   ```

3.7 Conditions that prevent a relocation

Several conditions can prevent a relocation from completing. The section entitled “Conditions that will prevent a relocation” in Chapter 27 of *z/VM V6R2.0 CP Planning and Administration*, SC24-6178-02, describes a detailed list of conditions.

In this book, we describe several typical conditions that prevent relocation:

- Guest state conditions:
  - The user ID is logged off or is still logging on.
  - The user ID is not disconnected.
  - The guest is a multiconfiguration virtual machine; the directory definition of the user begins with an IDENTITY statement.
- Device conditions:
  - The guest configuration includes an unsupported device.
  - The guest has a link to a local minidisk that cannot be linked on the destination site.
  - The guest configuration includes a real FCP, OSA, Hipersocket, or CTC device for which the destination system has no available device with the same EQID.
- Device state conditions:
  - The guest has a device for which the matching device on the destination system is offline or the device is not available.
  - The guest has a non-full pack minidisk on a device for which the matching real device on the destination system is not attached to the system.
- Virtual facility conditions:
  - The guest is a coupling facility machine or the guest configuration includes access to a coupling facility service machine. That is, the CFUSER option is specified in the virtual machine definition of the user. Or, the guest has a device that is coupled to a coupling facility service machine.
  - The guest has an inter-user communication vehicle (IUCV) connection to a system service (other than to *MSG or *MSGALL, which are allowed) or to another virtual machine.
- Configuration conditions:
  - The guest has a dedicated CPU.
  - The guest has dedicated cryptographic capability.
  - The guest configuration includes more than one virtual processor type, which precludes the relocation of any guest with a virtual zIIP or zAAP.
  - The guest is defined with all-type CP virtual processors and the destination is an all-type IFL environment.
- Resource limit conditions:
  - The guest storage size does not fit into the available space on the destination.
  - The guest’s virtual disk in storage usage violates the usage restrictions on the destination system.

4

Scenario one: Creating a single system image cluster with four new z/VM members

In this chapter, we discuss the planning and creation of a single system image (SSI) cluster with four members, all of which are created from the installation media. This scenario includes enabling and configuring the following products:

- TCP/IP (mandatory installation selection)
- Programmable Operator (optional installation selection)

We also describe the following priced features:

- Dirmaint (encouraged)
- Remote Spooling Communications Subsystem (RSCS) (optional)
- Performance Toolkit (encouraged)

4.1 Overview

The scenario described in this chapter is the implementation of a first level four-member SSI cluster. We configured other software products to run in the environment, as well. We set up a Linux guest under one of the members and performed a live guest relocation across the members without any disruption of operations to the Linux guest.

To recreate a practical scenario of setting up an SSI cluster and identifying the planning considerations for setting it up, we created a cluster with multiple members. Although an SSI cluster can be set across 1, 2, 3, or 4 members, we describe setting up a four-member SSI cluster and then demonstrate live guest relocation across all four members.

4.2 Setting up the SSI cluster

This section describes the planning considerations made for setting up the four-member SSI cluster. We called our SSI cluster ITSOSSIA with members ITSOSSI1, ITSOSSI2, ITSOSS3, and ITSOSSI4.

4.2.1 Planning worksheet

We decided to install all products onto minidisks rather than install onto a shared file system. Figure 4-1 shows our installation worksheet.

Installation method (first-level or second-level) First

Record an"M" if you will load the product to a minidisk or an "F" if you will load the product to the VMSYS file pool in the Install To column

Install To	Product	Install To	Product	Install To	Product
M	VM	M	OSA	M	PERFTK
M	VMHCD	M	RACF	M	DIRM
M	RSCS	M	ICKDSF	M	TCPIP

Default system lanuguage: AMENG

DASD type and model: 3390-9

SCSI Volume size:

Common service filepool name: SSISFSPL

Installation Type:

Non-SSI System Name:

X SSI Number of Members 4 SSI Cluster Name: ITSOSSIA

Figure 4-1 Installation planning worksheet

Because the setup is for a four-member SSI cluster, we plan to have first-level z/VM logical partitions (LPARs) running. Two LPARs will run on a z10EC machine, and two LPARs will run on a z196 machine. We plan to install all products and then customize the products that are required for this installation. This method gives us an opportunity to identify any additional installation considerations that are specific to setting up an SSI cluster.

4.2.2 Hardware definition

We used the following hardware to create this scenario:

- A z10 Enterprise Class machine and a z196 machine. On each of these machines, we created two LPARs with the following configuration:
  - Mode: ESA/390
  - Control programs (CPs): Four initially and two reserved
  - Central storage: 8 GB
  - Expanded storage: 2 GB
- In the setup that we created, ITSOSSI1 and ITSOSSI2 run on the z10 EC machine. and ITSOSSI3 and ITSOSSI4 run on the z196 machine. Figure 4-2 shows the worksheet that we completed for our SSI cluster members.

SSI Member Name(s) / IPL LPAR Name(s) or UserID Name(s):		
Slot Number	Member Name	IPL LPAR/User ID
1	ITSOSSI1	A1A
2	ITSOSSI2	A1B
3	ITSOSSI3	A2A
4	ITSOSSI4	A2B

Figure 4-2 LPAR allocation and member names across z196 and z10

- Operator console addresses that are defined to Console 1 and Console 2:
  - ITSOSSI1 and ITSOSS12: F200 and F280
  - ITSOSSI3 and ITSOSS14: F200 and F280
- DASD that is assigned for this setup:

  ITSOSSI1,ITSOSS12, ITSOSSI3, and ITSOSS14: 9E20 - 9E2F

4.2.3 Defining the software

After the installation of the products, which are listed in Figure 4-1 on page 36, we customized the following products for our use:

- TCP/IP
- Remote Spooling Communications Subsystem (RSCS)
- Directory Manager
- Performance Toolkit
- Programmable Operator

4.2.4 Configuring the DASD

Plan carefully for an SSI installation, because it involves multiple machines having shared and non-shared volumes. Unlike previous versions of z/VM, you must specify common and release volumes, as well as the system residence, paging, and spool volumes. For this setup, we used all 3390 Model 9 volumes for DASD. We show the planned layout schema for the shared and non-shared volumes in Figure 4-3 on page 38. Because they are Model 9 IBM extended count key data (ECKD) volumes, we can allocate fewer disks for common, release, and work volumes. Consider the following information when setting up DASD for an SSI cluster:

- You need one set of cluster-wide shared common volumes. This set contains shared data files for the SSI cluster, such as the shared system configuration file, shared source directory file, and cluster-wide minidisks. These files are owned by the PMAINT user ID.
- You need one set of shared release volumes for each z/VM release in the cluster. The minidisks on this volume are owned by the release-specific user ID MAINT620.
- You need one set of non-shared member-specific system volumes. This set contains member-specific data, such as standard system minidisks that are owned by MAINT, volumes for paging, and temporary disks.
- You need spool volumes for each member. These spool volumes are shared by the SSI cluster.

Figure 4-3 shows our DASD configuration.

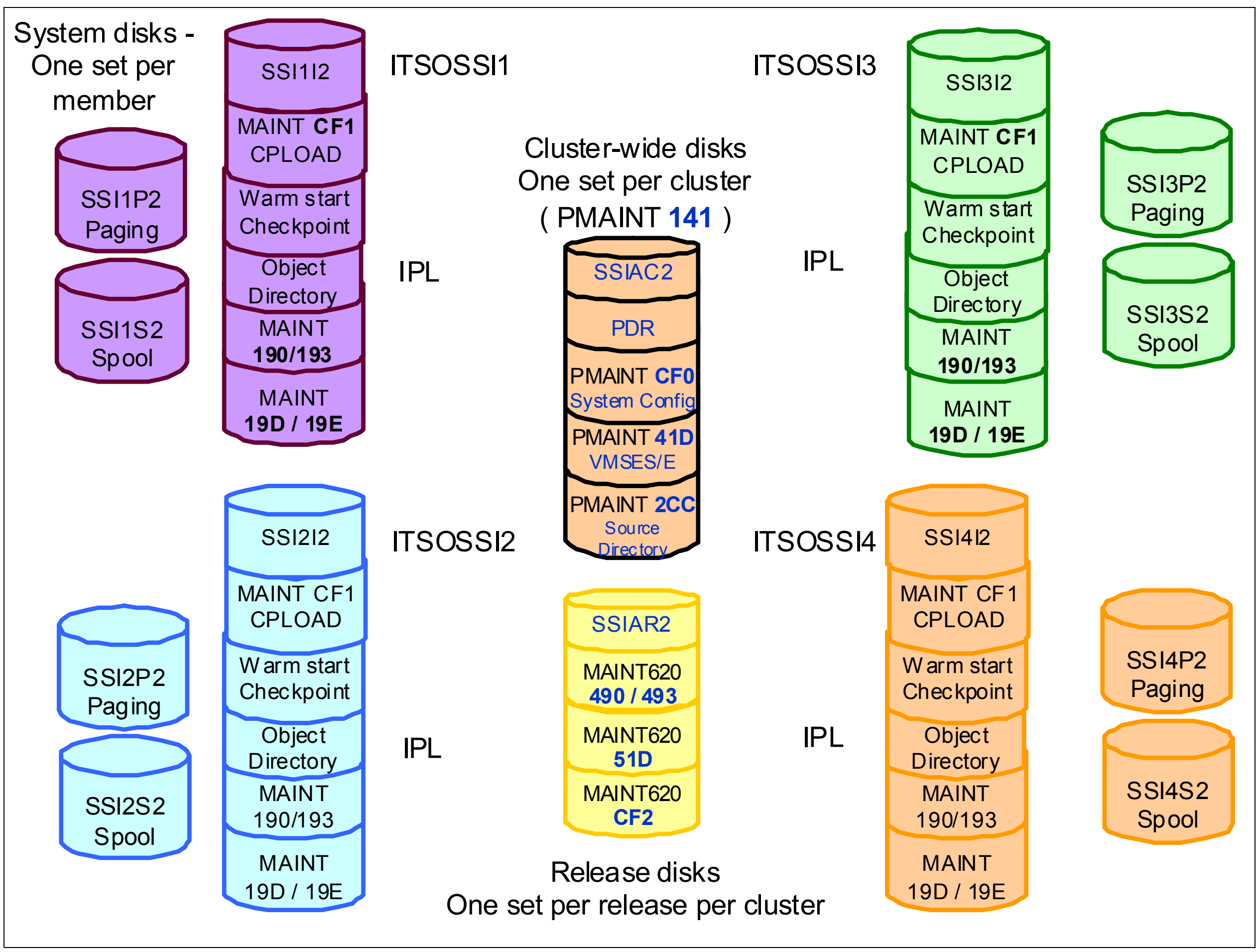


Figure 4-3 Organization of DASD volumes and minidisks in the SSI cluster ITSOSSIA

To create this setup, we carefully planned the DASD volumes that we used, identifying which volumes were shared and non-shared. For this scenario, we described the available DASD volumes in detail in the planning worksheet for the SSI installation that is shown in Figure 4-4 on page 39.

		Common Volumes					
		Volume Type	Default Label	New Label	Address		
		COMMON	VMCOM1	SSIAC2	9E20		
		COMMON2	VMCOM2		Not required for model 9		
		RELVOL	620RL1	SSIAR2	9E29		
		RELVOL2	620RL2		Not required for model 9		
Dedicated Volumes							
Volume Type	Default Label	New Label	Address	Volume Type	Default Label	New Label	Address
Member 1:				Member 2:			
RES	M01RES	SSI1I2	9E21	RES	M02RES	SSI2I2	9E22
SPOOL	M01S01	SSI1S2	9E25	SPOOL	M02S01	SSI2S2	9E26
PAGE	M01P01	SSI1P2	9E2A	PAGE	M02P01	SSI2P2	9E2B
WORK	M01W01		Not required for model 9	WORK	M02W01		Not required for model 9
Member 3:				Member 4:			
RES	M03RES	SSI3I2	9E23	RES	M04RES	SSI4I2	9E24
SPOOL	M03S01	SSI3S2	9E27	SPOOL	M04S01	SSI4S2	9E28
PAGE	M03P01	SSI3P2	9E2C	PAGE	M04P01	SSI4P2	9E2D
WORK	M03W01		Not required for model 9	WORK	M04W01		Not required for model 9
Note: You must *not* use any of IBM's default volume labels for a volume other than the volume for which it is orginially defined							

Figure 4-4 DASD setup planned for the installation

Figure 4-5 shows a graphical representation of the SSI DASD layout.

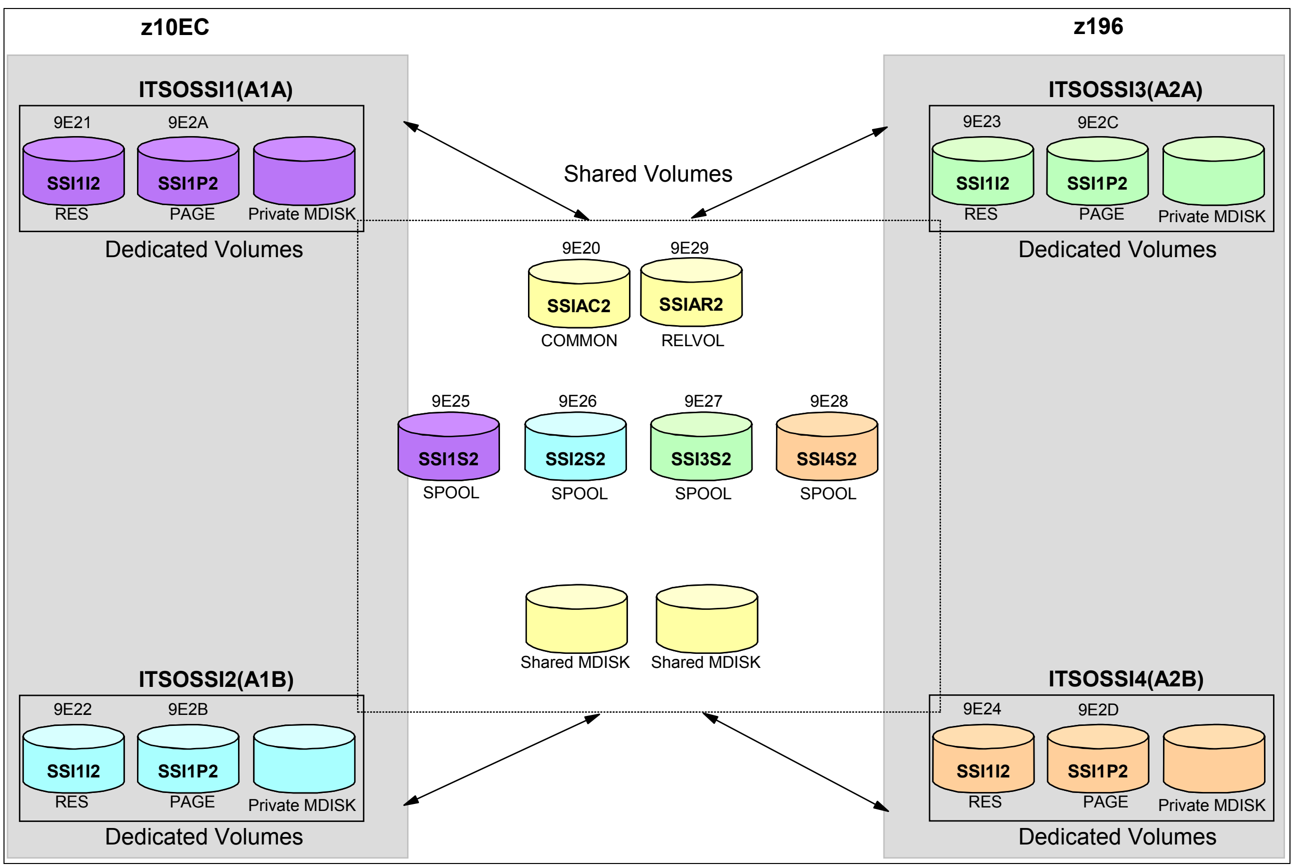


Figure 4-5 DASD layout: Shared and non-shared volumes

4.2.5 Configuring the network

We set up the following connections between each of the four LPARs:

- Two channel-to-channel (CTC) connections between each LPAR to act as inter-system facility for communications (ISFC) connections
- One CTC for RSCS connections among the four LPARs
- One CTC from each LPAR to an external ITSO RSCS

Each LPAR also uses a set of open systems adapter (OSA) cards connected to the same network. Therefore, they can all use the same equivalency identifier (EQID). We shared one OSA card between TCP/IP and VSWITCH on each of the SSI members.

We use the following available OSA device addresses for IP connectivity:

- ITSOSSI1 and ITSOSSI2: 3080-308F and 30A0-30AF
- ITSOSSI3 and ITSOSSI4: 2100-210F and 2120-212F

Configuring ISFC

Table 4-1 show the SSI cluster connections that are available for ISFC CTC.

Table 4-1 SSI cluster connections for ISFC

ITSOSSI1	ITSOSSI2	ITSOSSI3	ITSOSSI4
4A60-4A63	5A50-5A53		
4A68-4A6B	5A58-5A5B		
5A60-5A63	4A50-4A53		
5A68-5A6B	4A58-4A5B		
43A0-43A3		5A50-5A53	
43A8-43AB		5A58-5A5B	
53A0-53A3		4A50-4A53	
53A8-53AB		4A58-4A5B	
43B0-43B3			5A50-5A53
43B8-43BB			5A58-5A5B
53B0-53B3			4A50-4A53
53B8-53BB			4A58-4A5B
	43A0-43A3	5A60-5A63	
	43A8-43AB	5A68-5A6B	
	53A0-53A3	4A60-4A63	
	53A8-53AB	4A68-4A6B	
	43B0-43B3		5A60-5A63
	43B8-43BB		5A68-5A6B
	53B0-53B3		4A60-4A63
	53B8-53BB		4A68-4A6B

ITSOSSI1	ITSOSSI2	ITSOSSI3	ITSOSSI4
		43B0-43B3	53A0-53A3
		43B8-43BB	53A8-53AB
		53B0-53B3	43A0-43A3
		53B8-53BB	43A8-43AB

The allocation of the ISFC connections across the members is one of the most crucial steps in installation planning. Figure 4-6 shows the allocation of the CTC addresses based on Table 4-1 on page 40.

Real addresses for the COMMON volume on each member LPAR			
Member 1 Address	Member 2 Address	Member 3 Address	Member 4 Address
9E20	9E20	9E20	9E20

CTC Device addresses:

From:	Member 1			From:	Member 2		
To:	Member 1	N/A		To:	Member 1	5A50	4A50
To:	Member 2	4A60	5A60	To:	Member 2	N/A	
To:	Member 3	53A8	43A8	To:	Member 3	53A2	43A2
To:	Member 4	43B2	53B2	To:	Member 4	43B8	53B8
From:	Member 3			From:	Member 4		
To:	Member 1	4A58	5A58	To:	Member 1	5A52	4A52
To:	Member 2	4A62	5A62	To:	Member 2	5A68	4A68
To:	Member 3	N/A		To:	Member 3	43AA	53AA
To:	Member 4	53BA	43BA	To:	Member 4	N/A	

Figure 4-6 CTC layout across four-member SSI cluster

Figure 4-7 on page 42 shows a graphical representation of the SSI CTC allocation for ISFC.

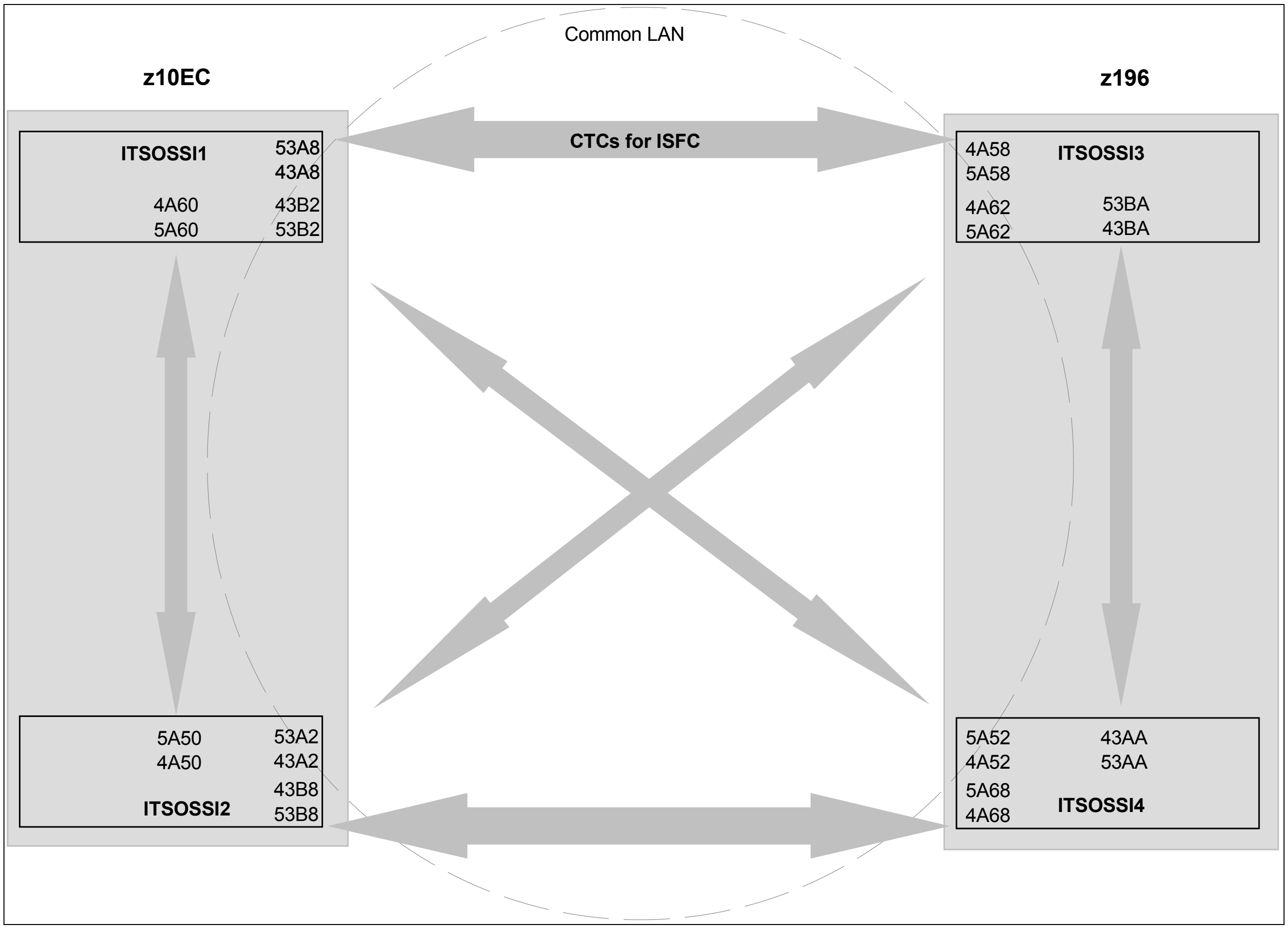


Figure 4-7 CTC layout for ISFC

Configuring Remote Spooling Communications Subsystem

Remote Spooling Communication Subsystem (RSCS) is a networking product. It enables users on one system to send messages, files, commands, and jobs to other users within a network. It provides data file transfer and print services to, from, and through the one z/VM system on which it runs, using both its own and TCP/IP networks. It extends the scope of a single system to an entire network of computers and devices. RSCS transfers data (as spool files) between its local system and remote devices and printers or other systems. It also acts as a print server for remote printers that are attached to other VM systems or a TCP/IP network. Through RSCS, users can send and receive messages, files, commands, and print and submit jobs within their network.

Table 4-2 on page 43 shows the SSI cluster connections that are available by RSCS in our lab environment.

Table 4-2 SSI cluster connections for RSCS

ITSOSSI1	ITSOSSI2	ITSOSSI3	ITSOSSI4	WTSCVMXA
4A90-4A93				4A50-4A53
4A98-4A9B				5A58-5A5B
5A90-5A93				4A50-4A53
5A98-5A9B				4A58-4A5B
	4A90-4A93			4A60-4A63
	4A98-4A9B			5A68-5A6B
	5A90-5A93			4A60-4A63
	5A98-5A9B			4A68-4A6B
		4A90-4A93		43A0-43A3
		4A98-4A9B		53A8-53AB
		5A90-5A93		43A0-43A3
		5A98-5A9B		43A8-43AB
			4A90-4A93	43B0-43B3
			4A98-4A9B	53B8-53BB
			5A90-5A93	43B0-43B3
			5A98-5A9B	43B8-43BB

Figure 4-8 on page 44 shows a graphical representation of the SSI CTC allocation for RSCS.

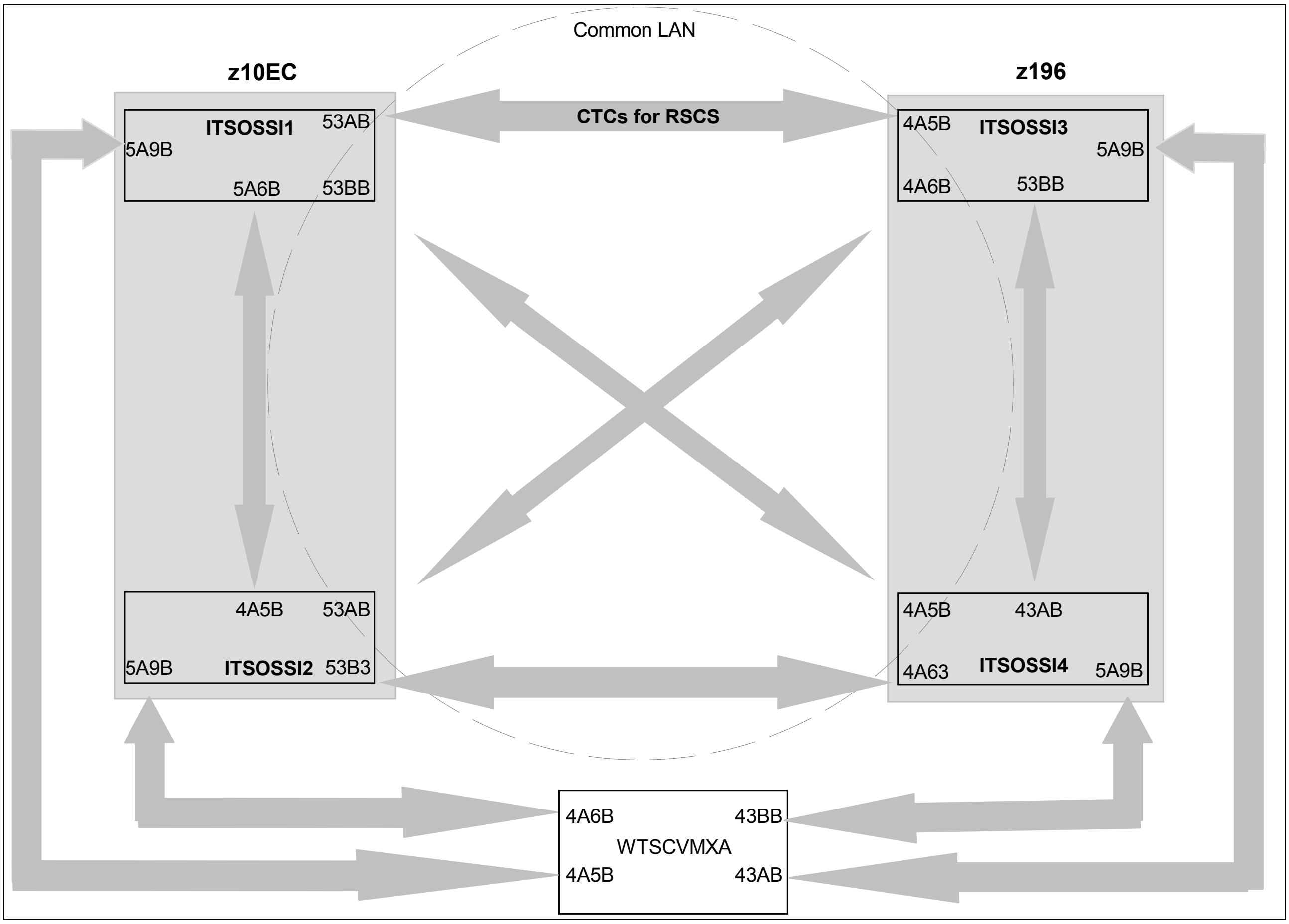


Figure 4-8 CTC layout for RSCS

Configuring TCP/IP

Table 4-3 on page 45 shows the TCP/IP setup that is based on the input data that is provided for this scenario.

Table 4-3 TCP/IP setup sheet for the four members

Host name	ITSOSSI1	ITSOSSI2	ITSOSSI3	ITSOSSI4
Domain name	itso.ibm.com	itso.ibm.com	itso.ibm.com	itso.ibm.com
Domain name server (DNS) IP	9.12.6.7	9.12.6.7	9.12.6.7	9.12.6.7
Gateway IP	9.12.4.1	9.12.4.1	9.12.4.1	9.12.4.1
Interface name	OSA1	OSA1	OSA1	OSA1
Device number	3080-3082	30A0-30A2	2100-2102	2120-2122
IP address	9.12.4.232	9.12.4.233	9.12.4.234	9.12.4.235
Subnet mask	255.255.252.0	255.255.252.0	255.255.252.0	255.255.252.0
Maximum transmission unit (MTU) discovery	Enabled	Enabled	Enabled	Enabled
Interface	QDIO	QDIO	QDIO	QDIO
Router type	None	None	None	None
Router advertisements	Off	Off	Off	Off
MTU size	1492	1492	1492	1492

4.2.6 Guests

You must plan for the Linux guests that participate in live guest relocation (LGR), as well. Linux guests must meet certain eligibility criteria and requirements for the destination member:

- The guest operating system must be supported for relocation.
- The source and target server environments for the relocatable members must be comparable and architecturally supported.
- If relocation domains are established and certain policies are set, the relocation must comply with these relocation domains and policies.
- The destination environment must have sufficient system resources to accommodate the relocating guest.
- System resources that are required by the relocating guest must be available on the target environment.

4.3 Installation process

For a detailed description of the installation process, see the *z/VM V6R2.0 Installation Guide,* GC24-6246.

4.3.1 Instplan command

We ran INSTPLAN EXEC and entered the following information:

- The LPAR and member names as detailed in the installation planning work sheet that is shown in Figure 4-1 on page 36. Figure 4-9 shows the output after entering this information for our cluster and four members.

```
HCPIPX8475I THE PRODUCTS YOU SELECTED TO LOAD TO MINIDISK ARE:
VM OSA PERFTK VMHCD RACF DIRM RSCS ICKDSF TCPIP

THE PRODUCTS YOU SELECTED TO LOAD TO SFS ARE:
NONE

THE SYSTEM DEFAULT LANGUAGE SELECTED:
AMENG

THE COMMON SERVICE FILEPOOL NAME IS:
SSISFSPL

THE INSTALL TYPE YOU SELECTED IS:
SSI

THE SSI CLUSTER NAME IS:
ITSOSSIA

THE NUMBER OF MEMBERS IS
4
   MEMBER NAME 1: ITSOSSI1IPL LPAR/USERID 1: A1A
   MEMBER NAME 2: ITSOSSI2IPL LPAR/USERID 2: A1B
   MEMBER NAME 3: ITSOSSI3IPL LPAR/USERID 3: A2A
   MEMBER NAME 4: ITSOSSI4IPL LPAR/USERID 4: A2B

AFTER INTSALLATION IS COMPLETE, MEMBERS WILL BE IPLed FROM:
First-Level

THE DASD TYPE YOU SELECTED TO LOAD ON IS:
3390 Mod 9

THE VOLUMES NEEDED TO LOAD z/VM ARE:

   COMMON:VMCOM1
   RELEASE:620RL1
   MEMBER1:M01RES M01301 M01P01
   MEMBER2:M02RES M02301 M02P01
   MEMBER3:M03RES M03301 M03P01
   MEMBER4:M04RES M04301 M04P01

DO YOU WANT TO CONINUE? (Y/N)
Y
```

Figure 4-9 Cluster and member output

- The DASD, as planned in Figure 4-4 on page 39. In this case, we defined 14 numbers for model 3390-9 DASD. Figure 4-6 on page 41 shows the address of the common volume for each member, which, in this case, is the same address for all members.
- Figure 4-6 on page 41 also shows the CTC connections for the entire cluster. At installation time, we can only define two connections between each member. We can add more connections after the installation is complete, if required.
- The last window from the INSTPLAN EXEC command shows that the DASD will be formatted and the connections will be defined based on the data input.

4.3.2 INSTALL command

Next, we executed the INSTALL command to perform the following actions:

- Format the DASD
- Copy the content of the DVD to its respective minidisks
- Install the z/VM system
- Start each member of the SSI cluster
- Run PUT2PROD on each member of the SSI cluster

The successful completion of the INSTALL command is indicated by the message that is shown in Figure 4-10.

```
VMFP2P1233I The following products have been put into production. Recycle the
            appropriate servers.
VMFP2P1233I GCS
VMFP2P2760I PUT2PROD processing completed successfully
**********************************************************
**********************************************************
HCPMLP8392I INSTALL EXEC ENDED SUCCESSFULLY
Ready; T=4.16/4.42 16:15:13
```

Figure 4-10 Completion of the INSTALL command

4.3.3 INSTSCID command

We ran the INSTSCID command to set up the SYSTEM CONFIG files to include the correct system identifier statement for each member.

4.3.4 Configuring the system

After the last command, we started all four z/VM images in their respective LPARs.

Figure 4-11 on page 48 shows that the cluster is ready (SSI Mode: Stable) and all members are running (State: Joined).

```
HCPPLM1698I The mode of the SSI cluster is STABLE
USER DSC   LOGOFF AS  AUTOLOG1 USERS = 9
Q SSI
SSI Name: ITSOSSIA
SSI Mode: Stable
Cross-System Timeouts: Enabled
SSI Persistent Data Record (PDR) device: SSIAC2 on 9E20
SLOT SYSTEMID STATE     PDR HEARTBEAT       RECEIVED HEARTBEAT
   1 ITSOSSI1 Joined    11/02/11   16:37:36 11/02/11   16:37:36
   2 ITSOSSI2 Joined    11/02/11   16:37:46 11/02/11   16:37:46
   3 ITSOSSI3 Joined    11/02/11   16:37:48 11/02/11   16:37:48
   4 ITSOSSI4 Joined    11/02/11   16:37:44 11/02/11   16:37:44
LOG
CONNECT= 00:03:12 VIRTCPU= 000:00.00 TOTCPU= 000:00.04
LOGOFF AT 16:40:19 EDT WEDNESDAY 11/02/11
```

Figure 4-11   SSI cluster stable with four members

With all members running, we logged on to MAINT620 on the Hardware Management Console (HMC) of each member. We ran the program IPWIZARD to set up TCP/IP, which enabled us to log on to the members of the cluster without having to use the HMC. We configure TCP/IP for normal use later in this section.

SYSTEM CONFIG file

We enabled several functions in our SSI cluster by amending the following sections of the SYSTEM CONFIG file. The SSI cluster requires these changes:

- User volume section

  Included volumes that are used by our Linux guests
- Relocation domain section

  Created two domains, DMNZ10 and DMNZ196, for illustration
- VSWITCH section, which is used to create our VSWITCH:
  - The MacPrefix differs for each member, but the members use the same UserPrefix.
  - The name of the VSWITCH is the same for all members.
  - All our OSAs are connected to same network, so all OSAs have the same EQID.
  - The GRANTs are common to all members.

We made these changes as a preferred practice:

- Features section

  We adjusted this section to enable Auto_Warm_IPL, Clear_TDisk, and STP_TZ.
- Signal section

  We adjusted the time that is needed for shutdown.
- Console section

  We defined one console for each member.
- Timezone section

  Although we kept the Timezone section in the SYSTEM CONFIG file, it is not necessary due to the use of STP_Timezone.

Now, we can define the additional CTC devices, which we did not include earlier as part of the ISFC setup.

Example 4-1 shows sections of the SYSTEM CONFIG file after all the changes are made. Appendix C, “Additional material” on page 113 includes the full SYSTEM CONFIG file. This file resides on minidisk CF0, which is owned by the PMAINT machine.

Example 4-1 SYSTEM CONFIG with four members

```
/**********************************************************************/
/*              SYSTEM CONFIG FILE                                    */
/**********************************************************************/
.
.
/**********************************************************************/
/*                    System_Identifier Information                   */
/**********************************************************************/

     System_Identifier LPAR A1A ITSOSSI1
     System_Identifier LPAR A1B ITSOSSI2
     System_Identifier LPAR A2A ITSOSSI3
     System_Identifier LPAR A2B ITSOSSI4

/**********************************************************************/
/*          SSI Statement required for VMSSI feature                  */
/**********************************************************************/

  SSI ITSOSSIA  PDR_VOLUME SSIAC2,
        SLOT 1 ITSOSSI1,
        SLOT 2 ITSOSSI2,
        SLOT 3 ITSOSSI3,
        SLOT 4 ITSOSSI4

/**********************************************************************/
/*                 Checkpoint and Warmstart Information               */
/**********************************************************************/

  ITSOSSI1:     System_Residence,
                Checkpoint  Volid SSI1I2   From CYL 21  For 9 ,
                Warmstart   Volid SSI1I2   From CYL 30  For 9
  ITSOSSI2:     System_Residence,
                Checkpoint  Volid SSI2I2   From CYL 21  For 9 ,
                Warmstart   Volid SSI2I2   From CYL 30  For 9
  ITSOSSI3:     System_Residence,
                Checkpoint  Volid SSI3I2   From CYL 21  For 9 ,
                Warmstart   Volid SSI3I2   From CYL 30  For 9
  ITSOSSI4:     System_Residence,
                Checkpoint  Volid SSI4I2   From CYL 21  For 9 ,
                Warmstart   Volid SSI4I2   From CYL 30  For 9

/**********************************************************************/
/*                    CP_Owned Volume Statements                      */
/**********************************************************************/
.
.
/**********************************************************************/
```

```
    ITSOSSI1:   CP_Owned   Slot   1  SSI1I2
    ITSOSSI2:   CP_Owned   Slot   1  SSI2I2
    ITSOSSI3:   CP_Owned   Slot   1  SSI3I2
    ITSOSSI4:   CP_Owned   Slot   1  SSI4I2

/**********************************************************************/
/*                                                 COMMON VOLUME      */
.
.
/**********************************************************************/

                CP_Owned   Slot   5  SSIAC2

/**********************************************************************/
/*                                           DUMP & SPOOL VOLUMES     */
.
.
/**********************************************************************/

                CP_Owned   Slot  10  SSI1S2
                CP_Owned   Slot  11  SSI2S2
                CP_Owned   Slot  12  SSI3S2
                CP_Owned   Slot  13  SSI4S2

/**********************************************************************/
/*                                           PAGE & TDISK VOLUMES     */
.
.
/**********************************************************************/
/* Page and Tdisk volumes for Member 1                                */
/**********************************************************************/

    ITSOSSI1:   BEGIN
                CP_Owned   Slot 255  SSI1P2
    ITSOSSI1:   END

/**********************************************************************/
/* Page and Tdisk volumes for Member 2                                */
/**********************************************************************/

    ITSOSSI2:   BEGIN
                CP_Owned   Slot 255  SSI2P2
    ITSOSSI2:   END

/**********************************************************************/
/* Page and Tdisk volumes for Member 3                                */
/**********************************************************************/

    ITSOSSI3:   BEGIN
                CP_Owned   Slot 255  SSI3P2
    ITSOSSI3:   END

/**********************************************************************/
/* Page and Tdisk volumes for Member 4                                */
```

```
/*********************************************************************/

    ITSOSSI4:   BEGIN
                CP_Owned   Slot 255  SSI4P2
    ITSOSSI4:   END


/*********************************************************************/
/*         Activate ISLINK statements                                */
/*********************************************************************/

  ITSOSSI1:    ACTIVATE ISLINK 4A60 5A60  NODE ITSOSSI2
  ITSOSSI1:    ACTIVATE ISLINK 53A8 43A8  NODE ITSOSSI3
  ITSOSSI1:    ACTIVATE ISLINK 43B2 53B2  NODE ITSOSSI4
  ITSOSSI2:    ACTIVATE ISLINK 5A50 4A50  NODE ITSOSSI1
  ITSOSSI2:    ACTIVATE ISLINK 53A2 43A2  NODE ITSOSSI3
  ITSOSSI2:    ACTIVATE ISLINK 43B8 53B8  NODE ITSOSSI4
  ITSOSSI3:    ACTIVATE ISLINK 4A58 5A58  NODE ITSOSSI1
  ITSOSSI3:    ACTIVATE ISLINK 4A62 5A62  NODE ITSOSSI2
  ITSOSSI3:    ACTIVATE ISLINK 53BA 43BA  NODE ITSOSSI4
  ITSOSSI4:    ACTIVATE ISLINK 5A52 4A52  NODE ITSOSSI1
  ITSOSSI4:    ACTIVATE ISLINK 5A68 4A68  NODE ITSOSSI2
  ITSOSSI4:    ACTIVATE ISLINK 43AA 53AA  NODE ITSOSSI3

/*********************************************************************/
.
.
/*********************************************************************/
/* Shared User Volumes                                               */
/*********************************************************************/
               User_Volume_List  SSIAR2
               User_Volume_List  LXDE1C LX603D LX9B25 LX9B26
               User_Volume_Include  SSI*

.
.
/*********************************************************************/

 Features ,
   Disable ,                        /* Disable the following features */
     Set_Privclass ,                /* Disallow SET PRIVCLASS command */
   Enable  ,                        /* Disable the following features */
     Auto_Warm_IPL ,                /* No Prompt at IPL always        */
     Clear_TDisk   ,                /*       Clear TDisks at IPL time */
     STP_TZ        ,                /* Test for STP Support           */
   Retrieve ,                       /* Retrieve options               */
     Default  60 ,                  /* Default.... default is 20      */
     Maximum  255 ,                 /* Maximum.... default is 255     */
   MaxUsers noLimit ,               /* No limit on number of users    */
   Passwords_on_Cmds ,              /* What commands allow passwords? */
     Autolog  no  ,                 /* ... AUTOLOG does               */
     Link     no  ,                 /* ... LINK does                  */
     Logon    no  ,                 /* ... and LOGON does, too        */
   Vdisk Userlim 144000 blocks,     /* Maximum vdisk allowed per user */
   Disconnect_timeout off
```

```
/*********************************************************************/
/* Relocation Domains
/*********************************************************************/

   Relocation_Domain DMNZ10   MEMBER ITSOSSI1 ITSOSSI2
   Relocation_Domain DMNZ196  MEMBER ITSOSSI3 ITSOSSI4

.
.
/*********************************************************************/
/*                       VSWITCH Section                             */
/*********************************************************************/
ITSOSSI1: Begin
   VMLAN MACPREFIX 021111 USERPREFIX 02AAAA
   RDEVICE 3080-308F  EQID OSA1SET1 TYPE OSA
   DEFINE VSWITCH ITSOVSW1    RDEV 3083
ITSOSSI1: End

ITSOSSI2: Begin
   VMLAN MACPREFIX 022222 USERPREFIX 02AAAA
   RDEVICE 30A0-30AF  EQID OSA1SET1 TYPE OSA
   DEFINE VSWITCH ITSOVSW1    RDEV 30A3
ITSOSSI2: End

ITSOSSI3: Begin
   VMLAN MACPREFIX 023333 USERPREFIX 02AAAA
   RDEVICE 2100-210F  EQID OSA1SET1 TYPE OSA
   DEFINE VSWITCH ITSOVSW1    RDEV 2103
ITSOSSI3: End

ITSOSSI4: Begin
   VMLAN MACPREFIX 024444 USERPREFIX 02AAAA
   RDEVICE 2120-212F  EQID OSA1SET1 TYPE OSA
   DEFINE VSWITCH ITSOVSW1    RDEV 2123
ITSOSSI4: End

   MODIFY VSWITCH ITSOVSW1    GRANT ITSOLNX1
   MODIFY VSWITCH ITSOVSW1    GRANT ITSOLNX2
   MODIFY VSWITCH ITSOVSW1    GRANT ITSOLNX3
   MODIFY VSWITCH ITSOVSW1    GRANT ITSOLNX4
   MODIFY VSWITCH ITSOVSW1    GRANT ITSOLNX5
   MODIFY VSWITCH ITSOVSW1    GRANT ITSOLNX6
   MODIFY VSWITCH ITSOVSW1    GRANT ITSOLNX7
   MODIFY VSWITCH ITSOVSW1    GRANT ITSOLNX8
   MODIFY VSWITCH ITSOVSW1    GRANT ITSOLNX9

/*********************************************************************/
/*                    Console Definitions                            */
/*********************************************************************/

ITSOSSI1: Begin
  Operator_Consoles           F200 ,
                              System_3270 System_Console
  Emergency_Message_Consoles  F200 ,
```

```
                              System_Console
ITSOSSI1: End

ITSOSSI2: Begin
  Operator_Consoles           F280 ,
                              System_3270 System_Console
  Emergency_Message_Consoles  F280 ,
                              System_Console
ITSOSSI2: End

ITSOSSI3: Begin
  Operator_Consoles           F300 ,
                              System_3270 System_Console
  Emergency_Message_Consoles  F300 ,
                              System_Console
ITSOSSI3: End

ITSOSSI4: Begin
  Operator_Consoles           F380 ,
                              System_3270 System_Console
  Emergency_Message_Consoles  F380 ,
                              System_Console
ITSOSSI4: End

/*********************************************************************/
```

We tested this CONFIG file with the new format of the CPSYNTAX command, as shown in Example 4-2.

Example 4-2 CPSYNTAX command

```
cpsyntax system config z (lpar a1b
CONFIGURATION FILE PROCESSING COMPLETE -- NO ERRORS ENCOUNTERED.
Ready; T=0.27/0.27 14:54:19
cpsyntax system config z (lpar a2b
CONFIGURATION FILE PROCESSING COMPLETE -- NO ERRORS ENCOUNTERED.
Ready; T=0.27/0.27 14:54:23
cpsyntax system config z (lpar a2a
CONFIGURATION FILE PROCESSING COMPLETE -- NO ERRORS ENCOUNTERED.
Ready; T=0.27/0.27 14:54:28
cpsyntax system config z (lpar a1a
CONFIGURATION FILE PROCESSING COMPLETE -- NO ERRORS ENCOUNTERED.
Ready; T=0.27/0.27 14:54:39
```

Next, we restarted all the members of the cluster.

Example 4-3 is the command to shut down to restart all the members. We omitted several lines to improve the clarity of this example.

Example 4-3 Shut down

```
q ssi
SSI Name: ITSOSSIA
SSI Mode: Stable
Cross-System Timeouts: Enabled
SSI Persistent Data Record (PDR) device: SSIAC1 on 9A20
```

```
SLOT SYSTEMID STATE     PDR HEARTBEAT       RECEIVED HEARTBEAT
   1 ITSOSSI1 Joined    10/31/11   15:27:16 10/31/11   15:27:16
   2 ITSOSSI2 Joined    10/31/11   15:26:50 10/31/11   15:26:50
   3 ITSOSSI3 Joined    10/31/11   15:27:14 10/31/11   15:27:14
   4 ITSOSSI4 Joined    10/31/11   15:27:06 10/31/11   15:27:06
Ready; T=0.01/0.01 15:27:19
at itsossi1 cmd shutdown reipl
HCPSHU960I System shutdown may be delayed for up to 530 seconds
Ready; T=0.01/0.01 15:27:47
at itsossi2 cmd shutdown reipl
HCPSHU960I System shutdown may be delayed for up to 530 seconds
Ready; T=0.01/0.01 15:28:06
at itsossi4 cmd shutdown reipl
HCPSHU960I System shutdown may be delayed for up to 530 seconds
Ready; T=0.01/0.01 15:28:10
q ssi
SSI Name: ITSOSSIA
SSI Mode: Stable
Cross-System Timeouts: Enabled
SSI Persistent Data Record (PDR) device: SSIAC1 on 9A20
SLOT SYSTEMID STATE     PDR HEARTBEAT       RECEIVED HEARTBEAT
   1 ITSOSSI1 Down (shut down successfully)
   2 ITSOSSI2 Down (shut down successfully)
   3 ITSOSSI3 Joined    10/31/11   15:27:44 10/31/11   15:27:44
   4 ITSOSSI4 Down (shut down successfully)
Ready; T=0.01/0.01 15:28:13
shutdown reipl
SYSTEM SHUTDOWN STARTED
HCPSHU960I System shutdown may be delayed for up to 530 seconds
Ready; T=0.01/0.01 15:42:13
```

4.3.5 Setting up the software

In this section, we describe the software setup for our lab environment.

Dirmaint

We installed DIRMAINT using the instructions from the book, *z/VM V6R2.0 Getting Started with Linux on System z,* SC24-6194.

We defined MAINT and MAINT620 as administrators.

> **AUTOLOG1 PROFILE EXEC note:** With SSI, the PROFILE EXEC of AUTOLOG1 differs from previous z/VM versions. It was changed to start the DIRMAINT machine on the first member that is loaded after the start-up and the respective DIRMSAT*x* in the other members.

Remote Spooling Communications Subsystem

We configured RSCS according to the instructions in *RSCS Networking for z/VM, FL620 Program Directory,* GI11-9802-00. We used CTC connections among our four members and from one z/VM system to another z/VM system on the ITSO network.

Example 4-4 shows the main configurations changes to RSCS. We considered the following information in making these changes:

- SYSTEM NETID must be updated in two places: the MAINT620 490 mdisk (common to the four members) and on the MAINT 190 of each member. After the change to the 190 mdisk, update the segments with the PUT2PROD SAVECMS command in each SSI member.
- RSCS CONFIG file: The LOCAL definition is used to identify each member, and it does not need the LINKDEF/PARM. The same information is true for PROFILE GCS.
- In our tests, we do not use RSCSDNS and RSCSAUTH. We commented both RSCSDNS and RSCSAUTH out in the GCS profile. So, AUTOLOG1 will only XAUTOLOG GCS.

Example 4-4 RSCS definitions

```
---> SYSTEM NETID (MAINT620 490 and MAINT 190)
*CPUID NODEID NETID
1ADE50 ITSOSSI1 RSCS
1BDE50 ITSOSSI2 RSCS
2A3BD5 ITSOSSI3 RSCS
2B3BD5 ITSOSSI4 RSCS

---> RSCS CONFIG
*          Local Nodeid     Application ID
*          ------------     --------------
 LOCAL     ITSOSSI4         *  RSCS04
*
 LINKDEFINE WTSCVMXA    TYPE NJE LINE 5A9B QUEUE PRI NODE WTSCVMXA
 LINKDEFINE ITSOSSI1    TYPE NJE LINE 4A5B QUEUE PRI NODE ITSOSSI1
 LINKDEFINE ITSOSSI2    TYPE NJE LINE 4A6B QUEUE PRI NODE ITSOSSI2
 LINKDEFINE ITSOSSI3    TYPE NJE LINE 43AB QUEUE PRI NODE ITSOSSI3
 LINKDEFINE ITSOSSI4    TYPE NJE LINE 43AB QUEUE PRI NODE ITSOSSI4

*
*     Linkid   PARM Text
*     -------- ---------------------------------------------------------
 PARM WTSCVMXA STREAMS=2 MAXU=2 MAXD=10 LISTPROC=NO TA=1 TAPARM='TH=100'
 PARM ITSOSSI1 STREAMS=2 MAXU=2 MAXD=10 LISTPROC=NO TA=1 TAPARM='TH=100'
 PARM ITSOSSI2 STREAMS=2 MAXU=2 MAXD=10 LISTPROC=NO TA=1 TAPARM='TH=100'
 PARM ITSOSSI3 STREAMS=2 MAXU=2 MAXD=10 LISTPROC=NO TA=1 TAPARM='TH=100'
 PARM ITSOSSI4 STREAMS=2 MAXU=2 MAXD=10 LISTPROC=NO TA=1 TAPARM='TH=100'
*         What to
*         Reroute      Userid        Nodeid   Userid
*         -------      --------      -------  ------
 REROUTE  NOTRCVG  FOR  OPERATOR  TO  ITSOSSI1 OP1

---> Profile GCS
'RSCS START ITSOSSI1'
'RSCS START ITSOSSI2'
'RSCS START ITSOSSI3'
'RSCS START ITSOSSI4'
'RSCS START WTSCVMXA'

---> Profile EXEC, into AUTOLOG1 machine
00000 * * * Top of File * * *
00001 /*********************************************************************/
00002 /*  AUTOLOG1 PROFILE EXEC                                              */
```

```
00003 /*********************************************************************/
00004 --------------------  10  line(s) not displayed -------------------
00014 “PIPE CP XAUTOLOG GCS    “
00015 “PIPE CP SLEEP 30 SEC”

--> Changed PROFILE GCS, into GCS machine
/****************************************/
/* Profile GCS for the Recovery Machine */
/****************************************/

/*’XAUTOLOG RSCSDNS’ */
  ‘XAUTOLOG RSCS’
/*’XAUTOLOG RSCSAUTH’ */
```

Installing the programmable operator facility

We installed the programmable operator facility (PROP) by using the instructions in *z/VM V6R2.0 Getting Started with Linux on System z,* SC24-6194.

We used PROP to collect the consoles from z/VM and other machines automatically. We also created another CMS machine that is named VMLOGS to hold all console logs on DASD. We used this machine to test the shared spool functions.

The VMLOGS machine is unique. It runs on every member. All machines that need to hold their console logs on DASD include the following command in their PROFILE EXEC:

```
CP SPOOL CONSOLE START TO VMLOGS NAME’ userid() ‘CONSOLE’
```

When every console is closed, the shared spool sends the files to the VMLOGS machine.

Performance Toolkit

We installed the Performance Toolkit program by using the instructions in the *Program Directory for Performance Toolkit for VM,* which is at this website:

http://www.ibm.com/eserver/zseries/zvm/library/

We also used *z/VM V6R2.0 Getting Started with Linux on System z,* SC24-6194.

We used the LOCALMOD command, as instructed, for the FCONX $PROFILE file. So, this file became common for all members. We performed the remainder of the configuration as described in *z/VM V6R2.0 Getting Started with Linux on System z,* SC24-6194*.*

TCP/IP

To configure TCP/IP, we copied the PROFILE STCPIP to each TCP/IP 198 mdisk with the respective name of each member. We merged the definitions that were made by IPWIZARD in PROFILE TCPIP to enable FTPSERVE.

4.4 Relocation examples

In this section, we describe relocation scenarios from our lab environment.

4.4.1 Live guest relocation for Linux on System z

We created two Linux guest machines. One machine runs a Linux SUSE distribution and another machine runs a Red Hat Enterprise Linux distribution. Both machines adhere to the eligibility requirements that are necessary to use the relocate function.

Example 4-5 is a profile directory entry that we used when installing Linux on System z. The inclusion of CMS minidisks prevented this machine from being relocated.

Example 4-5 Profile directory entry to install a Linux guest

```
PROFILE LINCMS
  COMMAND CP SPOOL CONSOLE START TO VMLOGS
  IPL CMS PARM AUTOCR
  SPOOL 000C 2540 READER *
  SPOOL 000D 2540 PUNCH A
  SPOOL 000E 1403 A
  CONSOLE 009 3215 T
  LINK MAINT 0190 0190 RR
  LINK MAINT 019D 019D RR
  LINK MAINT 019E 019E RR
  LINK LINUXCMS 191 191 MR
```

Example 4-6 is the other profile entry. The LINDFLT profile does not include any minidisks that are defined on any local DASD. The LINDFLT profile adds the OPTION CHPIDV that is mandatory to relocate Linux machines. Before we put the Linux guest into production, we only needed to change the INCLUDE from LINCMS to LINDFLT to allow the relocate function to work.

Example 4-6 Profile directory entries for Linux guests

```
PROFILE LINDFLT
   CLASS G
   OPTION CHPIDV ONE
   VMRELOCATE ON
   CONSOLE 0009 3215 T
   SPOOL 000C 2540 READER *
   SPOOL 000D 2540 PUNCH A
   SPOOL 000E 1403 A
```

If the Linux guest needs a PROFILE EXEC, perhaps to run a CMS command before starting Linux, it must detach all local minidisks before the IPL of Linux.

When the PROFILE EXEC only contains CP commands, it is not necessary to IPL CMS. You can convert the PROFILE EXEC to command entries in its directory entry. To monitor the Linux guest while it runs, we added the commands that are shown in Example 4-7 to the profile directory entries. In a non-SSI environment, these commands reside in the PROFILE EXEC file of the Linux guest.

Example 4-7 CP commands that we added to the profile directory entry

```
   COMMAND CP SPOOL CONSOLE START TO VMLOGS
   COMMAND CP TERM LINEND %
   COMMAND CP SET PF12 RETRIEVE
   COMMAND CP TERM MORE 1 0
   COMMAND CP SET RUN ON
```

```
   COMMAND CP TERM HOLD OFF
   COMMAND CP SET OBSERVER OPERATOR
```

Example 4-8 is the directory entry for the Linux SUSE guest ITSOLNX3.

Example 4-8 Linux guest ITSOLNX3 running SUSE

```
USER ITSOLNX3 ITSOSSI 6G 6G G
*
* SLES11 - SP1
* 0201 = swap space
* 0202 = / root fs
*
   INCLUDE LINDFLT
   IPL 202
   MACH ESA 2
   MDISK 0201 3390 0001 1000 LX9B25 MR
   MDISK 0202 3390 1001 9016 LX9B25 MR
```

Example 4-9 is the directory entry for the Red Hat guest ITSOLNX4.

Example 4-9 Linux guest ITSOLNX4 running Red Hat

```
USER ITSOLNX4 ITSOSSI 4G 4G G
*
* RHEL 5.6
* 0201 = swap space
* 0202 = / root fs
*
   INCLUDE LINDFLT
   IPL 202
   MACH ESA 2
   MDISK 0201 3390 0001 1000 LX9B26 MR
   MDISK 0202 3390 1001 9016 LX9B26 MR
```

When these machines are running in disconnected mode, we can issue the VMRELOCATE command to check their eligibility for relocation or move them to any member in the cluster. Example 4-10 on page 59 shows the sequence of commands to use.

The first query command shows the two machines running at ITSOSSI2 on the z10. The VMRELOCATE TEST command checks whether the resources that the machines use are available on the target member ITSOSSI3.

Command note: You can use the **AT<*member*> CMD** command to send the VMRELOCATE command to the member where the machine is running.

After a positive response, we transferred the machines using the command VMRELOCATE MOVE.

The last two queries in Example 4-10 on page 59 show the machines running on ITSOSSI3 on the z196.

All of this activity is transparent to the applications that run on both Linux guests. The PING command did not lose any packets.

Example 4-10 VMRELOCATE commands

```
q user itsolnx3 at all
ITSOSSI2: ITSOLNX3 - DSC
Ready; T=0.01/0.01 13:59:44
q user itsolnx4 at all
ITSOSSI2 : ITSOLNX4 - DSC
Ready; T=0.01/0.01 13:59:50
at itsossi2 cmd vmrelocate test user itsolnx4 to ITSOSSI3
User ITSOLNX4 is eligible for relocation to ITSOSSI3
Ready; T=0.01/0.01 14:11:24
at itsossi2 cmd vmrelocate test user itsolnx3 to ITSOSSI3
User ITSOLNX3 is eligible for relocation to ITSOSSI3
Ready; T=0.01/0.01 14:11:49
at itsossi2 cmd vmrelocate move user itsolnx3 to ITSOSSI3
Relocation of ITSOLNX3 from ITSOSSI2 to ITSOSSI3 started
User ITSOLNX3 has been relocated from ITSOSSI2 to ITSOSSI3
Ready; T=0.01/0.01 14:12:31
at itsossi2 cmd vmrelocate move user itsolnx4 to ITSOSSI3
Relocation of ITSOLNX4 from ITSOSSI2 to ITSOSSI3 started
User ITSOLNX4 has been relocated from ITSOSSI2 to ITSOSSI3
Ready; T=0.01/0.01 14:12:58
q user itsolnx3 at all
ITSOSSI3: ITSOLNX3 - DSC
Ready; T=0.01/0.01 14:13:23
q user itsolnx4 at all
ITSOSSI3 : ITSOLNX4 - DSC
Ready; T=0.01/0.01 14:13:30
close cons
```

See Appendix A, “Frequently asked questions” on page 87 for the other scenarios that we tested.

The Linux view of relocation

We captured the window that is shown in Figure 4-12 on page 60 from a Linux session that was collected during a relocate process. Figure 4-12 on page 60 shows that the relocation had no effect on the Linux guest.

We took the first **vmcp**[1] and **bonnie**[2] commands while they were running on the ITSOSSI1 member on the z10 machine.

We sent the **relocate** command while running the second **bonnie** command, at the **rewriting** step.

We took the last **vmcp** and **bonnie** commands while they were running on the ITSOSSI3 member in the z196 machine.

[1] vmcp is a way to send commands to the Control Program, when running under z/VM. It is part of the s390-tools package.

[2] bonnie is a performance benchmark program that targets various aspects of UNIX file systems. It was installed from the basic SUSE software packages.

```
itsolnx2:~ # vmcp 'QUERY USERID *'
ITSOLNX2 AT ITSOSSI1
itsolnx2:~ # bonnie -s 500
Bonnie 1.4: File './Bonnie.2548', size: 524288000, volumes: 1
Writing with putc()...         done:  34243 kB/s  98.4 %CPU
Rewriting...                   done: 116362 kB/s  14.5 %CPU
Writing intelligently...       done: 380033 kB/s  96.3 %CPU
Reading with getc()...         done:  36318 kB/s  96.3 %CPU
Reading intelligently...       done: 1305160 kB/s  99.2 %CPU
Seeker 3...Seeker 2...Seeker 1...start 'em...done...done...done...
              ---Sequential Output (nosync)--- ---Sequential Input-- --Rnd Seek-
              -Per Char- --Block--- -Rewrite-- -Per Char- --Block--- --04k (03)-
Machine    MB K/sec %CPU K/sec %CPU K/sec %CPU K/sec %CPU K/sec %CPU   /sec %CPU
itsoln 1* 500 34243 98.4380033 96.3 116362 14.5 36318 96.31305160 99.2 67567.6 88.9
itsolnx2:~ # bonnie -s 500
Bonnie 1.4: File './Bonnie.2552', size: 524288000, volumes: 1
Writing with putc()...         done:  34220 kB/s  98.5 %CPU
Rewriting...                   done: 117170 kB/s  15.0 %CPU
Writing intelligently...       done: 357185 kB/s  82.5 %CPU
Reading with getc()...         done:  33200 kB/s  90.7 %CPU
Reading intelligently...       done: 3512840 kB/s 102.7 %CPU
Seeker 3...Seeker 2...Seeker 1...start 'em...done...done...done...
              ---Sequential Output (nosync)--- ---Sequential Input-- --Rnd Seek-
              -Per Char- --Block--- -Rewrite-- -Per Char- --Block--- --04k (03)-
Machine    MB K/sec %CPU K/sec %CPU K/sec %CPU K/sec %CPU K/sec %CPU   /sec %CPU
itsoln 1* 500 34220 98.5357185 82.5 117170 15.0 33200 90.73512840  103 176087.3 70.7
itsolnx2:~ # vmcp 'QUERY USERID *'
ITSOLNX2 AT ITSOSSI3
itsolnx2:~ # bonnie -s 500
Bonnie 1.4: File './Bonnie.2560', size: 524288000, volumes: 1
Writing with putc()...         done:  40578 kB/s  99.0 %CPU
Rewriting...                   done: 218718 kB/s  13.2 %CPU
Writing intelligently...       done: 471186 kB/s  92.4 %CPU
Reading with getc()...         done:  42734 kB/s  96.3 %CPU
Reading intelligently...       done: 3554839 kB/s 103.9 %CPU
Seeker 3...Seeker 2...Seeker 1...start 'em...done...done...done...
              ---Sequential Output (nosync)--- ---Sequential Input-- --Rnd Seek-
              -Per Char- --Block--- -Rewrite-- -Per Char- --Block--- --04k (03)-
Machine    MB K/sec %CPU K/sec %CPU K/sec %CPU K/sec %CPU K/sec %CPU   /sec %CPU
itsoln 1* 500 40578 99.0471186 92.4 218718 13.2 42734 96.33554839  104 1254.4  0.6
itsolnx2:~ #
```

Figure 4-12 Inside Linux

4.5 Problems encountered

Proper planning of the environment means that even a complex setup of a four-member SSI cluster installation can be seamless and without challenges. However, we strongly suggest that you research the new syntax of commands, such as the PUT2PROD SAVECMS command, so that you will be familiar with them.

Also, if you are an existing z/VM user, remember the new enhancements and commands, especially for DIRMAINT, such as the use of the new universal IDs as IDENTITY. IDENTITY and SUBCONFIG are entries that are treated separately. You must run the commands DIRM GET and DIRM REPLACE to use them.

4.6 Lessons learned

Initial planning is key to the success of an SSI installation. Careful and proper planning is crucial, especially in the area of the CTC definitions on the ISFC and RSCS, if used.

DIRMAINT is a suggested feature. It saves time and effort in the setup of USER and IDENTITY entries and their minidisk definitions across all four members of the SSI cluster.

5

Scenario two: Converting two existing stand-alone z/VM systems to a single system image cluster

This chapter describes converting two existing stand-alone z/VM systems into a single system image (SSI) cluster with two members. This section describes our actions in detail, the problems that we encountered, the lessons that we learned, and our suggestions.

5.1 Overview

This scenario is typical for most existing clients when they migrate from stand-alone z/VM systems. In this scenario, we created two non-SSI z/VM systems and placed them into an SSI cluster. In the installation methodology, we selected a non-SSI installation in the initial installation panels, which resulted in a single z/VM system. Even though the installation is designed for a single z/VM image, use this method to migrate from a non-SSI environment to an SSI cluster more easily. In addition to the system residence, spool and work volumes, we must specify the common volumes and the release volumes even for a non-SSI installation.

Unlike the SSI installation, which requires 3390 volumes, you can perform non-SSI installations on Small Computer System Interface (SCSI) volumes. However, these non-SSI installations need one extended count key data (ECKD) volume for the Persistent Data Record (PDR) and two ECKD volumes for your IBM RACF® databases. The IPL packs can still be SCSI, if you are converting from non-SSI to SSI.

If you want to migrate your existing stand-alone z/VM systems running z/VM 5.3 or higher, perform the following tasks:

1. Upgrade one of the existing stand-alone z/VM systems to a Version 6.2 stand-alone non-SSI z/VM system.
2. Migrate this z/VM Version 6.2 stand-alone non-SSI system to a single-member SSI cluster.
3. Clone the SSI system.
4. Move the guests from their remaining pre-Version 6.2 stand-alone z/VM systems onto the members of the SSI cluster.

In our environment, each z/VM system runs on a separate central processor complex. All DASDs are visible to both logical partitions (LPARs) with common addressing. Figure 5-1 on page 63 shows our scenario.

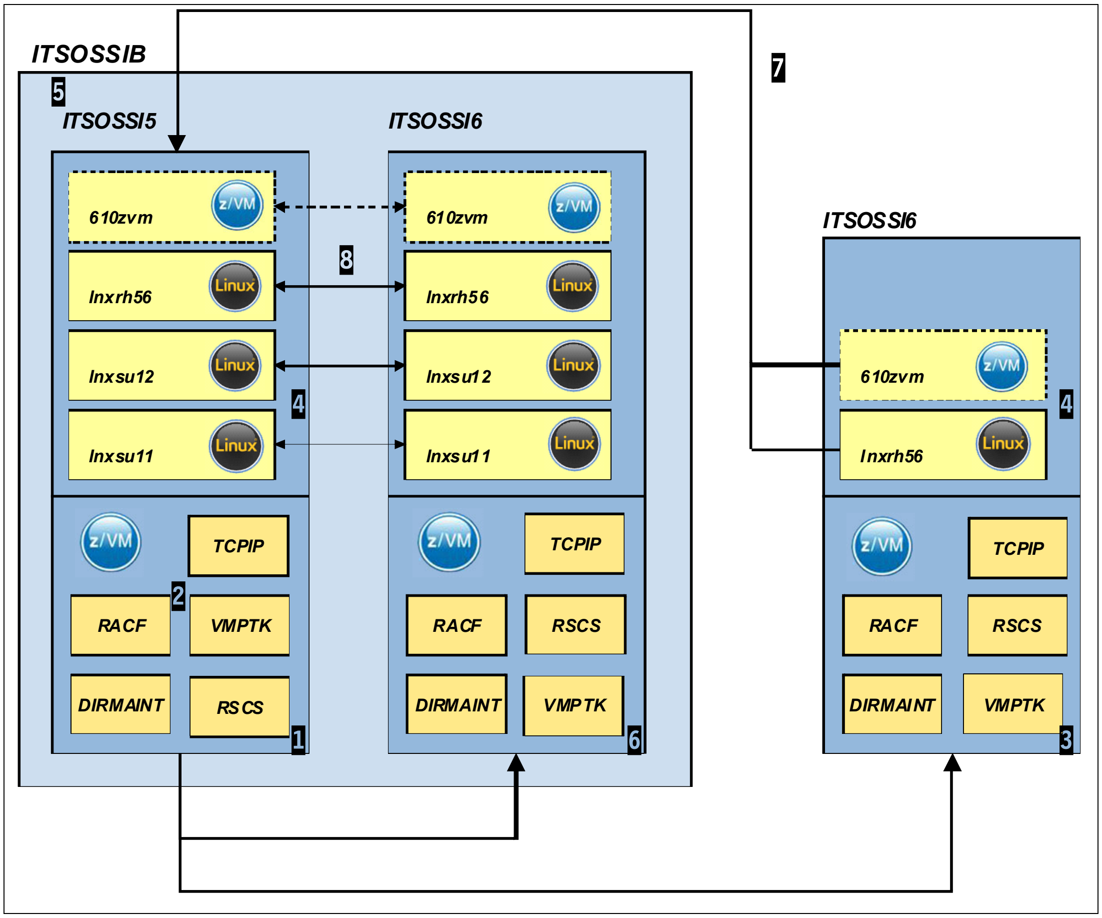


Figure 5-1 Overview of scenario two

5.1.1 Preparing and executing scenario two

The following numbers correspond to the numbers that are shown in Figure 5-1:

1 Create non-SSI system ITSOSSI5.

2 Enable and configure the following software products:

- Directory Maintenance Facility (DIRMAINT)
- IBM Resource Access Control Facility (RACF) Security Manager
- Remote Spooling Communication Subsystem (RSCS)
- TCP/IP
- Performance Toolkit for z/VM (VMPTK)

3 Create a second non-SSI system ITSOSSI6. For simplicity, we created a clone from ITSOSSI5.

4 Create guests on each non-SSI system. We created Linux guests LNXSU11 and LNXSU12 on ITSOSSI5. We created z/VM guest 610ZVM and Linux guest LNXRH56 on ITSOSSI6.

5 Convert the non-SSI system ITSOSSI5 to a single-member z/VM SSI cluster ITSOSSIB.

6 Add an additional member (ITSOSSI6) to the z/VM SSI cluster ITSOSSIB by cloning the existing member ITSOSSI5.

7 Move the guests from the non-SSI system ITSOSSI6 (610ZVM and LNXRH56) to the SSI cluster.

8 Start the relocation tests.

We describe these steps in more detail in the following sections.

5.2 Setting up ITSOSSI5

We started by creating ITSOSSI5, a first-level stand-alone non-SSI z/VM system at Version 6.2 with Directory Maintenance, RACF, TCP/IP, RSCS, and VMPTK enabled.

5.2.1 Planning worksheet

We created a spreadsheet based on the installation planning worksheet in the *z/VM V6R2.0 Installation Guide,* GC24-6246, which contained the values that are described in the following sections. Figure 5-2 shows DVD Installation Worksheet 1, which provides the details of the type of installation that is being performed.

DVD Installation Worksheet 1

Installation method (first-level or second-level) ____First____

Record an"M" if you will load the product to a minidisk or an "F" if you will load the product to the VMSYS file pool in the **Install To** column

Install To	Product	Install To	Product	Install To	Product
M	VM	M	OSA	M	PERFTK
M	VMHCD	M	RACF	M	DIRM
M	RSCS	M	ICKDSF	M	TCPIP

Default system lanuguage: ____ameng____

DASD type and model: ____3390-9____

SCSI Volume size: ________

Common service filepool name: ____SSISFSPB____

Installation Type:

____x____ Non-SSI System Name: ____ITSOSSI5____

________ SSI Number of Members________ SSI Cluster Name: ________

Figure 5-2 ITSOSSI5 Installation Worksheet 1

5.2.2 Defining the hardware

Stand-alone non-SSI z/VM systems at Version 6.2 have a new DASD configuration that matches the DASD configuration of an SSI member. They have one or more common and release disks, which facilitate future migrations to an SSI configuration. ITSOSSI5 runs on a System z10. Table 5-1 on page 65 shows our ITSOSSI5 system hardware definition.

Table 5-1 ITSOSSI5 definition

Device type	Volume serial number	Address	Use
3390-09	SSIBC1	9D20	Common disk
3390-09	SSIBR1	9D21	Release disk
3390-09	SSI5RS	9D22	SYSRES volume
3390-09	SSI5S1	9D23	Spool volume
3390-09	SSI5P1	9D24	Paging volume
Device type		**Address**	
OSA		3080	
OSA		3081	
OSA		3082	
TPCIP		9.12.4.236	
RSCS		4A90	

5.2.3 Defining the software

We enabled and configured the following software products:

- Directory Maintenance Facility (DIRMAINT)[1]
- RACF[1]
- RSCS[1]
- TCP/IP
- Performance Toolkit (VMPTK)[1]

5.2.4 Installing the z/VM stand-alone client

We installed z/VM Version 6.2 from the supplied DVDs and followed the installation instructions in the *z/VM V6R2.0 Installation Guide,* GC24-6246.

We entered the following information on the z/VM Installation Planning window that is shown in Figure 5-3 on page 66:

- We selected the following products to be installed to minidisks by typing `M` to the left of the product name:
  - VM
  - VMHCD
  - RSCS
  - OSA
  - RACF
  - ICKDSF
  - PERFTK
  - DIRM
  - TCP IP
- For the System Default Language, we selected **AMENG**.

[1] Optional priced features that are preinstalled on the z/VM V6.2 base product media

- For the System DASD model, we selected **3390 Mod 9**. For the FBA DASD size, we used an FBA DASD size of 6.0.
- For the System Type, we selected **SSI Install**. For the number of members, we typed `4`. For the SSI Cluster Name, we typed `ITSOSSIA`.

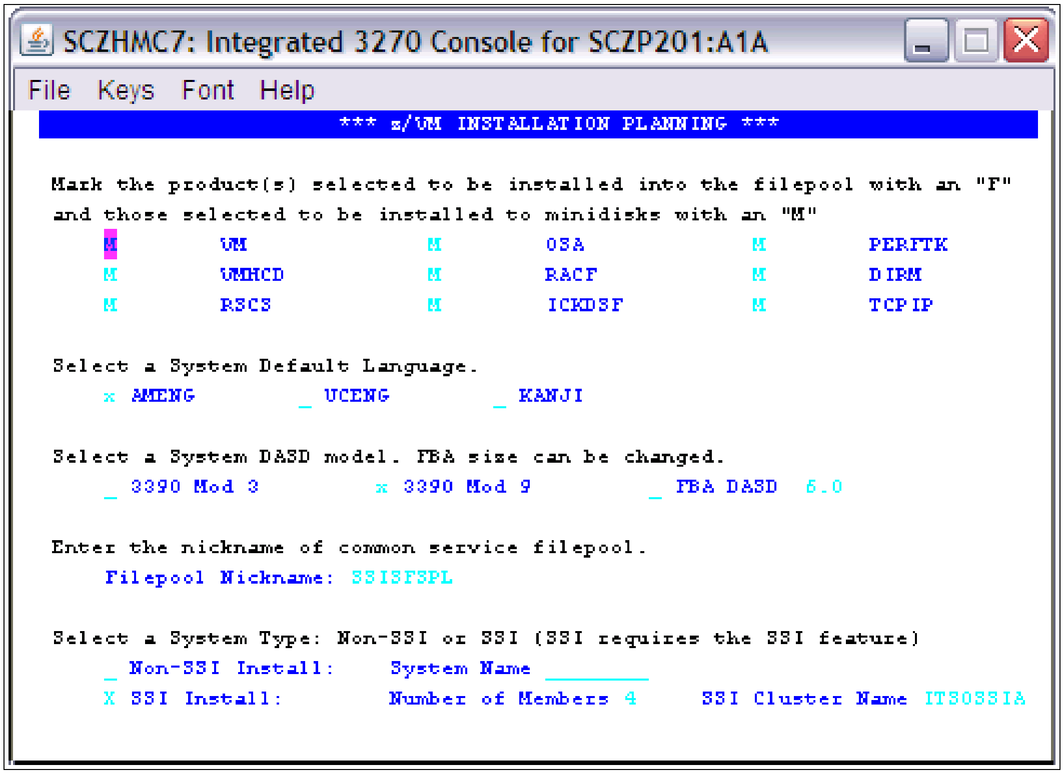


Figure 5-3 First z/VM Installation Planning window

5.2.5 Installing additional software products

We then enabled the products that are defined in 5.2.3, “Defining the software” on page 65 by using the instructions in the appropriate program directory. For our z/VM environment, we mainly followed the required installation steps in the program directories. For actual environments, it might be necessary to implement certain optional steps. For example, in the RACF environment in non-lab environments, it might be necessary to determine audit and control options for VM events. These details are beyond the scope of our book and, therefore, are not described.

We set up the DIRMAINT-RACF connector after DIRMAINT was installed as an optional step during the RACF setup. This functionality facilitates the relationship between DIRMAINT and RACF.

5.3 Setting up ITSOSSI6

We created ITSOSSI6 by cloning ITSOSSI5, with the same software products enabled.

5.3.1 Defining the hardware

ITSOSSI6 runs on a System z196. Table 5-2 shows our ITSOSSI6 system hardware definition. Both logical partitions (LPARs) share all DASD and have the same addressing.

Table 5-2 ITSOSSI6 definition

Device type	Volume serial number	Address	Use
3390-09	SSI6C1	9C25	Common disk
3390-09	SSI6R1	9C26	Release disk
3390-09	SSI6RS	9D26	SYSRES volume
3390-09	SSI6S1	9D27	Spool volume
3390-09	SSI6P1	9C27	Paging volume
Device type		**Address**	
OSA		2100	
OSA		2101	
OSA		2102	
TPCIP		9.12.4.237	
RSCS		4A90	

5.3.2 Defining the software

As a clone of ITSOSSI5, ITSOSSI6 has all of the same software enabled, that is, Directory Maintenance[2], RACF[2], TCP/IP, RSCS[2], and VMPTK[2].

5.3.3 Installation

We performed the following installation steps.

Setting up the DASD

We performed the following installation steps:

1. We attached the DASD to ITSOSSI5 and performed a dynamic device reconfiguration (DDR) copy of each of the volumes onto the DASD for ITSOSSI6. This copy of the volumes resulted in a second set of volumes with the same labels and owners as the ITSOSSI5 volumes.
2. You must not change the labels and owners of the DASD, because the SYSTEM CONFIG file still contains the ITSOSSI5 system ID. It is not possible to change it until the system is running. Also, the user directory contains references to the ITSOSSI5 volumes.

[2] Optional priced features that are preinstalled on the z/VM V6.2 base product media

3. We started ITSOSSI6 as a second-level guest under ITSOSSI5, ensuring that the correct set of volumes was attached to the user ID from which it was started. First, we performed these steps:
   a. To start ITSOSSI6, we wrote a new copy of the stand-alone program loader on ITSOSSI6 sysres, because we needed to modify the common volume address. We wrote this new copy by entering the command:
      ```
      salipl 9d26 (extent 1 iplparms fn=SYSTEM ft=CONFIG pdnum=1 pdvol=9c25
      ```
   b. We redefined the console 009 as 020 to enter the IPL parameters ‘CLEAN NOAUTOLOG’.
   c. We performed the IPL from 9D26, which is the ITSOSSI6 sysres.
4. We performed XAUTOLOG RACFVM to log on to MAINT620. From MAINT620, we changed the label and the owner of all the CP-owned volumes to reflect the ITSOSSI6 configuration by issuing the following commands:
   ```
   CPFMTXA 123 SSI6RS LABEL
   CPFMTXA 122 SSI6S1 LABEL
   CPFMTXA 131 SSI6R1 LABEL
   LINK PMAINT 141 141 MR
   CPFMTXA 141 SSI6C1 LABEL
   LINK $PAGE$ A01 A01 MR
   CPFMTXA A01 SSI6P1 LABEL
   CPFMTXA 141 SSI6C1 OWNER NOSSI ITSOSSI6
   CPFMTXA 131 SSI6R1 OWNER NOSSI ITSOSSI6
   CPFMTXA 123 SSI6RS OWNER NOSSI ITSOSSI6
   CPFMTXA 122 SSI6S1 OWNER NOSSI ITSOSSI6
   CPFMTXA A01 SSI6P1 OWNER NOSSI ITSOSSI6
   ```

Setting up the other system files

We performed the following installation steps:

1. We linked the PMAINT CF0 disk to change the system ID and DASD references in the SYSTEM CONFIG file. We reissued the SALIPL command, because we changed the label on the common volume.
2. We rebuilt the saved segments, which were deleted by the CLEAN start, by issuing the following commands:
   ```
   SERVICE CMS BLDNUC
   PUT2PROD
   ```
3. We logged on to PMAINT and amended the VM SYSPINV file on the 41D disk. If you do not perform this step, it is impossible to apply service to this z/VM image.

Setting up the directory

We performed the following installation steps:

1. We performed the XAUTOLOG DIRMAINT and issued the command DIRM USER WITHPASS to obtain a copy of the current user directory.
2. We amended the USER WITHPASS file to change the system ID and DASD references.
3. We checked the syntax of USER WITHPASS by issuing the command DIRECTXA USER WITHPASS (on our system, the DIRECTXA module is on PMAINT551).
4. We shut down DIRMAINT and put the new directory online by issuing the commands:
   – DIRM SHUTDOWN
   – DIRECTXA USER WITHPASS

5. We executed the following steps to synchronize the online directory with the source copy that is held by DIRMAINT:
   a. Link DIRMAINT 1DF as MR.
   b. Access 1DF G.
   c. Delete the file USER DIRECT G.
   d. Copy the amended USER WITHPASS file to USER INPUT G.
6. We restarted DIRMAINT and issued the command DIRM USER WITHPASS to ensure that Directory Maintenance had the correct version of the directory.

Setting up TCP/IP

We performed the following installation steps:

1. We logged on to TCPMAINT.
2. We changed the PROFILE TCPIP and SYSTEM DTCPARMS on the TCPMAINT 198 disk to reflect the OSA address on ITSOSSI6.
3. We changed the TCPIP DATA on the TCPMAINT 592 disk to reflect the host name of ITSOSSI6.

Setting up RSCS

We performed the following installation steps:

1. We logged on to MAINT620, accessed 190 as G, and amended the SYSTEM NETID file to reflect the CPU ID and host IP address.
2. We logged on to 6VMRSC20 and amended the RSCS definitions for the links to the other z/VM systems.
3. We logged on to MAINT620 and built the CMS and GCS saved segments to include the amended SYSTEM NETID file by issuing the following commands:

```
SERVICE CMS BLDNUC
SERVICE GCS BLDNUC
PUT2PROD
```

Starting ITSOSSI6

We shut down ITSOSSI6 as a second-level z/VM guest. We performed an IPL on ITSOSSI6 on the z196.

5.4 Converting ITSOSSI5 to a member of an SSI cluster

In this section, we describe converting a system to become a member of an SSI cluster.

5.4.1 Setting RACF to use as a shared database

In an SSI cluster, the RACF primary and backup databases must be on full-pack minidisks and must be shared by all members of the cluster. We used the same size RACF databases that were defined on the stand-alone z/VM system. We performed the following steps:

1. We allocated two full-pack minidisks, 9A25 and 9B20, to hold the primary and the backup RACF databases.

2. We moved the RACF databases from minidisk to full-pack while keeping the current database allocation. See the “RACF Database on Full-pack Minidisk” section in the *z/VM V6R2.0 RACF Security Server System Programmer’s Guide*, SC24-6219.
3. We shared the RACF databases among the members. See the “Sharing RACF Databases in a z/VM Single System Image Cluster” section in the *z/VM V6R2.0 RACF Security Server System Programmer’s Guide*, SC24-6219.
4. We then restarted RACF to verify that these databases were usable.

5.4.2 Completing the conversion

We performed the following steps by following the directions in “Converting a z/VM System to a Single-Member z/VM SSI Cluster” in *z/VM V6R2.0 CP Planning and Administration*, SC24-6178-02:

1. Update the system configuration file.
2. Update the user directory. We added a full-pack minidisk to PMAINT for our shared user volume SSI5M1.
3. Prepare the control program (CP)-owned volumes. We changed the owner information on all of the CP-owned volumes, and we created the persistent data record (PDR).
4. Modify the start-up parameters for the VMPSFS file pool.
5. Shut down the system, and restart it into the SSI cluster.
6. Update the user directory to SSI-Enabled. We used Directory Maintenance (DIRMAINT) to update the user directory.

5.5 Creating an ITSOSSI6 system by cloning ITSOSSI5

Although a stand-alone z/VM system, ITSOSSI6, existed, it is easier to create an SSI member in the ITSOSSIB cluster by cloning ITSOSSI5 and moving the workload from the original ITSOSSI6 onto the new member.

5.5.1 Cloning ITSOSSI5

We followed the process that is described in the “Adding a Member to a z/VM SSI Cluster by Cloning an Existing Member” chapter in *z/VM V6R2.0 CP Planning and Administration*, SC24-6178-02. To be able to IPL the original non-SSI system, ITSOSSI6, we allocated three new volumes for the sysres, spooling, and paging. We named these volumes 9A27 (sysres), 9E2E (spooling), and 9E2F (paging).

Preparing the CP-owned volumes for ITSOSSI6

We prepared the new spooling and paging volumes, SSI6S1 and SSI6P1, by formatting, labelling, and allocating (PAGE and SPOL) the volumes and then setting up the SSI ownership of them. Example 5-1 shows the steps that we performed for the spool volume SSI6S1.

Example 5-1 Formatting the spool volume

```
cpfmtxa 9e2e ssi6s1
CPFMTXA:
FORMAT WILL ERASE CYLINDERS 00000-nnnnn ON DISK 9E2E
DO YOU WANT TO CONTINUE? (YES | NO)
```

```
yes
HCPCCF6209I INVOKING ICKDSF. ...
ICK00002I ICKDSF PROCESSING COMPLETE. MAXIMUM CONDITION CODE WAS 0
ENTER ALLOCATION DATA
TYPE CYLINDERS
spol 1-end
end
HCPCCF6209I INVOKING ICKDSF. ...
ICK00002I ICKDSF PROCESSING COMPLETE. MAXIMUM CONDITION CODE WAS 0

cpfmtxa 9e2e ssi6s1 owner itsossib.itsossi6
```

Modifying the TCP/IP configuration for ITSOSSI5 and ITSOSSI6

We performed these steps:

1. On MAINT620, we linked the TCPMAINT 198 disk and renamed PROFILE TCPIP to ITSOSSI5 TCPIP.
2. We copied this file to ITSOSSI6 TCPIP and updated it with the information for ITSOSSI6.
3. We created the following member-specific configuration files, ITSOSSI5 DTCPARMS and ITSOSSI6 DTCPARMS.
4. Next, we linked TCPMAINT 592 and updated the TCPIP DATA:

```
ITSOSSI5: HOSTNAME ITSOSSI5
ITSOSSI6: HOSTNAME ITSOSSI6
```

Copying the ITSOSSI5 sysres to the ITSOSSI6 sysres

On MAINT620, we copied SSI5RS to SSI6RS. MAINT620 already has a full-pack minidisk link to SSI5RS as virtual address 123. We attached device 9A27 (SSI6RS) to MAINT620. We performed the copy:

```
flashcopy 123 0 end to 9a27 0 end synchronous label ssi6rs
```

We formatted the checkpoint and warm-start areas on SSI6RS, as shown in Example 5-2. The information for the cylinders is on the SYSTEM_RESIDENCE statement in the SYSTEM CONFIG file on the PMAINT CF0 disk.

Example 5-2 Formatting the Checkpoint and warm-start areas

```
cpfmtxa 9a27 ssi6rs 21.9
INVOKING ICKDSF.
FORMAT WILL ERASE CYLINDERS 21-29 ON DISK vdev
DO YOU WANT TO CONTINUE? (YES | NO)
yes
HCPCCF6209I INVOKING ICKDSF. ...
ICK00002I ICKDSF PROCESSING COMPLETE. MAXIMUM CONDITION CODE WAS 0
ENTER ALLOCATION DATA
TYPE CYLINDERS
end
HCPCCF6209I INVOKING ICKDSF. ...
ICK00002I ICKDSF PROCESSING COMPLETE. MAXIMUM CONDITION CODE WAS 0

cpfmtxa 9a27 ssi6rs 30.9
INVOKING ICKDSF.
FORMAT WILL ERASE CYLINDERS 30-38 ON DISK vdev
DO YOU WANT TO CONTINUE? (YES | NO)
```

```
yes
HCPCCF6209I INVOKING ICKDSF. ...
ICK00002I ICKDSF PROCESSING COMPLETE. MAXIMUM CONDITION CODE WAS 0
ENTER ALLOCATION DATA
TYPE CYLINDERS
end
HCPCCF6209I INVOKING ICKDSF. ... ICK00002I ICKDSF PROCESSING COMPLETE. MAXIMUM
CONDITION CODE WAS 0
```

Updating the DIRMAINT configuration

We deviated from the steps that are shown in *z/VM V6R2.0 CP Planning and Administration*, SC24-6178-02. The manual instructs you to update the CONFIGSS DATADVH with the new satellite server and to use **datamove** to move the user IDs first. However, this approach caused problems with DIRMAINT, because these user IDs did not exist. It worked better to perform this step last.

We updated the EXTENT CONTROL file by adding the new ITSOSSI6 volumes to the REGIONS: and SSI_VOLUMES: sections. We then replaced the updated EXTENT CONTROL file and reloaded the data to activate the changes.

Updating the user directory

Follow these steps to update the user directory:

1. We added entries to the directory for all the system-allocated areas on DASD, such as the checkpoint area and the spool volumes. If you do not add the entries, DIRMAINT does not know about these areas and potentially can overwrite them with new minidisks for users that you add later. Example 5-3 shows our changes in the user directory. In our setup, we did not allocate any TDISK space. However, most installations define TDISK space, so ensure that the appropriate entries for TDISK space are added.

   Example 5-3 USER DIRECT showing system-allocated areas

   ```
   USER $ALLOC$ NOLOG
      MDISK 0A00 3390 000 001 SSIBC1 R
      MDISK 0A01 3390 000 001 SSIBR1 R
      MDISK 0A02 3390 000 001 SSI5RS R
      MDISK 0A03 3390 000 001 SSI6RS R
   *
   USER $DIRECT$ NOLOG
      MDISK 0A01 3390 001 020 SSI5RS R
      MDISK 0A02 3390 001 020 SSI6RS R
      MDISK 0B01 3390 131 100 SSIBC1 R
   *
   USER $SYSCKP$ NOLOG
      MDISK 0A01 3390 021 009 SSI5RS R
      MDISK 0A02 3390 021 009 SSI6RS R
   *
   USER $SYSWRM$ NOLOG
      MDISK 0A01 3390 030 009 SSI5RS R
      MDISK 0A02 3390 030 009 SSI6RS R
   *
   USER $PAGE$ NOLOG
      MDISK A01 3390 000 END SSI5P1 R
      MDISK A02 3390 000 END SSI6P1 R
   *
   ```

```
USER $SPOOL$ NOLOG
   MDISK 0A01 3390 000 END SSI5S1 R
   MDISK 0A02 3390 000 END SSI6S1 R
*
USER $TDISK$  NOLOG
*
```

2. We updated the directory to add the sysres for ITSOSSI6 to identify where to locate the object directory for the system:

   ```
   dirmaint directory change 1 ssi 123 3390 ssr5rs ssi6rs
   ```

3. We added the new, required DATAMOV2 and DIRMSAT2 entries. The supplied user directory includes commented-out entries for DATAMOV2 and DIRMSAT2, which can be used as templates for the new user IDs. Check that these entries include the minidisk definitions that are similar to the minidisk entries in DIRMSAT and DATAMOVE.
4. We also created DVHPROFA DIRMSAT2 on the DIRMAINT 191 disk by copying the existing DVHPROFA DIRMSAT file. We used the following DIRMAINT command:

   ```
   DIRM FILE DVHPROFA DIRMSAT2 A DVHPROFA DIRMSAT2 C
   ```

5. We updated the multiconfiguration virtual machine definitions to add the subconfiguration entries for ITSOSSI6. The DIRM SCAN BUILD ON ITSOSSI5 provides you with a list of all the subconfiguration entries that are defined for ITSOSSI5 (Example 5-4). Use this list as a reference for the new entries that are required for ITSOSSI6.

   Example 5-4 Sample output from the DIRM SCAN BUILD ON ITSOSSI5 command

   ```
   Userid:  ======================= Qualifying Record =====================
   MAINT       BUILD ON ITSOSSI5 USING SUBCONFIG MAINT-1
   AVSVM       BUILD ON ITSOSSI5 USING SUBCONFIG AVSVM-1
   TSAFVM      BUILD ON ITSOSSI5 USING SUBCONFIG TSAFVM-1
   GCS         BUILD ON ITSOSSI5 USING SUBCONFIG GCS-1
   AUDITOR     BUILD ON ITSOSSI5 USING SUBCONFIG AUDITR-1
   AUTOLOG1    BUILD ON ITSOSSI5 USING SUBCONFIG AUTLG1-1
   CMSBATCH    BUILD ON ITSOSSI5 USING SUBCONFIG CMSBAT-1
   DISKACNT    BUILD ON ITSOSSI5 USING SUBCONFIG DSKACT-1
   .
   .
   .
   VSMEVSRV    BUILD ON ITSOSSI5 USING SUBCONFIG VSMEVS-1
   DTCSMAPI    BUILD ON ITSOSSI5 USING SUBCONFIG DTCSMA-1
   LOHCOST     BUILD ON ITSOSSI5 USING SUBCONFIG LOHCOS-1
   ZVMLXTS     BUILD ON ITSOSSI5 USING SUBCONFIG ZVMLXT-1
   Scan Complete.
   ```

6. For each user ID that is listed in the scan output, we entered a command that is similar to this command:

   ```
   dirmaint add maint-2 like maint-1 build on itsossi6 in maint
   ```

   Although it is possible to set up a REXX program to perform this function, we do not suggest that you process this task as a REXX exec. We learned that DIRMAINT was unable to handle our multiple requests, and the results can be unpredictable.

   One suggested way to process multiple DIRMAINT commands is to use the DIRM BATCH command. Create a file with the correct DIRMAINT commands and execute them with DIRM BATCH. See Example 5-5 on page 74 and Example 5-6 on page 74.

Example 5-5 DIRM CMDS A batch file containing the DIRM statements

```
add maint-2 like maint-1 build on itsossi6 in maint
add avsvm-2 like avsvm-1 build on itsossi6 in avsvm
...
```

Example 5-6 Executing the batch file

```
dirm batch DIRM CMDS A
```

7. We obtained a copy of the source directory by issuing DIRM USER WITHPASS, and we copied this directory to the PMAINT 2CC disk as USER DIRECT.
8. Next, we created an object directory on ITSOSSI5 by issuing DIRM DIRECT.
9. We then created the object directory for ITSOSSI6, as shown in Example 5-7. We detached our link to the ITSOSSI5 sysres (device 123), attached SSI6RS to our user ID as disk 123, created the object directory on SSI6RS, detached SSI6RS, and reattached the ITSOSSI5 sysres.

Example 5-7 Creating new object directory for ITSOSSI6

```
detach 123
attach 9a27 to * as 123
directxa user direct c
detach 123
link maint 123 123 m
```

10. The last step in the DIRMAINT configuration was to update and reload the CONFIGSS DATADVH file with the new satellite and to datamove the user IDs for ITSOSSI6. Example 5-8 shows the contents of our CONFIGSS DATADVH file.

Example 5-8 Our CONFIGSS DATADVH

```
RUNMODE=                              OPERATIONAL
ONLINE=                               IMMED
SATELLITE_SERVER=                     DIRMSAT  ITSOSSI5
SATELLITE_SERVER=                     DIRMSAT2 ITSOSSI6
DATAMOVE_MACHINE=                     DATAMOVE ITSOSSI5 *
DATAMOVE_MACHINE=                     DATAMOV2 ITSOSSI6 *
```

Updating the SYSTEM CONFIG file for the ITSOSSIB cluster

We updated the SYSTEM CONFIG file on the PMAINT CF0 disk. Example 5-9 shows the statements that we updated for ITSOSSI6, but it does not include all of the statements that are contained in this file.

Example 5-9 Selected statements from our SYSTEM CONFIG file

```
/*********************************************************************/
/*                   System_Identifier Information                   */
/*********************************************************************/
System_Identifier LPAR A25 ITSOSSI5
System_Identifier LPAR A19 ITSOSSI6
/*********************************************************************/
/*                   SSI Cluster                                     */
/*********************************************************************/
SSI ITSOSSIB PDR_VOLUME SSIBC1,
          SLOT 1 ITSOSSI5,
```

```
          SLOT 2 ITSOSSI6
/**********************************************************************/
/*               Checkpoint and Warmstart Information                 */
/**********************************************************************/
ITSOSSI5: System_Residence,
    Checkpoint  Volid SSI5RS    From CYL 21  For 9 ,
    Warmstart   Volid SSI5RS    From CYL 30  For 9

ITSOSSI6: System_Residence,
    Checkpoint  Volid SSI6RS    From CYL 21  For 9 ,
    Warmstart   Volid SSI6RS    From CYL 30  For 9
/**********************************************************************/
/*                    CP_Owned Volume Statements                      */
/**********************************************************************/
ITSOSSI5: CP_Owned   Slot   1  SSI5RS
ITSOSSI6: CP_Owned   Slot   1  SSI6RS
/**********************************************************************/
CP_Owned   Slot   5  SSIBC1
/**********************************************************************/
CP_Owned   Slot  10  SSI5S1                        1
CP_Owned   Slot  11  SSI6S1
/**********************************************************************/
ITSOSSI5: CP_Owned   Slot 255  SSI5P1
ITSOSSI6: CP_Owned   Slot 255  SSI6P1
/**********************************************************************/
/* Shared User Volumes                                                */
/**********************************************************************/
User_Volume_List  SSIBR1
User_Volume_List  SSI5M1 SSI6M1 LX9A28 LX9A29
/**********************************************************************/
Devices ,
   Online_at_IPL   0000-FFFF,
   Offline_at_IPL  9C25-9C27 9D26 9D27,         2
   Sensed          0000-FFFF
/**********************************************************************/
/*               ISCF Links                                          */
/**********************************************************************/
ITSOSSI5: ACTIVATE ISLINK 4290 5290 NODE ITSOSSI6
ITSOSSI6: ACTIVATE ISLINK 5050 4050 NODE ITSOSSI5
/**********************************************************************/
/*               NETWORK DEFINITIONS                                  */
/**********************************************************************/
ITSOSSI5: DEFINE VSWITCH VSWITCH1 RDEV 3083 IP
ITSOSSI6: DEFINE VSWITCH VSWITCH1 RDEV 2103 IP
```

The following numbers refer to the numbers that are shown in Example 5-9 on page 74:

1 The spool volumes SSI5S1 and SSI6S1 do not have their member names as a qualifier, because they are shared volumes.

2 We added devices 9C25-9C27, 9D26, and 9D27 to `Offline_at_IPL` on the `Devices` statement to prevent problems with duplicate volsers at IPL time. These devices are the system volumes of the stand-alone ITSOSSI6 non-SSI system.

To ensure that no syntax errors occurred, we used the CPSYNTAX EXEC, which is on the MAINT 193 disk, to check the SYSTEM CONFIG file.

Updating the cluster ITSOSSIB for ITSOSSI6

Follow these steps:

1. We added ITSOSSI6 to the member list for the ITSOSSIB cluster:

   ```
   set ssi slot 2 itsossi6
   ```

2. We activated the ISFC links in preparation for the IPL of ITSOSSI6:

   ```
   ACTIVATE ISLINK 4290 5290 NODE ITSOSSI6
   ```

3. We defined the ITSOSSI6 spool volume to CP:

   ```
   define cpowned   slot  11  ssi6s1
   ```

4. Then, we were ready to IPL ITSOSSI6 for the first time as a member of the ITSOSSIB cluster.

Starting ITSOSSI6

We started ITSOSSI6 and specified CLEAN NOAUTOLOG. Then, we executed XAUTOLOG RACFVM.

Updating the VMSES/E system-level product inventory table

We logged on to MAINT620 on ITSOSSI6. We updated the Virtual Machine Serviceability Enhancements Staged/Extended (VMSES/E) system-level product inventory table (file VM SYSPINV on the PMAINT 41D disk) by issuing the following command:

```
vmfupdat syspinv system itsossi6 itsossi5
```

Building the saved segments and named saved systems

Follow these steps:

1. We updated the SYSTEM NETID on the MAINT 190 disk to include details of ITSOSSI6, as shown in Example 5-10.

   Example 5-10   SYSTEM NETID

   ```
   25DE50 ITSOSSI5 RSCS
   193BD5 ITSOSSI6 RSCS
   ```

2. We ran XAUTOLOG on the file pool servers, VMSERVS, VMSERVU, VMSERVR, and VMSERVP, and built the following CMS named saved system and the other saved segments and named saved systems:

   ```
   put2prod savecms
   put2prod segments all
   service gcs bldnuc
   put2prod
   ```

Starting the service virtual machines

Finally, we ran XAUTOLOG on AUTOLOG2 to start all the service virtual machines (SVMs) that are defined in its profile exec.

Migrating the workload from the stand-alone non-SSI ITSOSSI6

We moved the workload from the stand-alone non-SSI ITSOSSI6 to the new SSI member ITSOSSI6. We created the Linux guests and VM guest using the original directory definitions, as shown in Example 5-11 and Example 5-12.

Example 5-11 Directory entry for LNXRH65

```
USER LNXRH56 HCHT57UI 512M 1G G
   INCLUDE IBMDFLT
   IPL CMS PARM AUTOCR
   IUCV ALLOW
   IUCV ANY
   MACH ESA 4
   OPTION CHPIDV ONE
   POSIXINFO UID 59
   NICDEF C200 TYPE QDIO LAN SYSTEM VSWITCH1
   MDISK 0201 3390 1 1000 LX9A29 MR
   MDISK 0202 3390 1001 9016 LX9A29 MR
   MDISK 0191 3390 1 50 SSI6M1
```

Example 5-12 Directory entry for 610ZVM

```
USER 610ZVM 8D99H950 99M 999M BG
   INCLUDE IBMDFLT
   ACCOUNT ITS30000
   CPU 0 BASE CPUID 616161
   IPL CMS PARM AUTOCR
   MACH XA
   OPTION MAINTCCW DEVMAINT CHPIDV ONE
   DEDICATE D850 D850
   DEDICATE D851 D851
   DEDICATE D852 D852
   DEDICATE D853 D853
   DEDICATE D854 D854
   NICDEF 3080 TYPE QDIO LAN SYSTEM VSWITCH1
   SPECIAL 08E1 3270
   SPECIAL 08E2 3270
   SPECIAL 08E3 3270
   SPECIAL 08E4 3270
   SPECIAL 08E5 3270
   SPECIAL 08E6 3270
   MDISK 0191 3390 51 10 SSI6M1
```

5.6 Suggestions

In our scenario, cloning the SSI member ITSOSSI5 (see 5.5, “Creating an ITSOSSI6 system by cloning ITSOSSI5” on page 70) was complicated. If possible, we suggest that you build an SSI cluster, as described in Chapter 4, “Scenario one: Creating a single system image cluster with four new z/VM members” on page 35, and then move your guests into the new SSI cluster.

5.7 Relocation examples

Chapter 4, “Scenario one: Creating a single system image cluster with four new z/VM members” on page 35 provides examples of live guest relocation (LGR). We cover examples here that differ from the examples that are shown in scenario one.

5.7.1 Live guest relocation of a z/VM guest

Live guest relocation (LGR): LGR is officially supported for Linux guests running under z/VM 6.2 only. This z/VM guest relocation example is unsupported.

In this example, we tried the VMRELOCATE command on a second-level z/VM guest running Version 6.1. Before we issued the VMRELOCATE command, we logged on to MAINT on 610ZVM and ran an exec (see Example 5-13) to issue ping requests periodically. Example 5-14 show the VMRELOCATE command that we issued.

Example 5-13 Test exec

```
/**/
Address CMS
Do forever
'Q T'
'PING 9.12.4.236'
'CP SLEEP 10 SEC'
End
```

Example 5-14 VMRELOCATE command for 610ZVM guest

```
at itsossi5 cmd vmrelocate move 610zvm itsossi6
16:17:07 Relocation of 610ZVM from ITSOSSI5 to ITSOSSI6 started by MAINT620
16:17:07 Relocation of 610ZVM from ITSOSSI5 to ITSOSSI6 started
16:17:08 DASD D850 ATTACHED TO 610ZVM D850 BY 610ZVM WITH DEVCTL HYPERPAV BASE
16:17:08 DASD D853 ATTACHED TO 610ZVM D853 BY 610ZVM WITH DEVCTL HYPERPAV BASE
16:17:08 DASD D851 ATTACHED TO 610ZVM D851 BY 610ZVM WITH DEVCTL HYPERPAV BASE
16:17:08 DASD D854 ATTACHED TO 610ZVM D854 BY 610ZVM WITH DEVCTL HYPERPAV BASE
16:17:08 DASD D852 ATTACHED TO 610ZVM D852 BY 610ZVM WITH DEVCTL HYPERPAV BASE
610ZVM  : User 610ZVM has been relocated from ITSOSSI5 to ITSOSSI6
16:17:08 User 610ZVM has been relocated from ITSOSSI5 to ITSOSSI6
16:17:08 User 610ZVM has been relocated from ITSOSSI5 to ITSOSSI6
Ready; T=0.01/0.01 16:17:08
```

Example 5-15 on page 78 shows the output from the ping requests that were issued during the relocation request for 610ZVM.

Example 5-15 Response to PING command on 610ZVM during relocation

```
Ready; T=0.03/0.04 16:17:02
test
16:17:04 TIME IS 16:17:04 EST FRIDAY 11/11/11
16:17:04 CONNECT= 00:18:44 VIRTCPU= 000:00.59 TOTCPU= 000:00.79
Ping Level 610: Pinging host 9.12.4.236.
                Enter #CP EXT to interrupt.
PING: Ping #1 response took 0.001 seconds. Successes so far 1.
 16:17:08 HCPIPQ2091I I/O priority setting available to CP has changed.
```

```
 16:17:08 HCPIPQ2091I New setting is enabled with range 000 to 000.
16:17:14 TIME IS 16:17:14 EST FRIDAY 11/11/11
16:17:14 CONNECT= 00:18:54 VIRTCPU= 000:00.60 TOTCPU= 000:00.80
Ping Level 610: Pinging host 9.12.4.236.
                Enter #CP EXT to interrupt.
PING: Ping #1 response took 0.001 seconds. Successes so far 1.
16:17:24 TIME IS 16:17:24 EST FRIDAY 11/11/11
16:17:24 CONNECT= 00:19:03 VIRTCPU= 000:00.60 TOTCPU= 000:00.80
Ping Level 610: Pinging host 9.12.4.236.
                Enter #CP EXT to interrupt.
PING: Ping #1 response took 0.001 seconds. Successes so far 1.
16:17:34 TIME IS 16:17:34 EST FRIDAY 11/11/11
16:17:34 CONNECT= 00:19:14 VIRTCPU= 000:00.61 TOTCPU= 000:00.81
Ping Level 610: Pinging host 9.12.4.236.
                Enter #CP EXT to interrupt.
PING: Ping #1 response took 0.001 seconds. Successes so far 1.
```

5.7.2 Relocating an SAP system in a typical System z setup

This relocation example shows the relocation of an SAP Application Server in a typical user environment on System z. The IBM DB2 database server runs in a z/OS environment. The SAP Application Server Central Instance® is on a Linux on System z server. The SAP Application Server® is running on our z/VM system ITSOSSI5. The SAP Application Server Dialog Instance (DI) is running on an IBM AIX® system. See Figure 5-4 on page 80.

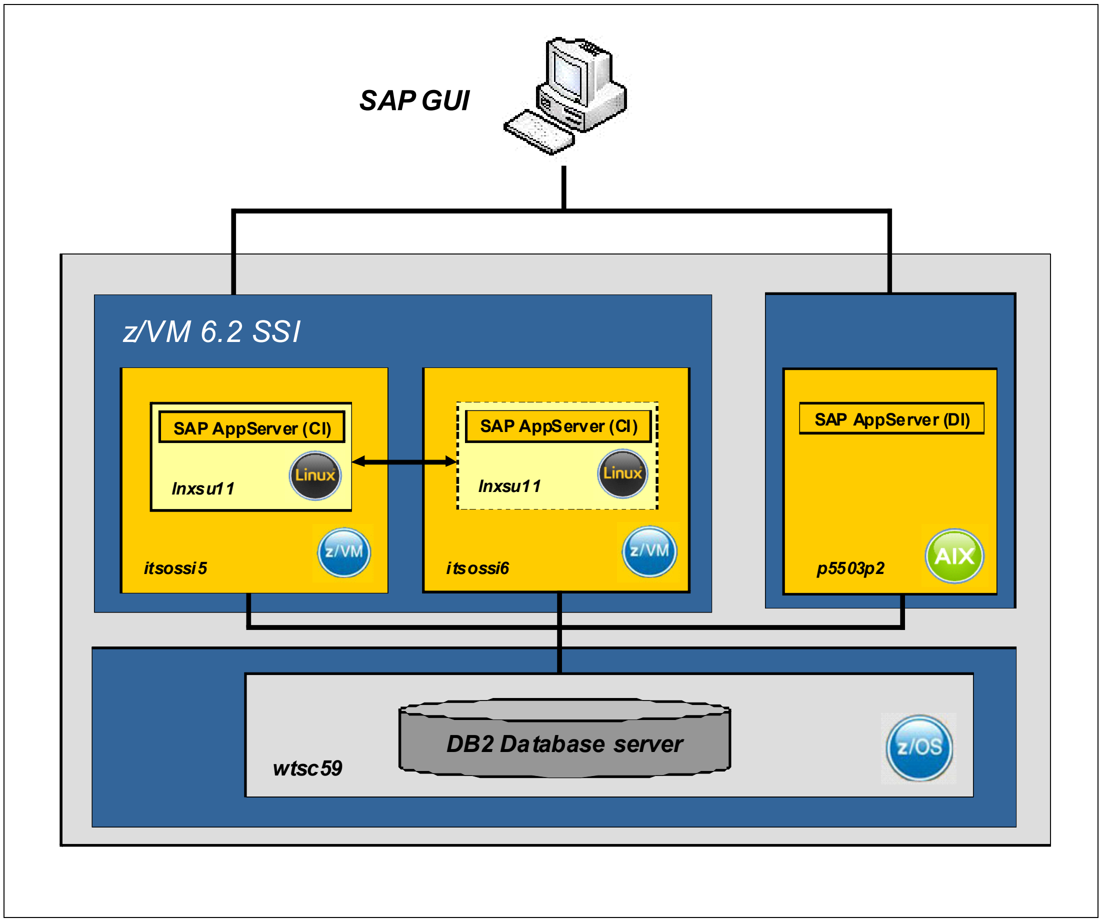


Figure 5-4 SAP solution on a System z with the relocation feature of z/VM 6.2

SAP is often used in a System z setup. An important requirement for SAP clients is 24x7 availability to help ensure business continuity, avoid business interruptions, and reduce lost revenue.

With LGR, z/VM 6.2 provides an additional feature for 24x7 availability of SAP systems. Now, you can relocate an SAP Application Server on a Linux for System z system non-disruptively. Use LGR for applying service to the z/VM system, for workload-balancing activities, or for migrations to new System z hardware. In our example, we relocate the SAP AppServer (CI) non-disruptively between the z/VM systems ITSOSSI5 and ITSOSSI6.

Example 5-16 shows the SAP AppServer (CI) guest definition in the z/VM directory.

Example 5-16 SAP guest definition

```
USER LNXSU11 XXXXXXXX 6G 16G G
   CPU 01
   CPU 00
   CPU 02
   CPU 03
   IPL 202
   MACHINE ESA 4
   OPTION CHPIDV ONE                                         1
   NICDEF C200 TYPE QDIO LAN SYSTEM VSWITCH1                 2
   NICDEF 6300 TYPE QDIO LAN SYSTEM VSW999 DEVICES 3
   NICDEF 6303 TYPE QDIO LAN SYSTEM VSW199 DEVICES 3
```

```
NICDEF 7000 TYPE QDIO LAN SYSTEM VSW199 DEVICES 3
SPOOL 000C 3505 A
SPOOL 000D 3525 A
SPOOL 000E 1403 A
CONSOLE 0009 3215 T
MDISK 0201 3390 1 10016 LX9980 MR                   3
MDISK 0202 3390 1 10016 LX9981 MR                   4
MDISK 0301 3390 1 10016 LX9982 MR                   5
MDISK 0302 3390 1 10016 LX9983 MR
MDISK 0303 3390 1 10016 LX9984 MR
MDISK 0304 3390 1 10016 LX9985 MR                   6
MDISK 0305 3390 1 10016 LX9986 MR
MDISK 0306 3390 1 10016 LX9987 MR                   7
MDISK 0307 3390 1 10016 LX9988 MR
MDISK 0308 3390 1 10016 LX9989 MR
MDISK 0309 3390 1 10016 LX998A MR
```

The following numbers refer to the numbers that are shown in Example 5-16 on page 80:

1 sets CHPID virtualization to `ONE`
2 vSwitch definitions
3 swap space
4 /root/ filesystem
5 /usr/sap/ filesystem
6 /sapmnt/ filesystem
7 /install/ filesystem

Relocation process

Follow these steps:

1. Issue the VMRELOCATE TEST command to check the z/VM guest eligibility.

   Example 5-17 VMRELOCATE TEST command

   ```
   vmrelocate test lnxsu11 itsossi6
   User LNXSU11 is eligible for relocation to ITSOSSI6
   Ready; T=0.01/0.01 16:17:56
   ```

2. Issue the VMRELOCATE MOVE command.

   Example 5-18 VMRELOCATE MOVE command

   ```
   vmrelocate move lnxsu11 itsossi6
   Relocation of LNXSU11 from ITSOSSI5 to ITSOSSI6 started
   User LNXSU11 has been relocated from ITSOSSI5 to ITSOSSI6
   Ready; T=0.01/0.01 16:19:12
   ```

Monitoring results

We monitored the relocation process for two scenarios.

Scenario one

We monitored for a relocation of the SAP Application Server (CI) without any load, which means without any SAP activities. For the monitoring, we issued the VMRELOCATE STATUS each 0.3 seconds to see the states of the relocation process. Example 5-19 on page 82 and

Example 5-20 on page 82 show the results of monitoring the relocation process without SAP activities.

Example 5-19 Performance data for SAP AppServer (CI) without load

```
top - 16:30:53 up 12:45,  2:45,  1 user,  load average: 0.01, 0.00, 0.00
Tasks: 101 total,   1 running, 100 sleeping,   0 stopped,   0 zombie
Cpu(s):  0.1%us,  0.0%sy,  0.0%ni, 99.9%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Mem:   6171128k total,  3947952k used,  2223176k free,   118032k buffers
Swap:  7211416k total,        0k used,  7211416k free,  3500412k cached
```

Example 5-20 Relocation time of SAP AppServer (CI) without load

```
16:33:49
  User     From     To       By       Status          Elapsed
  LNXSU11  ITSOSSI5 ITSOSSI6 ITSO     Moving Memory   00:00:00
.
.
.
16:33:56
  User     From     To       By       Status          Elapsed
  LNXSU11  ITSOSSI5 ITSOSSI6 ITSO     Moving Memory   00:00:07
16:33:56
  User     From     To       By       Status          Elapsed
  LNXSU11  ITSOSSI5 ITSOSSI6 ITSO     Moving Guest    00:00:07
16:33:56
  User     From     To       By       Status          Elapsed
  LNXSU11  ITSOSSI5 ITSOSSI6 ITSO     Final Mem Copy  00:00:08
16:33:58
  User     From     To       By       Status          Elapsed
  LNXSU11  ITSOSSI5 ITSOSSI6 ITSO     Final Mem Copy  00:00:09
16:33:58
  User     From     To       By       Status          Elapsed
  LNXSU11  ITSOSSI5 ITSOSSI6 ITSO     Cleaning Up     00:00:09
16:33:58
```

This scenario provided the following results:

- The elapsed time for the relocation process was 9 seconds.
- The quiesce time for the z/VM guest was about 2 seconds. See the status “`Final Mem Copy`”.

Scenario two

The second relocation of the SAP Application Server (CI) included active SAP applications. Example 5-21 and Example 5-22 on page 83 show the results of monitoring the relocation process with a full SAP load.

Example 5-21 Performance data for SAP AppServer (CI) with load

```
top - 09:50:07 up 23:19,  2 users,  load average: 2.65, 1.84, 1.41
Tasks: 131 total,   3 running, 128 sleeping,   0 stopped,   0 zombie
Cpu(s): 30.8%us,  3.2%sy,  0.0%ni, 62.2%id,  0.0%wa,  0.2%hi,  0.7%si,  2.8%st
Mem:   6171128k total,  5424008k used,   747120k free,   108564k buffers
Swap:  7211416k total,        0k used,  7211416k free,  4386852k cached
```

Example 5-22 Relocation of SAP AppServer (CI) with full SAP load

```
09:51:14
  User     From     To       By       Status         Elapsed
  LNXSU11  ITSOSSI5 ITSOSSI6 ITSO    Moving Memory   00:00:00
.
.
.
09:51:51
  User     From     To       By       Status         Elapsed
  LNXSU11  ITSOSSI5 ITSOSSI6 ITSO    Moving Memory   00:00:37
09:51:51
  User     From     To       By       Status         Elapsed
  LNXSU11  ITSOSSI5 ITSOSSI6 ITSO    Moving Guest    00:00:37
09:51:52
  User     From     To       By       Status         Elapsed
  LNXSU11  ITSOSSI5 ITSOSSI6 ITSO    Moving Guest    00:00:38
09:51:52
  User     From     To       By       Status         Elapsed
  LNXSU11  ITSOSSI5 ITSOSSI6 ITSO    Final Mem Copy  00:00:38
09:51:53
  User     From     To       By       Status         Elapsed
  LNXSU11  ITSOSSI5 ITSOSSI6 ITSO    Final Mem Copy  00:00:39
09:51:54
  User     From     To       By       Status         Elapsed
  LNXSU11  ITSOSSI5 ITSOSSI6 ITSO    Cleaning Up     00:00:40
09:51:54
  User     From     To       By       Status         Elapsed
  LNXSU11  ITSOSSI5 ITSOSSI6 ITSO    Cleaning Up     00:00:40
09:51:54
```

This scenario provided the following results:

- The elapsed time for the relocation process was 40 seconds.
- The quiesce time for the z/VM guest was about 2 seconds. See the status “`Final Mem Copy`”.

Figure 5-5 on page 84 and Figure 5-6 on page 85 show the results of the relocation, visible in the SAP Operation System Monitor (SAP transaction ST06).

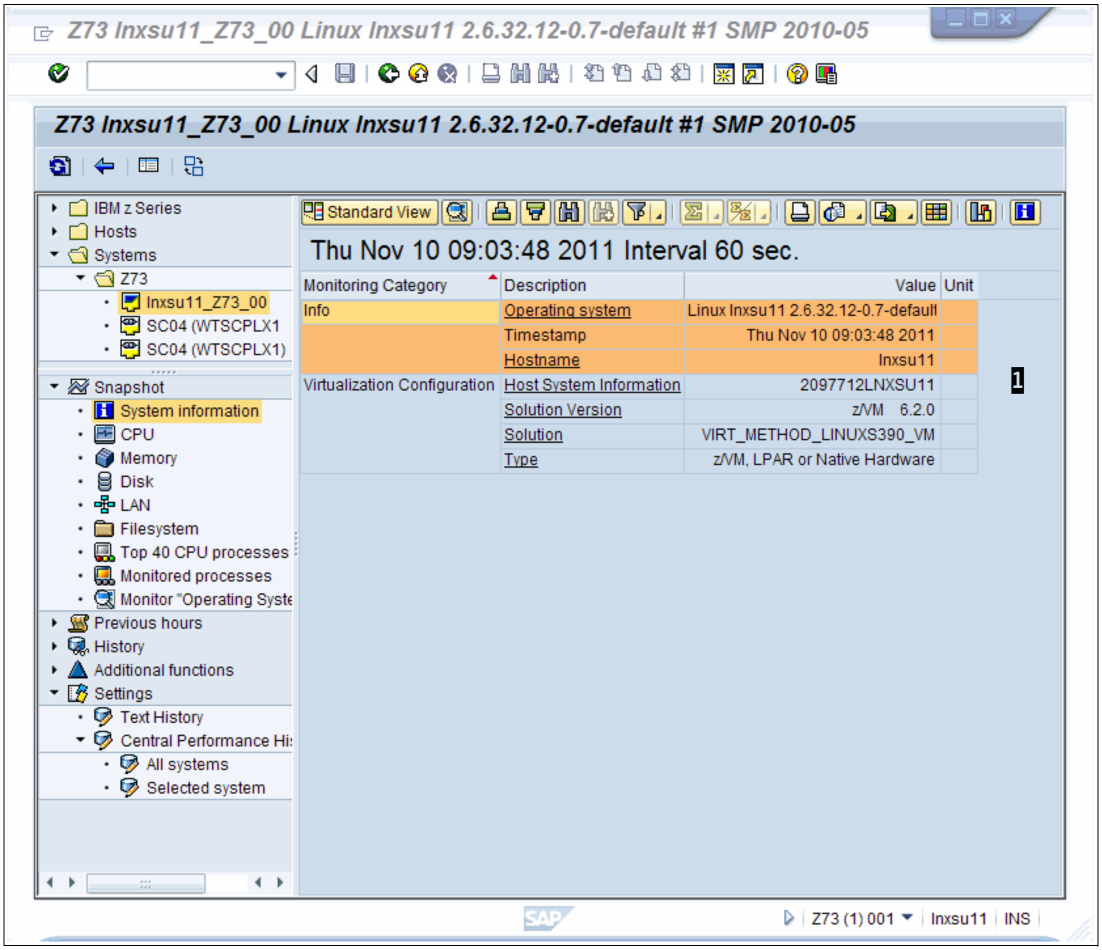


Figure 5-5 Host system information before the relocation

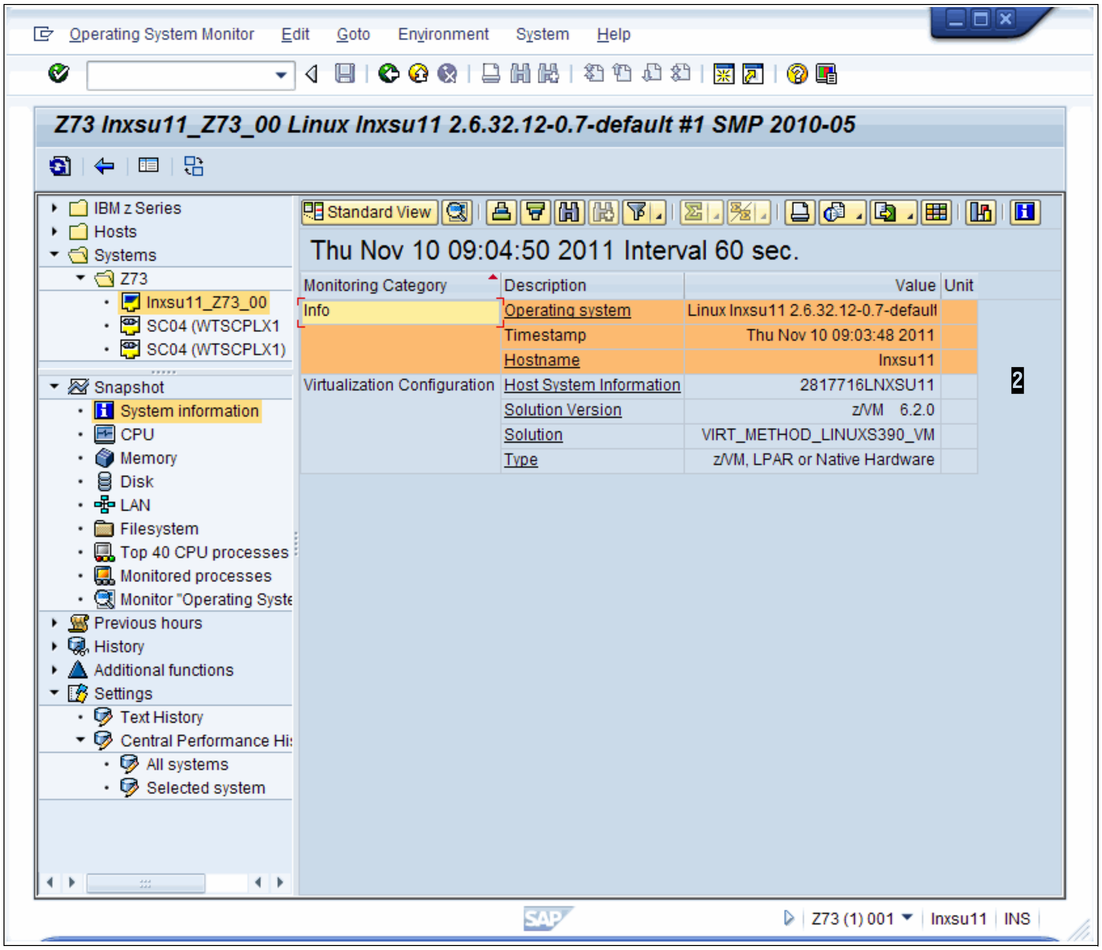


Figure 5-6 Host system information after the relocation

The host systems are shown in the field “Virtualization Configuration - Host System Information”.

A

Frequently asked questions

This appendix shows common conditions that can prevent a guest from being relocated by live guest relocation (LGR). We offer suggestions about how to resolve the issues that can stop the relocation. Additionally, we discuss other problems we saw in our tests.

A.1 Guests do not relocate

The VMRELOCATE TEST command is used to check whether a Linux server is eligible for relocation. Nevertheless, during the VMRELOCATE MOVE command, a problem might happen during the move, such as a channel-to-channel (CTC) connection can be lost or the guest might become ineligible. The VMRELOCATE command failure lists all those reasons.

In the following sections, we describe several reasons why a Linux guest is ineligible for relocation.

A.1.1 Linux with a DEVNO minidisk in use elsewhere

Example A-1 shows the directory definition for a Linux guest, ITSOLNX2, which uses a full-pack minidisk address 300 that is defined as DEVNO rather than specifying the volume serial number and extents.

> **INCLUDE statement:** We commented out the INCLUDE for the CMS definitions, because the CMS definitions stop a Linux guest from relocating.

Example A-1 Linux directory entry with a full-pack minidisk

```
USER ITSOLNX2 ITSOSSI 4G 4G G
*  INCLUDE LINCMS
   INCLUDE LINDFLT
   IPL 300
   MACH ESA 4
   NICDEF 060 Example A-3 0 TYPE QDIO LAN SYSTEM ITSOVSW1
   MDISK 0300 3390 DEVNO 9A26 MR
```

Problem

Example A-2 shows that, in this case, the full-pack DASD was not available at the destination member.

Example A-2 VMRELOCATE TEST command

```
vmrelocate test user itsolnx2 to itsossi1
HCPRLH1940E ITSOLNX2 is not relocatable for the following reason(s):
HCPRLI1981I ITSOLNX2: Source system DEVNO defined minidisk 0300 on real device 9
A26 cannot be created on the destination system because the equivalent real devi
ce 9A26 is neither a DENVO defined minidisk nor free
Ready(01940); T=0.01/0.01 15:15:09
```

Solution

Detaching this address from the system at the destination solved the problem. We used the command **at ... cmd** without needing to log on at the other member, as shown in Example A-3.

Example A-3 Use of the at command

```
q 9a26
DASD 9A26 CP SYSTEM DEVNO    1
Ready; T=0.01/0.01 15:15:52
at itsossi1 cmd q 9a26
```

```
DASD 9A26 CP SYSTEM US9A26   0
Ready; T=0.01/0.01 15:16:13
at itsossi1 cmd det  9a26 system
DASD 9A26  DETACHED SYSTEM
Ready; T=0.01/0.01 15:16:46
at itsossi1 cmd q 9a26
DASD 9A26 US9A26
Ready; T=0.01/0.01 15:16:50
vmrelocate test user itsolnx2 to itsossi1
User ITSOLNX2 is eligible for relocation to ITSOSSI1
Ready; T=0.01/0.01 15:17:04
```

A.1.2 Linux with CMS mdisks defined

In the second test, we cleared the comment on the INCLUDE LINCMS statement and included the mandatory OPTION card. LINCMS defines the CMS disks that are used to IPL CMS (Example A-4).

Example A-4 Linux with the CMS mdisks

```
USER ITSOLNX2 ITSOSSI 4G 4G G
   INCLUDE LINCMS
*  INCLUDE LINDFLT
   IPL 300
   MACH ESA 4
   OPTION CHPIDV ONE
   NICDEF 0600 TYPE QDIO LAN SYSTEM ITSOVSW1
   MDISK 0300 3390 DEVNO 9A26 MR
```

Problem

These CMS mdisks are defined on local non-shared volumes, which causes an error.

Example A-5 Linux with CMS mdisks

```
vmrelocate test user itsolnx2 to itsossi1
HCPRLH1940E ITSOLNX2 is not relocatable for the following reason(s):
HCPRLI1996I ITSOLNX2: Virtual machine device 0190 is a link to a local minidisk
HCPRLI1996I ITSOLNX2: Virtual machine device 019D is a link to a local minidisk
HCPRLI1996I ITSOLNX2: Virtual machine device 019E is a link to a local minidisk
Ready(01940); T=0.01/0.01 15:38:45
```

Solution

The CMS minidisks are necessary only during the installation of Linux. After the installation, the CMS minidisks are not necessary. To solve the problem, we detached these CMS minidisks from the Linux machine dynamically. Removing the INCLUDE cards makes the definitions permanent.

After detaching the CMS minidisks, the VMRELOCATE TEST did not show any errors (Example A-6 on page 90).

The VMRELOCATE MOVE transferred the machine to the ITSOSSI1 member.

Example A-6 Relocating after detaching the CMS minidisks

```
send cp itsolnx2 det 190 19d 19e
Ready; T=0.01/0.01 15:48:18
ITSOLNX2: 0190 019D 019E DETACHED
vmrelocate test user itsolnx2 to itsossi1
User ITSOLNX2 is eligible for relocation to ITSOSSI1
Ready; T=0.01/0.01 15:48:32
vmrelocate move user itsolnx2 to itsossi1
Relocation of ITSOLNX2 from ITSOSSI4 to ITSOSSI1 started
User ITSOLNX2 has been relocated from ITSOSSI4 to ITSOSSI1
ITSOLNX2: User ITSOLNX2 has been relocated from ITSOSSI4 to ITSOSSI1
ITSOLNX2: User ITSOLNX2 has been relocated from ITSOSSI4 to ITSOSSI1
ITSOLNX2: CONNECT= 00:10:37 VIRTCPU= 000:04.92 TOTCPU= 000:05.05
ITSOLNX2: LOGOFF AT 15:48:48 EST MONDAY 11/07/11 BY SYSTEM
USER DSC   LOGOFF AS  ITSOLNX2 USERS = 19    FORCED BY SYSTEM
Ready; T=0.01/0.01 15:48:48
q user itsolnx2 at all
ITSOSSI1 : ITSOLNX2 - DSC
Ready; T=0.01/0.01 15:49:30
close cons
```

LOGOFF notices: We moved Linux and all its applications to ITSOSSI1. Example A-6 shows the LOGOFF notices on ITSOSSI4, because the guest is no longer running on the source member. The control program (CP) cleans up the data structures in the same way as though you logged off the guest.

A.1.3 Using a LINUX guest with V-Disk minidisk definition

V-DISKs are minidisks that are emulated in memory that can be shared between VM machines. Normally, V-DISKs are used in Linux machines as SWAP DASDs.

Problem

Example A-7 shows a V-DISK that is defined as a minidisk to prevent the relocation.

Example A-7 V-DISk minidisk statement

```
type        itsolnx1 direct
USER ITSOLNX1 ITSOSSI 4G 4G G
   INCLUDE LINDFLT
   IPL 300
   MACH ESA 4
   NICDEF 0600 TYPE QDIO LAN SYSTEM ITSOVSW1
   MDISK 0111 FB-512 V-DISK 128000
   MDISK 0300 3390 0001 10016 LXDE1C MR
   MDISK 0400 3390 0001 30050 LX603D MR
Ready; T=0.01/0.01 11:34:11
vmrelocate test itsolnx1 to itsossi3
HCPRLH1940E ITSOLNX1 is not relocatable for the following reason(s):
HCPRLI1970I ITSOLNX1: Virtual machine device 0111 is a shareable VDISK
Ready(01940); T=0.01/0.01 11:34:17
```

Solution

If the V-DISK is used for SWAP only and is not shared, no restrictions exist to use a V-DISK that is defined by the DEFINE command. So, changing from MDISK V-DISK to COMMAND DEFINE VFB-512 allows the relocation. Example A-8 shows the updated directory.

> **Important:** To use DASDs that are emulated in memory effectively, they must be formatted and defined as SWAP every time that Linux is started.

Example A-8 Updated directory statements

```
type        itsolnx1 direct

USER ITSOLNX1 ITSOSSI 4G 4G G
   INCLUDE LINDFLT
   COMMAND DEFINE VFB-512 AS 0111 BLK 128000
   IPL 300
   MACH ESA 4
   NICDEF 0600 TYPE QDIO LAN SYSTEM ITSOVSW1
*  MDISK 0111 FB-512 V-DISK 128000
   MDISK 0300 3390 0001 10016 LXDE1C MR
   MDISK 0400 3390 0001 30050 LX603D MR

Ready; T=0.01/0.01 11:53:50
vmrelocate test itsolnx1 to itsossi3
User ITSOLNX1 is eligible for relocation to ITSOSSI3
Ready; T=0.01/0.01 11:53:58
```

A.1.4 Moving a CMS guest

Relocating a CMS guest is not supported by LGR. However, you can take advantage of the VM single system image (VMSSI) features and the ability for the CMS guest to log on to any member of an SSI cluster.

Problem

Example A-9 shows many restrictions, which cannot be solved by commands, on this guest. All minidisks are local and cannot be detached.

Example A-9 A CMS guest

```
USER VMLOGS ITSOSSI 64M 128M ABDEG
*
* Keep CONSOLE/SPOOL files on DASD for some time (90 days)
*
   INCLUDE IBMDFLT
   IPL CMS PARM AUTOCR
   MACH ESA
   MDISK 0191 3390 2902 1000 SSIAC1 MR LABEL CMS191
   MDISK 0193 3390 3902 0005 SSIAC1 MR LABEL CMS193
```

Solution

We cannot use LGR to relocate a CMS guest, but we still can use VMSSI functions (see Example A-10 on page 92). The objective of this machine is to collect and save spool files. We can stop it in one member (FORCE at ITSOSSI4) and start it again on another member

(XAUTOLOG at ITSOSSI1). The cross-system spool functions help you avoid losing a file. See the commands in Example A-10.

Linux guests: Use this same approach with Linux guests that cannot use the relocate function due to size conflicts, application characteristics, and so on. If it is a Linux guest, you must use the SIGNAL SHUTDOWN command instead of the FORCE command.

Example A-10 Manual relocation of a CMS guest

```
vmrelocate test user vmlogs to itsossi2
HCPRLH1940E VMLOGS is not relocatable for the following reason(s):
HCPRLE1956I VMLOGS: Single path CHPID virtualization is not enabled
HCPRLE1963I VMLOGS: Virtual machine is using VMCF
HCPRLI1996I VMLOGS: Virtual machine device 0190 is a link to a local minidisk
HCPRLI1996I VMLOGS: Virtual machine device 019D is a link to a local minidisk
HCPRLI1996I VMLOGS: Virtual machine device 019E is a link to a local minidisk
HCPRLI1996I VMLOGS: Virtual machine device 0401 is a link to a local minidisk
HCPRLI1996I VMLOGS: Virtual machine device 0402 is a link to a local minidisk
HCPRLI1996I VMLOGS: Virtual machine device 1193 is a link to a local minidisk
HCPRLL1980I VMLOGS: An identical NSS or DCSS CMS does not exist on the destinati
on system
at itsossi4 cmd force vmlogs
VMLOGS  : CONNECT= 99:59:59 VIRTCPU= 000:00.85 TOTCPU= 000:02.85
VMLOGS  : LOGOFF AT 16:24:44 EST MONDAY 11/07/11 BY MAINT620
USER DSC   LOGOFF AS  VMLOGS   USERS = 18    FORCED BY MAINT620
Ready; T=0.01/0.01 16:24:44
VMLOGS  : RDR FILE 0368 SENT FROM VMLOGS   CON WAS 0368 RECS 0439 CPY  001 T NOH
OLD NOKEEP
at itsossi1 cmd xautolog vmlogs
Command accepted
Ready; T=0.01/0.01 16:25:00
query user  vmlogs at all
ITSOSSI1 : VMLOGS   - DSC
Ready; T=0.01/0.01 16:25:07
close cons
CON FILE 0216 SENT TO   VMLOGS   RDR AS  0408 RECS 0150 CPY  001 T NOHOLD NOKEEP
Ready; T=0.01/0.01 16:27:45

 16:27:45  * MSG FROM VMLOGS  : File MAINT CONSOLE received with success.
```

A.1.5 Using DEDICATE devices

One problem that can prevent a relocation is the use of DEDICATE devices, such as when equivalency identifiers (EQIDs) are not set correctly or the equivalent device is not available on the destination member.

VSWITCH: VSWITCH is optimized to work in an SSI cluster. VSWITCH is the preferred way to work with a network. Avoid using a dedicated Open Systems Adapter (OSA) where possible.

We used one OSA as an example to simulate the problem. Because all OSAs in our environment already are associated with EQIDs, we deleted the EQID from one triplet.

To change the characteristics of any real device, it must be offline. Example A-11 shows the commands that we used to delete the EQID.

Example A-11 Deleting EQIDs

```
q osa
OSA  2120 ATTACHED TO TCPIP    2120 DEVTYPE OSA        CHPID 0C OSD
OSA  2121 ATTACHED TO TCPIP    2121 DEVTYPE OSA        CHPID 0C OSD
OSA  2122 ATTACHED TO TCPIP    2122 DEVTYPE OSA        CHPID 0C OSD
OSA  2123 ATTACHED TO DTCVSW1  0600 DEVTYPE OSA        CHPID 0C OSD
OSA  2124 ATTACHED TO DTCVSW1  0601 DEVTYPE OSA        CHPID 0C OSD
OSA  2125 ATTACHED TO DTCVSW1  0602 DEVTYPE OSA        CHPID 0C OSD
Ready; T=0.01/0.01 11:08:11
q 2126-2128
OSA  2126 FREE    , OSA  2127 FREE    , OSA  2128 FREE
Ready; T=0.01/0.01 11:08:29
vary off 2126-2128
2126 varied offline
2127 varied offline
2128 varied offline
3 device(s) specified; 3 device(s) successfully varied offline
Ready; T=0.01/0.01 11:08:45
set rdev 2126-2128 noeqid type osa
HCPZRP6722I Characteristics of device 2126 were set as requested.
HCPZRP6722I Characteristics of device 2127 were set as requested.
HCPZRP6722I Characteristics of device 2128 were set as requested.
3 RDEV(s) specified; 3 RDEV(s) changed; 0 RDEV(s) created
Ready; T=0.01/0.01 11:09:20
vary on  2126-2128
2126 varied online
2127 varied online
2128 varied online
3 device(s) specified; 3 device(s) successfully varied online
Ready; T=0.01/0.01 11:09:31
```

Next, we simulated the problem.

Problem

Example A-12 shows the amended directory entry for the Linux guest with the NICDEF commented out and the respective DEDICATE commands added.

Example A-12 Amended Linux directory entry

```
type itsolnx2 direct

USER ITSOLNX2 ITSOSSI 4G 4G G                                  11071537
*  INCLUDE LINCMS                                              11071537
   INCLUDE LINDFLT                                             11071537
   COMMAND DEFINE VFB-512 AS 0111 BLK 128000
   IPL 300                                                     11071537
   MACH ESA 4                                                  11071537
   DEDICATE 0600 2126
   DEDICATE 0601 2127
   DEDICATE 0602 2128
*  NICDEF 0600 TYPE QDIO LAN SYSTEM ITSOVSW1                   11071537
```

```
   MDISK 0300 3390 DEVNO 9A26 MR                                              11071537
Ready; T=0.01/0.01 11:24:58
```

Example A-13 shows that the Linux guest was started, that the OSA was checked, and that we tried the relocation.

Example A-13 Dedicate devices without EQID

```
xautolog itsolnx2
Command accepted
Ready; T=0.01/0.01 11:14:19
AUTO LOGON  ***       ITSOLNX2 USERS = 19
HCPCLS6056I XAUTOLOG information for ITSOLNX2: The IPL command is verified by the
IPL command processor.
q osa
OSA  2120 ATTACHED TO TCPIP    2120 DEVTYPE OSA        CHPID 0C OSD
OSA  2121 ATTACHED TO TCPIP    2121 DEVTYPE OSA        CHPID 0C OSD
OSA  2122 ATTACHED TO TCPIP    2122 DEVTYPE OSA        CHPID 0C OSD
OSA  2123 ATTACHED TO DTCVSW1  0600 DEVTYPE OSA        CHPID 0C OSD
OSA  2124 ATTACHED TO DTCVSW1  0601 DEVTYPE OSA        CHPID 0C OSD
OSA  2125 ATTACHED TO DTCVSW1  0602 DEVTYPE OSA        CHPID 0C OSD
OSA  2126 ATTACHED TO ITSOLNX2 0600 DEVTYPE OSA        CHPID 0C OSD
OSA  2127 ATTACHED TO ITSOLNX2 0601 DEVTYPE OSA        CHPID 0C OSD
OSA  2128 ATTACHED TO ITSOLNX2 0602 DEVTYPE OSA        CHPID 0C OSD
Ready; T=0.01/0.01 11:14:24
vmrelocate test itsolnx2 to itsossi3
HCPRLH1940E ITSOLNX2 is not relocatable for the following reason(s):
HCPRLI1982I ITSOLNX2: Source system real network device 2126 requires an EQID to
be assigned
HCPRLI1982I ITSOLNX2: Source system real network device 2127 requires an EQID to
be assigned
HCPRLI1982I ITSOLNX2: Source system real network device 2128 requires an EQID to
be assigned
Ready(01940); T=0.01/0.01 11:15:47
```

Solution

Note: Although we used an OSA as an example, this behavior is the same for OSA, Hipersockets, or Small Computer System Interface (SCSI) Fibre Channel Protocol (FCP) devices. All of these devices must be associated with a common EQID.

The solution in this case is to associate the devices with a valid EQID or to use another device that already has a valid EQID. Example A-14 on page 95 shows how to define an EQID dynamically. The correct definition must be in the SYSTEM CONFIG file to prevent problems after the next z/VM IPL. In our environment, we defined the EQID during the installation.

After the redefinition of the EQID, the Linux guest was xautologged in member ITSOSSI4 and relocated to member ITSOSSI3. The directory entry was not changed. The DEDICATE statements are now valid for ITSOSSI4, but not for ITSOSSI3, which has separate OSA addresses.

z/VM identified that the OSAs use the same EQID and made the correct attachments (2126 in ITSOSSI4 was changed to 2106 in ITSOSSI3) automatically. See the last QUERY command, which is the way that VMSSI works with EQIDs.

Important: This automatic translation was performed by the VMRELOCATE command. It is not valid for the IPL command. To avoid problems during the IPL, correct the directory definition to use NICDEF or COMMAND ATTACH.

Example A-14 Defining EQID dynamically

```
signal shutdown itsolnx2
Ready; T=0.01/0.01 11:21:34
HCPSIG2113I User ITSOLNX2 has reported successful termination
USER DSC   LOGOFF AS  ITSOLNX2 USERS = 18    AFTER SIGNAL
vary off 2126-2128
2126 varied offline
2127 varied offline
2128 varied offline
3 device(s) specified; 3 device(s) successfully varied offline
Ready; T=0.01/0.01 11:21:53
set rdev 2126-2128 eqid osa1set1 type osa
HCPZRP6722I Characteristics of device 2126 were set as requested.
HCPZRP6722I Characteristics of device 2127 were set as requested.
HCPZRP6722I Characteristics of device 2128 were set as requested.
3 RDEV(s) specified; 3 RDEV(s) changed; 0 RDEV(s) created
Ready; T=0.01/0.01 11:22:27
vary on 2126-2128
2126 varied online
2127 varied online
2128 varied online
3 device(s) specified; 3 device(s) successfully varied online
Ready; T=0.01/0.01 11:22:38
xautolog itsolnx2
Command accepted
Ready; T=0.01/0.01 11:23:01
AUTO LOGON  ***       ITSOLNX2 USERS = 19
HCPCLS6056I XAUTOLOG information for ITSOLNX2: The IPL command is verified by the
IPL command processor.
q osa
OSA  2120 ATTACHED TO TCPIP    2120 DEVTYPE OSA         CHPID 0C OSD
OSA  2121 ATTACHED TO TCPIP    2121 DEVTYPE OSA         CHPID 0C OSD
OSA  2122 ATTACHED TO TCPIP    2122 DEVTYPE OSA         CHPID 0C OSD
OSA  2123 ATTACHED TO DTCVSW1  0600 DEVTYPE OSA         CHPID 0C OSD
OSA  2124 ATTACHED TO DTCVSW1  0601 DEVTYPE OSA         CHPID 0C OSD
OSA  2125 ATTACHED TO DTCVSW1  0602 DEVTYPE OSA         CHPID 0C OSD
OSA  2126 ATTACHED TO ITSOLNX2 0600 DEVTYPE OSA         CHPID 0C OSD
OSA  2127 ATTACHED TO ITSOLNX2 0601 DEVTYPE OSA         CHPID 0C OSD
OSA  2128 ATTACHED TO ITSOLNX2 0602 DEVTYPE OSA         CHPID 0C OSD
Ready; T=0.01/0.01 11:23:22
ping 9.12.4.140
Ping Level 620: Pinging host 9.12.4.140.
                Enter #CP EXT to interrupt.
PING: Ping #1 response took 0.000 seconds. Successes so far 1.
Ready; T=0.01/0.01 11:23:44
vmrelocate test itsolnx2 to itsossi3
User ITSOLNX2 is eligible for relocation to ITSOSSI3
Ready; T=0.01/0.01 11:23:52
vmrelocate move itsolnx2 to itsossi3
```

```
Relocation of ITSOLNX2 from ITSOSSI4 to ITSOSSI3 started
User ITSOLNX2 has been relocated from ITSOSSI4 to ITSOSSI3
USER DSC   LOGOFF AS  ITSOLNX2 USERS = 18    FORCED BY SYSTEM
Ready; T=0.01/0.01 11:24:12
type itsolnx2 direct

USER ITSOLNX2 ITSOSSI 4G 4G G                                       11071537
*  INCLUDE LINCMS                                                   11071537
   INCLUDE LINDFLT                                                  11071537
   COMMAND DEFINE VFB-512 AS 0111 BLK 128000
   IPL 300                                                          11071537
   MACH ESA 4                                                       11071537
   DEDICATE 0600 2126
   DEDICATE 0601 2127
   DEDICATE 0602 2128
*  NICDEF 0600 TYPE QDIO LAN SYSTEM ITSOVSW1                        11071537
   MDISK 0300 3390 DEVNO 9A26 MR                                    11071537
*DVHOPT LNK0 LOG1 RCM1 SMS0 NPW1 LNGAMENG PWC20111103 CRC:         11080002

Ready; T=0.01/0.01 11:24:58
at itsossi3 cmd q osa
OSA  2100 ATTACHED TO TCPIP    2100 DEVTYPE OSA         CHPID 00 OSD
OSA  2101 ATTACHED TO TCPIP    2101 DEVTYPE OSA         CHPID 00 OSD
OSA  2102 ATTACHED TO TCPIP    2102 DEVTYPE OSA         CHPID 00 OSD
OSA  2103 ATTACHED TO DTCVSW2  0600 DEVTYPE OSA         CHPID 00 OSD
OSA  2104 ATTACHED TO DTCVSW2  0601 DEVTYPE OSA         CHPID 00 OSD
OSA  2105 ATTACHED TO DTCVSW2  0602 DEVTYPE OSA         CHPID 00 OSD
OSA  2106 ATTACHED TO ITSOLNX2 0600 DEVTYPE OSA         CHPID 00 OSD
OSA  2107 ATTACHED TO ITSOLNX2 0601 DEVTYPE OSA         CHPID 00 OSD
OSA  2108 ATTACHED TO ITSOLNX2 0602 DEVTYPE OSA         CHPID 00 OSD
Ready; T=0.01/0.01 11:25:11
at itsossi4 cmd q osa
OSA  2120 ATTACHED TO TCPIP    2120 DEVTYPE OSA         CHPID 0C OSD
OSA  2121 ATTACHED TO TCPIP    2121 DEVTYPE OSA         CHPID 0C OSD
OSA  2122 ATTACHED TO TCPIP    2122 DEVTYPE OSA         CHPID 0C OSD
OSA  2123 ATTACHED TO DTCVSW1  0600 DEVTYPE OSA         CHPID 0C OSD
OSA  2124 ATTACHED TO DTCVSW1  0601 DEVTYPE OSA         CHPID 0C OSD
OSA  2125 ATTACHED TO DTCVSW1  0602 DEVTYPE OSA         CHPID 0C OSD
Ready; T=0.01/0.01 11:25:30
at itsossi4 cmd q 2126 id
OSA  2126 1732-01 CU: 1731-01
     2126 EQID: OSA1SET1
Ready; T=0.01/0.01 11:25:42
at itsossi3 cmd q 2106 id
OSA  2106 1732-01 CU: 1731-01
     2106 EQID: OSA1SET1
Ready; T=0.01/0.01 11:26:00
ping 9.12.4.140
Ping Level 620: Pinging host 9.12.4.140.
                Enter #CP EXT to interrupt.
PING: Ping #1 response took 0.001 seconds. Successes so far 1.
Ready; T=0.01/0.01 11:27:17
close cons
```

A.1.6 Live guest relocation of zLinux guests outside domains

We modified LNXSU12 so that it only can be relocated in ITSOSSI5. This example is atypical. We show it only to demonstrate several of the responses that we received. Example A-15 shows the commands that we used to change the relocation domain for the Linux guest lnxsu12.

Example A-15 Modifying the relocation domain of LNXSU12

```
q vmrelo lnxsu12
10:32:25 Running on member ITSOSSI5
10:32:25 Relocation enabled in Domain SSI
Ready; T=0.01/0.01 10:32:25
set vmrelo user lnxsu12 domain itsossi5
10:33:05 Running on member ITSOSSI5
10:33:05 Relocation enabled in Domain ITSOSSI5
Ready; T=0.01/0.01 10:33:05
```

We then attempted to relocate LNXSU12 to ITSOSSI6. The relocation attempt failed, as we expected. See Example A-16.

Example A-16 Attempt to relocate LNXSU12 to a system on which it is not eligible to run

```
vmrelocate test lnxsu12 itsossi6
10:33:30 HCPRLH1940E LNXSU12 is not relocatable for the following reason(s):
10:33:30 HCPRLE1944I LNXSU12: Architecture incompatibility
Ready(01940); T=0.01/0.01 10:33:30
vmrelocate move lnxsu12 itsossi6
10:33:41 Relocation of user LNXSU12 from ITSOSSI5 to ITSOSSI6 did not complete.
Guest has not been moved
10:33:42 HCPRLH1940E LNXSU12 is not relocatable for the following reason(s):
10:33:42 HCPRLE1944I LNXSU12: Architecture incompatibility
Ready(01940); T=0.01/0.01 10:33:42
```

Problem

LNXSU12 was defined to run on ITSOSSI5 only. The responses to the relocation attempt indicate that it failed due to "`Architecture incompatibility`".

Example A-17 shows that we attempted to use a relocation command with the FORCE ARCHITECTURE operand.

Important: Be careful when using the FORCE option with the VMRELOCATE command. Depending on which resource is forced, the result can be unpredictable, including a guest abend.

Example A-17 VMRELOCATE with FORCE ARCHITECTURE

```
vmrelocate move lnxsu12 itsossi6 force architecture
10:39:46 Relocation of user LNXSU12 from ITSOSSI5 to ITSOSSI6 did not complete.
Guest has not been moved
10:39:47 HCPRLH1940E LNXSU12 is not relocatable for the following reason(s):
10:39:47 HCPRLE1944I LNXSU12: Architecture incompatibility
Ready(01940); T=0.01/0.01 10:39:47
```

Again, the relocation failed for the same reason.

Solution

We know that LNXSU12 failed to relocate, because its domain is set so that it can run on ITSOSSI5 only. Therefore, we ran the relocate command with the FORCE DOMAIN operand, as shown in Example A-18.

Example A-18 VMRELOCATE with FORCE DOMAIN operand

```
vmrelocate move lnxsu12 itsossi6 force domain
10:39:57 Relocation of LNXSU12 from ITSOSSI5 to ITSOSSI6 started with FORCE DOMA
IN in effect
10:39:58 User LNXSU12 has been relocated from ITSOSSI5 to ITSOSSI6

Ready; T=0.01/0.01 10:39:58
```

In Example A-18, the relocation of LNXSU12 from ITSOSSI5 to ITSOSSI6 is successful. The FORCE DOMAIN option succeeds only if the architecture features of ITSOSSI6 are a super-set of the features of ITSOSSI5. If they are not a super-set, the option FORCE ARCHITECTURE DOMAIN is required.

A.1.7 Relocation exceeds MAXQUIESCE time

Certain applications allow a defined amount of time in which the Linux guest can be stopped during the relocation. In these cases, you execute the VMRELOCATE command with the MAXQUIESCE option. In Example A-19, we defined a MAXQUIESCE time of 3 seconds.

Example A-19 Call VMRELOCATE with MAXQUIESCE option

```
vmrelocate move lnxsu11 itsossi6 maxquiesce 3
```

Problem

The **vmrelocate** command canceled the relocation, because the maxquiesce time is exceeded. See Example A-20. The Linux guest continues to run on the source system.

Example A-20 MAXQUIESCE time is exceeded

```
vmrelocate move lnxsu11 itsossi6 maxquiesce 3
Relocation of LNXSU11 from ITSOSSI5 to ITSOSSI6 started
Relocation of user LNXSU11 from ITSOSSI5 to ITSOSSI6 did not complete. Guest has
 not been moved
HCPRLH1939E Relocation of LNXSU11 to ITSOSSI6 is terminated for the following re
ason(s):
HCPRLU1930I LNXSU11: The maximum quiesce time was exceeded
Ready(01939); T=0.01/0.01 14:33:00
```

Solution

There are several possible solutions to this time problem, but you need to analyze the environment in more depth. Consider these ideas:

- Determine whether the application allows a longer quiesce time? (Simplest solution)
- Do any non-relocation activities make the relocation and quiesce time longer?
- Are parallel relocations active that increase the time?
- Are any changes in the Inter-System Facility for Communications (ISFC) setup necessary, such as faster CTC speeds or a larger number of CTCs?

A.1.8 Relocation exceeds available auxiliary paging space

We modified LNXSU12 so that its maximum memory size is set to 10 GB of storage. Example A-21 shows the directory entry for LNXSU12.

Example A-21 Directory entry for LNXSU12

```
USER LNXSU12 G4D703JO 10G 10G G
....
```

Problem

When we attempt to relocate LNXSU12 to ITSOSSI6, we get the messages that are shown in Example A-22.

Example A-22 Relocation attempt exceeds the maximum paging space destination

```
vmrelocate test lnxsu12 itsossi6
HCPRLH1940E LNXSU12 is not relocatable for the following reason(s):
HCPRLL1813I LNXSU12: Maximum pageable storage use (10320M) exceeds available aux
iliary paging space on destination (7211520K) by 3356160K
Ready(01940); T=0.01/0.01 10:54:09

vmrelocate move lnxsu12 itsossi6
Relocation of user LNXSU12 from ITSOSSI5 to ITSOSSI6 did not complete. Guest has
 not been moved
HCPRLH1940E LNXSU12 is not relocatable for the following reason(s):
HCPRLL1813I LNXSU12: Maximum pageable storage use (10320M) exceeds available aux
iliary paging space on destination (7211520K) by 3356160K
Ready(01940); T=0.01/0.01 10:55:32
```

Correction: The numbers in the HCPRLL1813I message, which are expressed in K bytes, for example, 7211520K, are only bytes. Development is correcting this message.

The message states that the maximum pageable storage use exceeds the available auxiliary paging space on the destination member. In our case, we only defined one paging disk, which is insufficient for 10-GB maximum guest storage. Example A-23 shows the paging space that is allocated on ITSOSSI6.

Example A-23 Actual paging space on target member ITSOSSI6

```
q alloc page
                EXTENT     EXTENT  TOTAL  PAGES   HIGH    %
VOLID  RDEV      START        END  PAGES IN USE   PAGE USED
------ ---- ---------- ---------- ------ ------ ------ ----
SSI6P1 9E2F          1      10016  1761K      0      0   0%
                                  ------ ------        ----
SUMMARY                            1761K      0          0%
USABLE                             1761K      0          0%
Ready; T=0.01/0.01 10:56:20
```

Solution

We have two possible solutions to reactivate the relocation:

- Execute the relocation with the FORCE STORAGE option, if you are sure that you have enough main storage on the target member. The paging space is not used. Example A-24 shows the VMRELOCATE command with the FORCE STORAGE option specified.

> **Important:** Be careful using the FORCE STORAGE option with the VMRELOCATE command. The result can be unpredictable, including a guest abend or a z/VM abend if the CP abended due to running out of paging space.

Example A-24 Relocate guest with FORCE STORAGE option

```
vmrelocate move lnxsu12 itsossi6 force storage
Relocation of LNXSU12 from ITSOSSI5 to ITSOSSI6 started with FORCE STORAGE in
effect
User LNXSU12 has been relocated from ITSOSSI5 to ITSOSSI6
Ready; T=0.01/0.01 11:00:42
```

- Add DASD for paging space. See Example A-25.

Example A-25 Increased paging space on destination member

```
q alloc page
                EXTENT     EXTENT  TOTAL  PAGES   HIGH    %
VOLID  RDEV      START        END  PAGES IN USE   PAGE USED
------ ---- ---------- ---------- ------ ------ ------ ----
SSI6P2 9C21          1      10016  1761K      0      0   0%
SSI6P1 9E2F          1      10016  1761K      2     15   1%
                                  ------ ------        ----
SUMMARY                            3521K      2          1%
USABLE                             3521K      2          1%
Ready; T=0.01/0.01 11:56:08
```

The check for the eligibility of the guest to relocate no longer shows any messages. See Example A-26.

Example A-26 Check for eligibility ends successfully

```
vmrelocate test lnxsu12 itsossi6
User LNXSU12 is eligible for relocation to ITSOSSI6
Ready; T=0.01/0.01 11:58:26

vmrelocate move lnxsu12 itsossi6
Relocation of LNXSU12 from ITSOSSI5 to ITSOSSI6 started
User LNXSU12 has been relocated from ITSOSSI5 to ITSOSSI6
Ready; T=0.01/0.01 11:59:12
```

A.1.9 Relocation exceeds available capacity of storage

We modified LNXSU12 so that its maximum memory size is set to 20-GB storage. See Example A-27 on page 100.

Example A-27 Directory entry for LNXSU12

```
USER LNXSU12 G4D703JO 20G 20G G
```

....

Problem

When we attempt to relocate LNXSU12 to ITSOSSI6, an additional message displays. This message is much more critical than the message in the scenario that is described in A.1.8, "Relocation exceeds available auxiliary paging space" on page 99. Now, the maximum storage can exceed the available capacity on the destination member. See Example A-28.

Example A-28 Attempt to relocate LNXSU12 exceeds the available capacity on the destination system

```
vmrelocate test lnxsu12 itsossi6
HCPRLH1940E LNXSU12 is not relocatable for the following reason(s):
HCPRLL1811I LNXSU12: Maximum storage use (21136204K) exceeds available capacity
on destination (19513900K) by 1622304K
HCPRLL1813I LNXSU12: Maximum pageable storage use (20640M) exceeds available aux
iliary paging space on destination (14423028K) by 6712613K
Ready(01940); T=0.01/0.01 12:01:39
```

Solution

When you get this message, consider the following possible solutions to fix the storage bottleneck on your destination system:

- Check whether it is possible to increase the main storage of your system.
- Add paging space to increase the storage capacity of the system.
- Execute the relocation with the FORCE STORAGE option

> **Important:** Be careful that you do not reach the capacity limit of your system. The FORCE STORAGE option with the VMRELOCATE command can lead to unpredictable results, including a guest abend or z/VM abend.

A.2 LPAR names are the same on separate CECs in the cluster

The system identifiers that are used on the SSI statement must be unique for each member. So, if your logical partition (LPAR) names are the same, you must use the machine type and serial number for the system identifier. For example, in our environment for the cluster ITSOSSIA, the installation process defined the SYSTEM CONFIG with the statements that are shown in Example A-29.

Example A-29 System identifier that is used by the installation process

```
/**********************************************************************/
/*                    System_Identifier Information                   */
/**********************************************************************/

     System_Identifier LPAR A1A ITSOSSI1
     System_Identifier LPAR A1B ITSOSSI2
     System_Identifier LPAR A2A ITSOSSI3
     System_Identifier LPAR A2B ITSOSSI4
```

To prevent problems with other LPARs with the same name, the definition that is shown in Example A-30 is better.

Example A-30 System_Identifiers changed to serial numbers

```
/**********************************************************************/
/*                   System_Identifier Information                    */
/**********************************************************************/

     System_Identifier 2097 1ADE50 ITSOSSI1
     System_Identifier 2097 1BDE50 ITSOSSI2
     System_Identifier 2817 2A3BD5 ITSOSSI3
     System_Identifier 2817 2B3BD5 ITSOSSI4
```

For more details, see *z/VM V6R2.0 CP Planning and Administration*, SC24-6178-02.

A.3 Incorrect system name

On the Installation Planning window, in the system name field, we entered the name of the final SSI cluster ITSOSSIB instead of the name of the stand-alone z/VM system, ITSOSSI5. After we installed z/VM, we decided to correct this name and amended the system name in the SYSTEM CONFIG file on the PMAINT CF0 disk. We did not change the ownership value on each of the CP-owned volumes. Because did not change the ownership value on each of the CP-owned volumes, we were unable to IPL z/VM. z/VM was unable to locate the spool or paging volumes, and we lost all the contents of the spool, including the saved segments.

We attached the volumes to another z/VM system and used the CPFMTXA command to change the following ownership values:

```
CPFMTXA 9D20 SSIBC1 OWNER NOSSI ITSOSSI5
CPFMTXA 9D21 SSIBR1 OWNER NOSSI ITSOSSI5
CPFMTXA 9D22 SSI5RS OWNER NOSSI ITSOSSI5
CPFMTXA 9D23 SSI5S1 OWNER NOSSI ITSOSSI5
CPFMTXA 9D24 SSI5P1 OWNER NOSSI ITSOSSI5
```

We then performed the IPL on z/VM and recovered the saved segments by logging on to MAINT620 and issuing the following command:

```
PUT2PROD SEGMENTS ALL
```

A.4 Recovery messages on the Linux console during relocation

In this section, we describe the messages on the Linux console as it recovers its OSA devices. You can see the messages by spooling this console or by using the SET OBSERVER command, as shown in Example A-31.

Example A-31 Observer command for Linux Guest ITSOLNX2

```
set observer itsolnx2 *
Ready; T=0.01/0.01 15:47:48
```

Example A-32 shows a sample of several of the messages that are displayed about the new member after the relocation request completed.

Example A-32 Linux messages

```
vmrelocate move itsolnx2 itsossi3
Relocation of ITSOLNX2 from ITSOSSI4 to ITSOSSI3 started
User ITSOLNX2 has been relocated from ITSOSSI4 to ITSOSSI3
ITSOLNX2: User ITSOLNX2 has been relocated from ITSOSSI4 to ITSOSSI3
ITSOLNX2: User ITSOLNX2 has been relocated from ITSOSSI4 to ITSOSSI3
ITSOLNX2: qeth.3acf0c: 0.0.0600: The qeth device driver failed to recover an err
or on the device
qeth: irb 00000000: 00 c2 40 17 7f f6 10 38 0e 02 00 00 00 80 00 00  ..@....8...
.....
qeth: irb 00000010: 01 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ..........
.....
qeth: sense data 00000000: 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ....
............
qeth: sense data 00000010: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ....
............
.
.

.....
qeth.fd0b7c: 0.0.0600: A recovery process has been started for the device
qdio: 0.0.0602 OSA on SC 2 using AI:1 QEBSM:0 PCI:1 TDD:1 SIGA:RW AO
qeth.736dae:
ITSOLNX2: CONNECT= 02:18:08 VIRTCPU= 000:04.33 TOTCPU= 000:04.61
ITSOLNX2: LOGOFF AT 13:57:00 EST THURSDAY 11/17/11 BY SYSTEM
USER DSC   LOGOFF AS  ITSOLNX2 USERS = 17    FORCED BY SYSTEM
Ready;                           T=0.01/0.01                          13:57:00
```

These messages are not error messages and can be ignored. When a Linux guest relocates to a new member, it must reestablish its network connection from the new member. These messages relate to reconnecting the new member.

B

New and updated commands

In this appendix, we describe several new or updated commands that we encountered during this project. To access the complete set of commands that are added to or changed in this version of z/VM, see the related component manual. We list the major sources of information in “Related publications” on page 115.

B.1 Single system image and live guest relocation commands

Table B-1 contains details about the control program (CP) commands for single system image (SSI) and live guest relocation (LGR).

Table B-1 CP commands

Command	Function
QUERY SSI	This command displays the single system image (SSI) name, member status, and connectivity status.
AT -sysname- CMD	Use the AT command to issue commands remotely on active member systems in an SSI cluster.
QUERY -rdev- ID	This command displays the device and control unit information from the sense ID data for a specified device address, if they are known. It also displays the device equivalency ID (EQID) if an EQID exists for the device.
SET -rdev- <*NoEQID* | *EQID eqid*> TYPE -type-	The EQID *eqid* assigns the device eqivalency ID (EQID) to the RDEV. The *eqid* is a string of 1 - 8 alphanumeric characters. For channel-to-channel attachment (CTCA), Fibre Channel Protocol (FCP), HiperSocket, and Open Systems Adapter (OSA) devices, this EQID must be unique or shared only by other devices of the same type. NOEQid removes a previously assigned EQID from this RDEV and reverts to a system-generated EQID. If no EQID was assigned by a user earlier, no action takes place.
VMRELOCATE	This command moves an eligible, running z/VM virtual machine transparently from one z/VM system to another within an SSI cluster. This command monitors and cancels virtual machine relocations that are in progress already.
DEFINE RELODOMAIN	This command defines or updates an SSI relocation domain.
QUERY RELODOMAIN	This command lists the members of one or more relocation domains.
SET VMRELOCATE	This command dynamically controls the relocation domain for a user.
QUERY user AT ALL	This command is updated to display users that are logged on to other cluster members.
QUERY NAMES	This command is updated to show users that are logged on elsewhere in the SSI cluster.
SET SSI	This command adds or changes existing entries in the SSI member list.

Table B-2 lists the z/VM utilities.

Table B-2 z/VM utilities

Command	Function
FORMSSI	This command creates, updates, unlocks, or displays a persistent data record (PDR) on cylinder zero of a disk.
CPFMTXA	The option `Owner` indicates that you want to specify or modify the ownership information for a CP-formatted device.

B.2 DIRMAINT commands

Table B-3 shows several of the updated DIRM commands.

Table B-3 DIRM commands

Command	Function
DIRM SSI	The SSI operand prepares a source directory for use on a node in an SSI cluster.
DIRM UNDOSSI	The UNDOSSI operand rolls back the BUILD statement changes that are made by the SSI operand and removes the SSI operand from the DIRECTORY statement.
DIRM VMRELOCATE	The VMRELOCATE operand queries, updates, or deletes the relocation capability that is associated with a user or profile entry.
DIRM ADD subconfig BUILD ON member IN identity	The ADD operand is updated for cloning SUBCONFIG entries. We describe the cloning capability more fully in B.2.1, “Adding identities”.

Due to the new structure of the USER DIRECT file, the IDENTITY and SUBCONFIG are treated separately. Each IDENTITY that is defined also has a SUBCONFIG definition for each member of the SSI cluster. A number of new DIRM options to process IDENTITY statements are available.

B.2.1 Adding identities

Use the following actions to create an identity named, for example, SRI:

- Create a file that contains the IDENTITY statements, including all of the definitions of all the common and shared resources, for example, SRI DIRECT A.

  Example B-1 contains an example of the IDENTITY statements.

  Example B-1 SRI DIRECT A file

```
IDENTITY SRI ******** 16M 16M G
   ACCOUNT SYSTEMS
   IPL CMS
   MACH ESA
   CONSOLE 0009 3215
   SPOOL 000C 2540 READER *
   SPOOL 000D 2540 PUNCH A
   SPOOL 000E 1403 A
   LINK MAINT 0190 0190 RR
   LINK MAINT 019E 019E RR
   LINK MAINT 019D 019D RR
```

- Create a file for each of the SUBCONFIG statements. Define the resources that are unique to each member. Include the minidisk statement that relates to that member, for example, SRI-*n* DIRECT A.

  Example B-2 on page 107 contains an example of the SUBCONFIG statements.

  Example B-2 SRI-1 DIRECT A

```
SUBCONFIG SRI-1
   MDISK 0191 3390 AUTOV 15 SSI1M1 MR
```

- Use the DIRM ADD command to add the IDENTITY statement file, for example, DIRM ADD SRI DIRECT A.
- Use DIRM ADD identity-1 BUILD ON member IN identity, for example, DIRM ADD SRI-*n* BUILD ON ITSOSSI*n* in SRI.

  Example B-3 shows the directory entry for the IDENTITY after the DIRM ADD BUILD command executes. The DIRM ADD BUILD command automatically updates the IDENTITY with the BUILD statements.

Example B-3 SRI DIRECT A after the DIRM ADD BUILD command executes

```
DENTITY SRI ******** 16M 16M G
   BUILD ON ITSOSSI1 USING SUBCONFIG SRI-1
   BUILD ON ITSOSSI2 USING SUBCONFIG SRI-2
   ACCOUNT SYSTEMS
   IPL CMS
   MACH ESA
   CONSOLE 0009 3215
   SPOOL 000C 2540 READER *
   SPOOL 000D 2540 PUNCH A
   SPOOL 000E 1403 A
   LINK MAINT 0190 0190 RR
   LINK MAINT 019E 019E RR
   LINK MAINT 019D 019D RR
```

- Example B-4 shows the output from the DIRM FOR user REVIEW. The build statements are replaced in the output by the SUBCONFIG entries.

Example B-4 Output from DIRM FOR SRI REVIEW command

```
IDENTITY SRI ******** 16M 16M G
DVHRXV3366I The following configurations will be used on SSI nodes.
DVHRXV3366I The following configuration SRI-1 will be used on SSI
DVHRXV3366I node ITSOSSI1.
SUBCONFIG SRI-1
   MDISK 0191 3390 6065 15 SSI1M1 MR
*DVHOPT LNK0 LOG1 RCM1 SMS0 NPW1 LNGAMENG PWC20111116 CRCÑI
DVHRXV3366I Preceeding records were included from SRI-1 configuration
DVHRXV3366I for node ITSOSSI1.
DVHRXV3366I The following configuration SRI-2 will be used on SSI
DVHRXV3366I node ITSOSSI2.
SUBCONFIG SRI-2
   MDISK 0191 3390 4172 15 SSI2M1 MR
*DVHOPT LNK0 LOG1 RCM1 SMS0 NPW1 LNGAMENG PWC20111116 CRCRV
DVHRXV3366I Preceeding records were included from SRI-2 configuration
DVHRXV3366I for node ITSOSSI2.
   ACCOUNT SYSTEMS
   IPL CMS
   MACH ESA
   CONSOLE 0009 3215
   SPOOL 000C 2540 READER *
   SPOOL 000D 2540 PUNCH A
   SPOOL 000E 1403 A
   LINK MAINT 0190 0190 RR
   LINK MAINT 019E 019E RR
   LINK MAINT 019D 019D RR
*DVHOPT LNK0 LOG1 RCM1 SMS0 NPW0 LNGAMENG PWC20111116 CRC"ª
```

```
DVHREV3356I The following are your user option settings:
DVHREV3356I Links DISABLED Logging ON RcvMsg ON Smsg OFF NeedPW OFF Lang
DVHREV3356I AMENG
```

B.2.2 Getting identities

If you need to review the configurations of an IDENTITY only, use DIRM FOR -*machine*- REV. This command shows the complete definition, for example:

```
dirm for sri rev
```

To get all parts or several parts of the machine, use the commands for each part. The names of each part are defined in the statement BUILD in the base definition:

- `dirm for sri get`
- `dirm for sri-1 get`
- `dirm for sri-2 get`

B.2.3 Using prototypes and batch functions

DIRMAINT is designed to help you create a number of users. If you need to create more multiconfiguration virtual machines, the use of prototypes and batch functions is useful.

Example B-5 is a prototype to create an IDENTITY entry. The file is named IDENT PROTODIR. It must not contain the BUILD statements, because they are not created yet.

Example B-5 IDENT PROTODIR: Prototype model for an IDENTITY

```
IDENTITY IDENT
   INCLUDE IBMDFLT
   ACCOUNT 9999 CLUSTER1
   IPL CMS PARM AUTOCR
   MACHINE ESA
   CONSOLE 001F 3215 T
   SPOOL 000C 2540 READER A
   SPOOL 000D 2540 PUNCH A
   SPOOL 000E 1403 A
```

Also, we created a model for the SUBCONFIG entries, one model for each member. We named the files SUBCON-1, SUBCON-2, SUBCON-3, and SUBCON-4, all with the file type PROTODIR. Each file is a model to define an MDISK in the member's respective group of DASDs, which were defined in our EXTENT CONTROL file.

Example B-6 is the SUBCON-1 PROTODIR file.

Example B-6 Prototype model for a SUBCONFIG in member 1

```
SUBCONFIG SUBCON-1
   MDISK 0191 3390 AUTOG 005 VM6201 MR READ WRITE MULTIPLE
```

Example B-7 on page 109 is the SUBCON-2 PROTODIR file.

Example B-7 Prototype model for a SUBCONFIG in member 2

```
SUBCONFIG SUBCON-2
   MDISK 0191 3390 AUTOG 005 VM6202 MR READ WRITE MULTIPLE
```

Example B-8 is the SUBCON-3 PROTODIR file.

Example B-8 Prototype model for a SUBCONFIG in member 3

```
SUBCONFIG SUBCON-3
   MDISK 0191 3390 AUTOG 005 VM6203 MR READ WRITE MULTIPLE
```

Example B-9 is the SUBCON-4 PROTODIR file.

Example B-9 Prototype model for a SUBCONFIG in member 4

```
SUBCONFIG SUBCON-4
   MDISK 0191 3390 AUTOG 005 VM6204 MR READ WRITE MULTIPLE
```

Example B-10 shows the commands that we used to save the files in DIRMAINT.

Example B-10 Saving the prototype files

```
DIRM FILE IDENT PROTODIR
DIRM FILE SUBCON-1 PROTODIR
DIRM FILE SUBCON-2 PROTODIR
DIRM FILE SUBCON-3 PROTODIR
DIRM FILE SUBCON-4 PROTODIR
```

Each time that we need one or more new multiconfiguration virtual machines, we use the prototypes. Example B-11 is a BATCH file that creates two new machines. We named the file DIRMADDU BATCH.

Example B-11 The batch file to create two new multiconfiguration virtual machines

```
add cmsusr3 like ident
add cmsus3-1 like subcon-1 build on itsossi1 in cmsusr3
add cmsus3-2 like subcon-2 build on itsossi2 in cmsusr3
add cmsus3-3 like subcon-3 build on itsossi3 in cmsusr3
add cmsus3-4 like subcon-4 build on itsossi4 in cmsusr3
for cmsusr3 setpw initpw
add cmsusr4 like ident
add cmsus4-1 like subcon-1 build on itsossi1 in cmsusr4
add cmsus4-2 like subcon-2 build on itsossi2 in cmsusr4
add cmsus4-3 like subcon-3 build on itsossi3 in cmsusr4
add cmsus4-4 like subcon-4 build on itsossi4 in cmsusr4
for cmsusr4 setpw initpw
```

Example B-12 shows the command that we used to send this file to be executed in DIRMAINT.

Example B-12 Executing the batch file

```
dirm batch dirmaddu batch
```

After we created these user IDs, we personalize each machine. We use individual DIRMAINT commands, such as the command that is shown in Example B-11 on page 110. Or, you can use the traditional GET/REP commands for get and replace functions.

B.3 VMSES commands

The syntax of the PUT2PROD command is modified to enable its use with SSI. You must run the PUT2PROD EXEC from the default MAINTvrm user ID or equivalent in each member of the cluster. Use PUT2PROD SAVECMS To save the CMS segments. Use PUT2PROD SEGMENTS ALL to save all the segments. You can obtain detailed information by using help vmses put2prod or referring to the VMSES manual.

C

Additional material

This book refers to additional material that can be downloaded from the Internet as described in the following sections.

Locating the web material

The web material that is associated with this book is available in softcopy on the Internet from the IBM Redbooks web server. Point your web browser at:

ftp://www.redbooks.ibm.com/redbooks/SG248006

Alternatively, you can go to the IBM Redbooks website at:

ibm.com/redbooks

Select the **Additional materials** and open the directory that corresponds with the IBM Redbooks form number, SG24-8006.

Using the web material

The additional web material that accompanies this book includes the following file:

File name	*Description*
SG248006PW.zip	Planning worksheets

System requirements for downloading the web material

The web material requires the following system configuration:

Hard disk space:	35 KB minimum
Operating System:	Windows/Linux
Software:	Spreadsheet software that supports the XLS format

Downloading and extracting the web material

Create a subdirectory (folder) on your workstation, and extract the contents of the web material compressed file into this folder.

Related publications

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

Other publications

These publications are also relevant as further information sources:

- *z/VM V6R2.0 CP Planning and Administration*, SC24-6178-02
- *z/VM V6R2.0 Getting Started with Linux on System z,* SC24-6194
- *z/VM V6R2.0 Installation Guide,* GC24-6246
- *z/VM CMS File Pool Planning, Administration and Operation,* SC24-6167
- *z/VM Migration Guide*, GC24-6201
- *z/VM V6R2.0 General Information*, GC24-6193-02
- *z/VM V6R2.0 VMSES/E Introduction and Reference*, GC24-6243-01
- *z/VM V6R2.0 Directory Maintenance Facility Commands Reference,* SC24-6188-02
- *z/VM V6R2.0 Connectivity*, SC24-6174-02
- *z/VM V6R2.0 CP Commands and Utilities Reference*, SC24-6175-02
- *z/VM V6R2.0 Migration Guide,* GC24-6201
- *z/VM V6R2.0 RACF Security Server System Programmer's Guide*, SC24-6219

Online resources

These websites are also relevant as further information sources:

- z/VM V6.2 resources

  http://www.vm.ibm.com/zvm620/

- z/VM program directories

  http://www.ibm.com/vm/progdir/

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services

Index

IBM Redbooks

An introduction to z/VM Single System Image (SSI) and Live Guest Relocation (LGR)

(0.2"spine)
0.17"<->0.473"
90<->249 pages

An introduction to z/VM Single System Image (SSI) and Live Guest Relocation (LGR)

Learn about the business benefits of SSI and LGR

Understand important concepts and planning

Step through implementation and practical approaches

IBM z/VM 6.2 introduces significant changes to z/VM in the form of multiple system clustering technology allowing up to four z/VM instances in a single system image (SSI) cluster. This technology is important, because it offers clients an attractive alternative to vertical growth by adding new z/VM systems. In the past, this capability required duplicate efforts to install, maintain, and manage each system. SSI reduces or eliminates these duplicate efforts.

Support for live guest relocation (LGR) allows you to move Linux virtual servers without disruption to the business, helping you to avoid planned outages. The z/VM systems are aware of each other and can take advantage of their combined resources. LGR enables clients to avoid loss of service due to planned outages by relocating guests from a system requiring maintenance to a system that remains active during the maintenance period.

Together, the SSI and LGR technologies offer substantial client value, and they are a major departure from past z/VM practices.

This IBM Redpaper publication gives you a broad understanding of the new SSI architecture and an overview of LGR. We show an LGR example that shows a typical SAP user environment. In our example, the SAP Application Server Central Instance resides on a Linux on System z guest and an IBM DB2 10 database server runs on z/OS.

This book is written for IT architects, who design the systems, and IT specialists, who build the systems.

SG24-8006-00 ISBN 0738436623