IBM

Experiences with Oracle 10g Database for Linux on zSeries

- Installing a single instance of Oracle Database 10g
- Installing Oracle 10g RAC
- Using ASM

Kathryn Arrell
Laurent Dupin
Dennis Dutcavich
Terry Elliott
Bruce Frank
Chris Little
Barton Robinson
Tom Russell

Redbooks

ibm.com/redbooks

Archived

Archived

International Technical Support Organization

Experiences with Oracle 10g Database for Linux on zSeries

August 2005

Archived

SG24-6482-00

Note: Before using this information and the product it supports, read the information in "Notices" on page xiii.

Archived

First Edition (August 2005)

This edition applies to Oracle Database 10g (10.1.0.3).

© Copyright International Business Machines Corporation 2005. All rights reserved.
Note to U.S. Government Users Restricted Rights -- Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Figures ix

Notices xiii
Trademarks xiv

Preface xv
The team that wrote this redbook xv
Become a published author xvii
Comments welcome xvii

Chapter 1. Overview of Oracle Database 10g for Linux on zSeries 1
1.1 What is Linux 2
1.2 Linux on zSeries 2
1.3 Why Oracle9i and Oracle 10g for Linux on zSeries 3
1.3.1 Expanded application portfolio 4
1.3.2 Cost savings 5
1.3.3 Server consolidation and faster time to market 5
1.4 Oracle Database **10g** for Linux on zSeries 7
1.4.1 Oracle9*i* Database Server and Oracle Database 10g 7
1.4.2 Oracle9*i* Application Server and AS 10g 8
1.4.3 Oracle application solutions for Linux on zSeries 8
1.5 What distributions of Linux Oracle supports 9
1.6 Obtaining a copy of Oracle Database 10g for Linux on zSeries 9
1.7 Sizing workloads for Oracle10g for Linux on zSeries 9

Chapter 2. Best practices for installing an Oracle Database 10g on Linux on zSeries 11
2.1 Memory sizing and configuration 12
2.2 CPU allocation 13
2.2.1 Sizing 13
2.2.2 CP allocation 14
2.2.3 Setting shares 15
2.3 Paging, swap, and I/O considerations 16
2.3.1 z/VM paging 17
2.3.2 Swap space for Linux 17
2.3.3 I/O considerations 18
2.3.4 Monitoring the system 19
2.4 Summary 19

Archived

© Copyright IBM Corp. 2005. All rights reserved.

Chapter 3. Installing Oracle 10g single instance 21
3.1 Installing Oracle Database 10g on zSeries with Linux 22
3.2 Preparing the system environment 22
3.2.1 Setting up the fixed buffer option with ECKD™ disks 23
3.3 Setting up an xWindows interface using VNC 24
3.4 Downloading the code 26
3.4.1 Finding the documentation 27
3.4.2 Checking the Linux kernel settings 28
3.5 Running the Universal Installer 29
3.5.1 Starting the OUI 30
3.5.2 Initial OUI panels 31
3.5.3 Inventory directory panel 32
3.5.4 Changing to root screen for oraInstroot script 33
3.5.5 File location panel 35
3.5.6 Installation type 36
3.5.7 Selecting database configuration 37
3.5.8 Database configuration options 37
3.5.9 Selecting the database management options 38
3.5.10 Selecting the database file storage 39
3.5.11 Selecting the backup and recovery options 40
3.5.12 Choose the database passwords 41
3.5.13 Summary 42
3.5.14 Install completing 43
3.5.15 Configuration Assistant panel 44
3.6 Verifying that the database is running 49
3.7 Enabling Async IO 50
3.8 Using the LOCK_SGA parameter 51
3.9 Using OEM to manage an Oracle database 52

Chapter 4. Installing an Oracle 10g Database with ASM 55
4.1 ASM overview 56
4.2 Setting up ASM 56
4.3 Binding disks to raw devices 57
4.4 Configuring ASM instance using DBCA 58
4.5 Managing ASM using SQL commands 66
4.5.1 Connect to the ASM instance 67
4.5.2 Creating a new diskgroup 67
4.5.3 Modifying an existing diskgroup 68
4.6 Managing ASM using OEM 70
4.7 ASM best practices 74

Chapter 5. Installing CRS and RAC 77
5.1 VM set up 78

Archived

5.2 Linux setup 82
5.2.1 Setting the kernel values 83
5.2.2 Moving the scripts to both nodes 83
5.2.3 Creating the raw devices 84
5.2.4 Create Oracle account 84
5.2.5 Set up logical volumes 84
5.2.6 Making symbolic links 85
5.2.7 Binding the raw devices 86
5.2.8 Set up the /etc/host file 87
5.2.9 Set up ssh to work without password 88
5.3 Preparation review 89
5.4 Oracle CRS installation 90
5.4.1 Cleaning up CRS if you need to reinstall 100
5.5 Oracle RAC installation 101
5.5.1 VIP configuration 106
5.6 Oracle Database creation 110
5.6.1 Setting up the user profile 119

Chapter 6. Using Tivoli Storage Manager and Tivoli Data Protect for Oracle Database 10g 121
6.1 IBM Tivoli Storage Manager overview 122
6.2 Tivoli Storage Manager architecture 123
6.3 Tivoli Data Protection for Oracle 123
6.4 RMAN and Tivoli Data Protection for Oracle 124
6.5 Overview of installation process of TSM and TDPO 124
6.5.1 Configuring RMAN 126
6.5.2 Installing TSM server 128
6.5.3 Install Tivoli Data Protect for Oracle 132
6.6 Back up the user tablespace 139
6.7 Restore and recover the users Tablespace 146
6.7.1 Restore and recover process 147
6.8 Summary 152

Chapter 7. Using Cobol and C/C++ with Oracle Database 10g 153
7.1 Working with Pro*Cobol and sample programs 154
7.1.1 Install the Pro*COBOL precompiler 154
7.1.2 Sample Pro*COBOL programs 156
7.2 Using ACUCOBOL-GT Version 6.1 157
7.2.1 Relinking ACUCOBOL-GT with Oracle 157
7.2.2 Work with the Oracle Pro*COBOL samples 161
7.2.3 Prepare and run the sample programs 162
7.3 Running MicroFocus Cobol 163
7.3.1 Makefile for sample Pro*COBOL programs 163

Archived

7.3.2 Makefile output for sample1 program 163
7.3.3 Execution of sample1 program 164
7.3.4 User programs 165
7.4 Oracle 10g Pro*C/C++ Precompiler 165
7.4.1 Run the Installer 165
7.4.2 Pro*C/C++ demonstration programs 166
7.4.3 Creating demo tables 167
7.4.4 Precompile and compile C source 167
7.4.5 Creating and executing sample2 168

Chapter 8. Monitoring VM and Linux 171
8.1 Oracle measurements 172
8.2 Configuration guidelines 172
8.2.1 Minimize Total Storage Footprint® 172
8.2.2 SGA must fit in memory 174
8.2.3 Use Oracle direct I/O 175
8.2.4 Use virtual disk for swap 176
8.2.5 Enable the timer patch 177
8.2.6 Use virtual switch 179
8.2.7 Use expanded storage for paging 180
8.2.8 Ensure sufficient page space 180
8.3 Storage analysis 181
8.3.1 Detecting storage problems - Paging 182
8.3.2 Detecting 2 GB storage problems - Paging 182
8.3.3 Detecting 2 GB problems - Demand scan 183
8.3.4 Detecting 2 GB problems - State analysis 185
8.4 I/O subsystem 187
8.4.1 LVM 187
8.5 Processor analysis 190
8.6 LPAR weights and options 193
8.6.1 Physical LPAR overhead 196
8.6.2 Converting weights to logical processor speed 197
8.6.3 LPAR analysis example 198
8.6.4 LPAR options 198
8.6.5 Shared versus dedicated processors 199

Chapter 9. Using Radius Server and z/OS RACF LDAP for Oracle DB user authentication 201
9.1 Overview 202
9.2 FreeRADIUS on Linux on z/OS 202
9.3 z/OS LDAP 205
9.4 Oracle DB Advanced Security Option (ASO) 208
9.5 Oracle client 211

Archived

Appendix A. VM setup and useful commands 213
VM setup 214
VM guest definition 214
VM System definition 215
Cloning 216
FLASHCOPY 217
Booting same Linux either as VM guest or LPAR 219
Useful VM commands 220
How to remove Oracle code 221

Appendix B. Overview of ESALPS 223
ESALPS overview 224
ESALPS features 224
Critical agent technology 225
Monitoring requirements 226
Standard interface 226

Related publications 227
IBM Redbooks 227
Other publications 227
Online resources 228
How to get IBM Redbooks 228
Help from IBM 228

Index 229

Archived

Archived

Figures

1-1 Linux application deployment on server farms 6
1-2 Linux application deployment on zSeries 6
2-1 Using shares to manage guest priorities 16
3-1 VNC Viewer with a window for oracle and root 26
3-2 VNC server IP address 30
3-3 VNC password 30
3-4 First panel 32
3-5 Welcome panel 32
3-6 Specify Inventory directory panel 33
3-7 orainstRoot script 34
3-8 Results from running orainstRoot script 35
3-9 Specify FIle Locations panel 36
3-10 Specify Installation Type panel 36
3-11 Select Database Configuration panel 37
3-12 Specify Database Configuration Options panel 38
3-13 Select Database Management Option panel 39
3-14 Select Database File Storage Option panel 40
3-15 Select Backup and Recovery Options panel 41
3-16 Specify Database Schema Passwords panel 42
3-17 Summary panel 43
3-18 Installation progress panel 44
3-19 Configuration Assistants panel 45
3-20 Database creation panel 45
3-21 Password management panel 46
3-22 Unlock Scott panel 47
3-23 Setup Privileges panel 47
3-24 Results of running root.sh script 48
3-25 End of Installation panel 49
3-26 Logon screen to OEM 52
3-27 Home screen for Oracle Enterprise Manager 53
4-1 Step 6 - Storage options 59
4-2 Step 7 - Create ASM instance 60
4-3 ASM instance creation confirmation 60
4-4 ASM parameters 61
4-5 ASM diskgroups 62
4-6 Create Disk Group 63
4-7 Disk group selection 64
4-8 Database files location 65

© Copyright IBM Corp. 2005. All rights reserved.

4-9 Recovery configuration 66
4-10 OEM primary panel 70
4-11 ASM main window 71
4-12 Diskgroup view. 72
4-13 Add Disk window 72
4-14 Files window 73
4-15 Performance window 74
5-1 Possible RAC setup on one zSeries system 78
5-2 For the first installation entry the inventory path 92
5-3 Request to run root.sh 92
5-4 Enter the destination path 93
5-5 Specify language 93
5-6 Enter the cluster nodes 94
5-7 Specify the network interface usage 95
5-8 Cluster registry location 96
5-9 File name for voting disk 97
5-10 Request to run root.sh 97
5-11 Summary for CRS 98
5-12 Installation of CRS 98
5-13 Configuration script 99
5-14 Configuration assistants. 99
5-15 End of CRS installation 100
5-16 Select the cluster nodes. 102
5-17 Select installation type 103
5-18 Select database configuration 103
5-19 Summary of the RAC installation 104
5-20 Running the root.sh script 105
5-21 End of Oracle RAC Database installation 109
5-22 Welcome screen for DBCA process 110
5-23 Select the DBCA operation 111
5-24 Select the RAC nodes 111
5-25 Select the database template for DBCA to use 112
5-26 SID same 112
5-27 Choosing OEM or GRID manager 113
5-28 Enter password information 113
5-29 Choose storage option. 114
5-30 Chose recovery option 114
5-31 Using custom scripts 115
5-32 Expand the service information 115
5-33 Validate the memory for SGA and PGA. 116
5-34 Check file locations 116
5-35 Select database creation options. 117
5-36 Review the database options. 117

Archived

5-37 Install panel 118
5-38 End of database creation panel 118
6-1 Configuring TSM server with Web Admin 130
6-2 Successful results after configuring with Web Admin 131
6-3 Set time out parameter 131
6-4 Registering the TDPO node 134
6-5 Registration for TDPO client completed 135
6-6 Creating additional backup storage pool space - Backup pool 137
6-7 Configuring the new backup pool volume 138
6-8 Additional backup pool complete 139
6-9 Query the storage pool for results 142
6-10 Results of query of storage pool 143
6-11 File space name to query the file space occupancy 144
6-12 Query file space occupancy 145
6-13 Query results 146
7-1 Select Installation Type panel 155
7-2 Summary panel 155
8-1 ESAUCD2 report (Linux memory analysis) 174
8-2 ESAUSR3 (user resource utilization) 175
8-3 ESAVDSK VDISK Analysis Report 177
8-4 ESAUSRQ - User queue and load analysis 179
8-5 ESAPAGE - Paging analysis 1 181
8-6 ESAPAGE - Paging analysis 2 181
8-7 ESASSUM subsystem activity 183
8-8 Custom extract for demand scan measurements 185
8-9 ESAXACT - Transaction delay analysis 1 186
8-10 ESAXACT - Transaction delay analysis 2 187
8-11 ESAUSEK - User DASD seeks report 188
8-12 ESAXACT transaction delay analysis 189
8-13 ESAUSR3 User Resource Utilization 190
8-14 CP commands to get a trace of diagnose 44 instructions 190
8-15 Sample CP trace of diagnose code 44 191
8-16 Rexx exec to reduce trace 191
8-17 Reduced trace of diagnose code 44 192
8-18 Fragment of Linux system map 193
8-19 ESALPAR Logical Partition Analysis Part 1 194
8-20 ESALPAR Logical Partition Analysis Part 2 195
8-21 LPAR physical CPU management time 197
9-1 Order of processing 202
9-2 Oracle Advanced Security pull-down 209
9-3 Authentication 210
9-4 Oracle Advanced Security Other Parameters tab 210
9-5 Promotion of radius 212

Archived

Archived

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 Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites 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 publication is intended to help those who are installing Oracle Database 10g for the first time. The information in this publication is not intended as the specification of any programming interfaces that are provided by Oracle. See the PUBLICATIONS section of the Oracle publications for more information about what publications are considered to be product documentation.

Information concerning Oracle's products was provided by Oracle. The material in this document has been produced by a joint effort between IBM and Oracle zSeries Specialists.The material herein is copyrighted by both IBM and Oracle.

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 IBM Corp. 2005. All rights reserved.

COPYRIGHT LICENSE:
This information contains sample application programs in source language, which illustrates 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. 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 IBM's application programming interfaces.

Trademarks

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

AIX®	Multiprise®	S/390®
ECKD™	MVS™	Tivoli®
ESCON®	OS/390®	VM/ESA®
Footprint®	PR/SM™	z/OS®
FICON®	Redbooks™	z/VM®
IBM®	Redbooks (logo) ™	zSeries®
ibm.com®	RACF®	

The following terms are trademarks of other companies:

PDB, Solaris, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, Intel logo, Intel Inside logo, and Intel Centrino logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group 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, and service names may be trademarks or service marks of others.

Archived

Preface

Linux on zSeries offers many advantages to customers who rely upon IBM mainframe systems to run their businesses. Linux on zSeries takes advantage of the qualities of service in the zSeries hardware—making it a robust industrial strength Linux. This provides an excellent platform for consolidating Oracle databases that exist in your enterprise.

This IBM® Redbook describes experiences gained while installing and testing Oracle10g for Linux® on zSeries®, such as:

- Installing a single instance database of Oracle10g instances for Linux on zSeries
- Installing Cluster Ready Services (CRS) and Real Application Clusters (RAC)
- Performing basic monitoring and tuning exercises
- Using options such as:
  - IBM's Tivoli Data Protector (TDP) and Tivoli® Storage Manager (TSM)
  - Automated Storage Manager (ASM)
  - LDAP and Radius Server for security
  - COBOL and C programs with Oracle

Interested readers include database consultants, installers, administrators, and system programmers.

The team that wrote this redbook

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

Kathryn Arrell is an Oracle Specialist at the IBM/Oracle International Competency Center at IBM San Mateo. Previously she worked as an ERP specialist at the ITSO in Poughkeepsie, New York.

Laurent Dupin has worked on z/VM® and Linux performance since he joined the EMEA zSeries Benchmark Center in Montpellier, France. He previously worked for 10 years as a z/OS® and Sysplex specialist for IBM Global Services, and became interested in Linux on zSeries during a two-year assignment at the Boeblingen Lab in 1999.

© Copyright IBM Corp. 2005. All rights reserved.

Dennis Dutcavich is a zSeries Oracle Specialist with the American sales division. Dennis is part of Sales and Distribution in the Americas. He is Technical Sales Specialist supporting Linux on zSeries opportunities.

Terry Elliott is a zSeries Specialist working in the IBM/Oracle International Competency Center at IBM San Mateo. He has over 30 years of experience in information technology. Before joining the IBM Oracle International Competency Center six months ago, Terry was as an ERP zSeries Performance Specialist.

Bruce Frank is a zSeries Oracle Specialist in the IBM/Oracle International Competency Center at IBM San Mateo.

Chris Little works at the Department of Human Services as a Unix/Linux/z/VM Administrator. It was his pressure that got DHS into the Early Adopters Program for Oracle, and he helped implement one of the first production Oracle databases on Linux/390.

Barton Robinson is president of Velocity Software, Inc. He started working with VM in 1975, specializing in performance starting in 1983. His previous publication experience includes the VM/HPO Tuning Guide published by IBM, and the VM/ESA® Tuning Guide published by Velocity Software. He is the author and developer of ESAMAP and ESATCP.

Tom Russell is a zSeries Specialist with IBM Canada. He spent several years on a special assignment with Oracle in Redwood Shores, California.

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

Neil Rasmussen
IBM Tivoli Development San Jose

Masayuki Kamitohno
zSeries ATS, IBM Japan

Roy Costa
Mike Ebbers
Julie Czubik
International Technical Support Organization, Poughkeepsie Center

Mark Polivka
Betsie Spann
Mike Morgan
Oracle Corporation

There were many others from the IBM labs in Boeblingen, Germany; Poughkeepsie, New York; and Endicott, New York; as well as well as the those from Oracle Corporation, Redwood Shores, California, who contributed to the

Archived

technical information provided in this book and the technical reviews of the material. We appreciate all the support we received.

Become a published author

Join us for a two- to six-week residency program! Help write an IBM Redbook dealing with specific products or solutions, while getting hands-on experience with leading-edge technologies. You'll team with IBM technical professionals, Business Partners and/or customers.

Your efforts will help increase product acceptance and customer satisfaction. As a bonus, you'll develop a network of contacts in IBM development labs, and increase your productivity and marketability.

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 Redbooks™ to be as helpful as possible. Send us your comments about this or other Redbooks in one of the following ways:

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

  ibm.com/redbooks

- Send your comments in an email to:

  redbook@us.ibm.com

- Mail your comments to:

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

Archived

Archived

1

Overview of Oracle Database 10g for Linux on zSeries

Oracle released Oracle9*i* Database Server for Linux on zSeries in August 2002. Since that time, many customers have consolidated database instances on to Linux guests on zSeries. In October 2004, Oracle released Oracle Database 10g (10.1.0.3). Several IBM sites have installed and tested the code. This document shares some of the experiences from these installations and from the development environment used at Oracle in Redwood Shores.

This chapter is an overview of Linux on zSeries and Oracle solutions.

© Copyright IBM Corp. 2005

1.1 What is Linux

Linux is a UNIX®-like operating system initially created by Linus Torvalds when he was a graduate student in 1991. The objective for developing Linux was to deliver a non-proprietary operating system and application development environment, completely independent of underlying hardware architectures, that would offer maximum freedom to move applications from one hardware platform to another by simply recompiling code, without expensive, labor-intensive porting efforts.

Linux is a fully networked 32-bit/64-bit architecture, that supports multiple users, multitasking, and multiprocessors, with a user-friendly Xwindows Graphical User Interface. Continued development and testing of Linux is governed by the Open Source community, which uses the Internet as the primary vehicle for technical exchange. Linux source code can be downloaded free of charge from the Internet, and programmers are free to modify the code. However, the integrity of the official kernel source code is managed and maintained by a strict submission and review process controlled by the Linux Review Board, an international standards body for Linux.

Linux, like other Open Source software, is distributed under the terms of the GNU Public License (GPL), and is packaged and distributed by approved distributors, such as Caldera, Red Hat, SuSE, and Turbo Linux. Distributor packages include the Linux operating system code that has been precompiled for specific hardware environments, along with other Open Source applications and middleware, such as Apache Web Server, SAMBA (file/print serving), Jabber (instant messaging), and IMAP/POP (mail servers). Distributors also offer Linux services and support packages, as does IBM Global Services.

1.2 Linux on zSeries

In 1998 IBM announced a commitment to support Linux on all its hardware platforms, including zSeries. The 31-bit version of the Linux operating is available from Linux distributors for S/390® G5 and G6 processors, and on 64-bit zSeries models. The Linux 64-bit support on the zSeries is available on Open Source and became available from SuSE in April 2001.

The Linux operating system has been running in a zSeries test environment since early 1999. As of December of 1999, the IBM Web site (`http://www.ibm.com/servers/eserver/zseries/solutions/s390da/linuxisv.html`) is continually updated to show all the Independent Software Vendors (ISVs) applications or infrastructure available for the zSeries platform.

Linux on zSeries is an ASCII environment that takes advantage of IBM zSeries hardware, especially for system availability and I/O performance. Of particular interest is the ability to run many Linux images under z/VM. This provides an excellent environment for server consolidation. It will more fully utilize system resources and facilitate system management.

Linux is not a replacement for other IBM operating systems on zSeries or S/390, and will coexist with z/OS, OS/390®, and VM/ESA. It supports such UNIX tools as sed, awk and grep, compilers like C, C++, Fortran, Smalltalk, and Ada. Network tools like Telnet, ftp, ping, and traceroute are supported as well.

1.3 Why Oracle9*i* and Oracle 10g for Linux on zSeries

The ability to combine the hardware characteristics of the IBM zSeries with the openness of Linux provides significant benefits to users.

IBM's zSeries has, over the years, demonstrated its unique ability to run multiple diverse work loads. This key strength also applies to Linux workloads. zSeries is particularly well suited to the hosting of multiple lightly to moderately loaded servers. (Examples of this type of server would be firewall servers, print and file serving, Domain Name Servers, Internet news servers, or Web serving that is not processor intensive.) Most IT installations use multiple outboard servers to perform these functions. Consolidation of these functions onto a single zSeries can provide the following benefits:

- zSeries qualities of service: No other platform offers the qualities of service available in zSeries and z/VM.
- Reduced hardware costs: Processors, storage, memory, etc. are now shared on a single zSeries.
- Reduced software costs: Software licenses spread over several machines and operating systems in most cases are reduced when consolidated onto zSeries processors.
- Reduced networking costs: Physical networking gear like routers and cabling are no longer necessary on a single zSeries. Servers communicate through inter-system facilities on zSeries (hardware or software).
- More efficient inter-server communication: Inter-server communication is faster than physical networking.
- Reduced systems management/support costs: Supporting multiple servers on a single platform requires less effort and fewer people.
- Reduced deployment time for new servers: Since new servers are added virtually instead of physically, the time required to create a new server is minutes, not weeks.

Archived

Linux on zSeries offers many advantages to customers who rely upon z/OS systems to run their businesses. While z/OS has key strengths in the areas of data acquisition through high-volume transaction processing and data management, its value is enhanced by the addition of Linux on zSeries in different ways.

Linux on zSeries brings the zSeries user the ability to access host data efficiently by using high speed, low latency, inter-partition communication. Benefit is derived from the elimination of outboard servers, routers, and other networking gear, reduced floor space, and reduced maintenance cost. Linux also has the well-earned reputation for rapid deployment of applications—giving zSeries users a key choice in how they choose to deploy an application.

Linux on zSeries takes advantage of the qualities of service in the zSeries hardware—making it a robust industrial strength Linux, while zSeries native services are available for applications that require the qualities of service inherent in z/OS as well as those of the zSeries hardware.

There are several other advantages to running Linux on zSeries. They are described in the following sections.

1.3.1 Expanded application portfolio

The non-proprietary environment of Linux opens the door for zSeries to perform as an application development and deployment server. The Linux application portfolio will greatly increase the number of applications available to zSeries customers who want to continue to leverage the critical advantages of the platform to run business applications on a highly available and reliable architecture. Customers will now have the flexibility to develop new applications directly for Linux on zSeries and run them on any Linux-supported platform; or to develop and test Linux on other platforms, such as the desktop, and then run the applications for Linux on zSeries, with a simple recompile. As applications from ISVs become generally available, customers will be able to deploy them quickly on zSeries without any special porting effort.

Note: The MP 3000 and 9672 machines are 31-bit only and support the s390 Linux distributions. Oracle 9*i* is a 31-bit application and is the only Oracle that these machines can run. zSeries machines are 64-bit processors and can run the 31-bit (s390) and the 64-bit (s390x) distributions of Linux. The s390x Linux distribution can run Oracle 10g, which is a 64-bit application, or Oracle 9*i*, a 31-bit Oracle database.

1.3.2 Cost savings

A new feature designed specifically for the Linux operating environment, the Integrated Facility for Linux (IFL), is now available on G5, G6, and all zSeries server models including the Multiprise® 3000. The IFL gives you the ability to dedicate processors to the Linux operating system on a logically partitioned machine, transparently to the z/OS operating system. Processors dedicated to the Linux LPAR under IFL are priced at a lower rate than those for the z/OS environment. The added capacity, because it is dedicated to Linux workloads, does not increase the software licensing fees of the zSeries environment. Software pricing is confined to the capacity of only those processors enabled to the zSeries LPARs on the system. (This reflects pricing as of the publishing of this book and may have changed since the publication date. Consult your IBM server sales representative to obtain the most current pricing structure information.)

1.3.3 Server consolidation and faster time to market

All of the great flexibility and openness of Linux combined with the outstanding qualities of service of zSeries results in an industrial-strength Linux environment.

zSeries has several options for partitioning server resources into multiple logical servers. The PR/SM™ hardware feature provides the ability to partition the physical machine into many logical servers (15 on a 9672 and 30 on a zSeries), or LPARs, with dedicated, or shared, CPU and memory. The z/VM operating system allows you to partition an LPAR horizontally into multiple logical operating system images. These unique partitioning features provide the capability to quickly and easily consolidate a large number of Linux servers onto a single zSeries server. This industry-leading zSeries technology for dynamically sharing processing resources across multiple logical systems delivers value to the Linux environment not provided by any other architecture on the market today.

Figure 1-1 on page 6 illustrates the typical Linux or UNIX application deployment strategy used by most installations deploying applications on non-zSeries servers today.

Archived

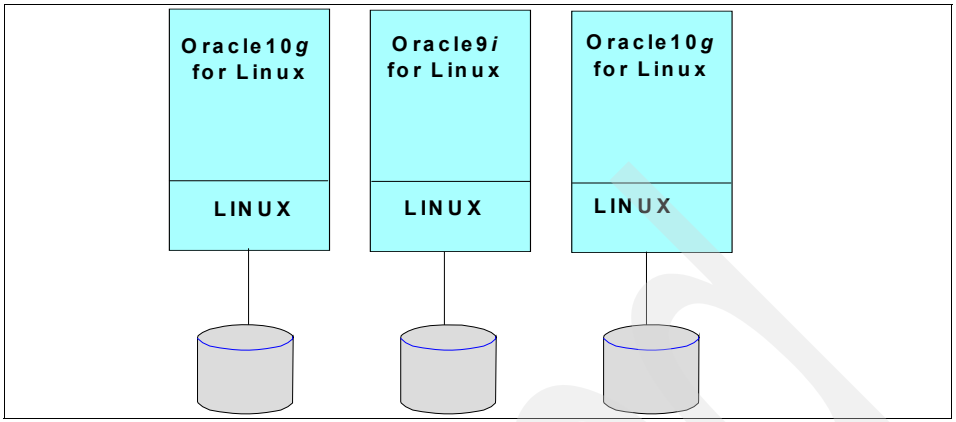


Figure 1-1 Linux application deployment on server farms

Without the robust operating environment provided by the z/OS operating system and the industry-leading reliability of the zSeries hardware, most installations choose to deploy each new application in its own isolated operating system environment, on its own server, with a dedicated database or database partition. This leads to large server farms, with applications that cannot be easily integrated, and require complex systems management.

Contrast the configuration in Figure 1-1 to the flexibility available on the zSeries-S/390 architecture illustrated in Figure 1-2.

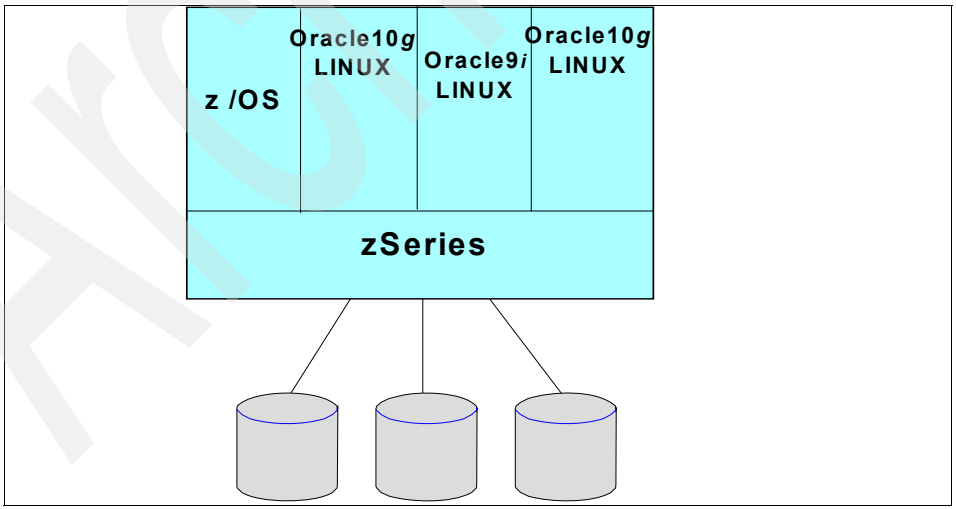


Figure 1-2 Linux application deployment on zSeries

Archived

The unique capabilities of zSeries for running multiple operating images simultaneously and sharing processing resources dynamically, supports many diverse workloads and multiple applications on a single server, with outstanding interpretability and integration between applications. Database sharing with integrity and ease of systems management are additional unique benefits that zSeries brings to the Linux operating environment.

In fact, many customers are blending the data richness of the zSeries environments with the Web capability of Linux applications to deliver highly integrated, cost-effective e-business solutions today.

1.4 Oracle Database 10g for Linux on zSeries

Oracle has four main families of products:

- Oracle Database Server (9*i* and 10g)
- Oracle9*i* Application Server and Oracle Application Server 10g
- Oracle 11*i* eBusiness Suite
- Oracle Collaboration Suite

See the following Web site for the latest information:

http://www.metalink.oracle.com

1.4.1 Oracle9*i* Database Server and Oracle Database 10g

Oracle delivered their production level Oracle9*i* Enterprise Edition Database Server on September 1, 2002. This release included the Oracle9*i* Release 2 (9.2) Enterprise Edition Server. The i in 9*i* stands for Internet and the *g* in 10g stands for grid.

This release includes most of the components of the Enterprise Edition including:

- Real application clusters
- OLAP
- Spatial
- Label security
- Partitioning
- Data mining
- Advanced security
- Pro*COBOL

The client package is also included, including Oracle Net Services, OCI, and the Pro*C/C++ precompiler.

Archived

The following functions are not yet provided:

- OracleText does not have the INSO filters.
- The Oracle Management Server (OMS) function of OEM does not run in Linux.
- No generic connectivity (ODBC and OLE).
- Authentication with Verisign, CyberTrust, and Entrust.

This provides a complete Oracle9*i* database or 10g Database for development, testing, and production in Linux on zSeries.

1.4.2 Oracle9*i* Application Server and AS 10g

In many ways, Oracle9*i* Application Server is well suited to Linux on zSeries. Linux, at its current state of development, is better suited for horizontally scaled workloads, such as print serving, Web serving, and simple application serving, or smaller scale database workloads. In building Oracle9*i* Application Server, Oracle uses many advanced elements for supporting Internet computing requirements like portals, security, commerce, etc. Some of these elements are not yet available outside the Intel® and more established UNIX environments. Oracle and IBM are working to ensure that these partner products can be delivered.

The Oracle 10g Application Server

Also on metalink, support for AS10g for Linux on zSeries is shown as projected. These are works in progress at the time of writing of this book.

1.4.3 Oracle application solutions for Linux on zSeries

In this section we discuss Oracle application solutions for Linux on zSeries.

The Oracle E-Business Suite

Oracle has delivered is full suite of products, Oracle Database 10g, Application Server 10g, and the Oracle E-Business Suite on Linux, on Intel hardware. Oracle EBS requires a 10.1.0.4 level of the database. On metalink, support for the split tier mode where the database will run on Linux on zSeries is shown as projected.

The Oracle Collaboration Suite

The newest suite of products is the Oracle Collaboration Suite. The product is a complete collaboration suite, including calendar, e-mail, files, voicemail, and workflow. It enables the consolidation of messaging and collaboration

infrastructure to achieve cost efficiency. Oracle Collaboration Suite is supported in a split tier mode where the database server can be run on Linux on zSeries.

1.5 What distributions of Linux Oracle supports

Oracle9*i* is a 31-bit product. Oracle9*i* for Linux on zSeries runs on SLES8 31 bit and SLES8 64 bit. There are plans to certify Oracle9*i* on SLES9.

Oracle 10g is a 64-bit product. Oracle 10g runs on SLES8 and SLES9 64 bit. There are plans to certify Oracle 10g on RedHat 4 U1.

The latest information on supported platforms is found at:

http://www.otn.oracle.com/support/metalink/content.html

1.6 Obtaining a copy of Oracle Database 10g for Linux on zSeries

The Oracle Database 10g Server can be purchased through the Oracle Store (https://store.oracle.com/), or CD images can be downloaded from the Oracle Technical Network (http://otn.oracle.com/).

1.7 Sizing workloads for Oracle10g for Linux on zSeries

The selection of an application to start testing the Oracle10*OK,* Database Server, or any application for that matter, is critical to the success of the test. It makes sense to select an application that has the characteristics of most of the applications that would be considered for server consolidation. It is very important to get a sizing of the workload to be moved to Linux on zSeries. It is also sensible to start with an application that is small or not complicated and use that as a vehicle for skill building in this area.

It is important to understand that a sizing is an estimate of capacity needs. The performance team in Poughkeepsie developed a sizing methodology for equating a workload on a UNIX or Intel platform to its equivalent MIPS on zSeries. While this process has been generally accurate for several years, it is not a capacity plan. Capacity plans are based on the past performance of a system. The changing capacity needs over time provide an accurate way to predict future needs.

If this is a new application that has never run before, the sizing becomes a bit more difficult. Predictions can generally be made from testing done during

development. If this is a vendor's application, they may be able to provide sizing assistance for their application in a Linux environment.

There are two significant reasons to start with a sizing. First, it is important to understand the resources needed to run the application. This includes peak utilization on this server as well as when others peak. The sum of all peaks becomes important as more servers are moved to Linux on zSeries. In fact, this step helps start the process of understanding what the Total Cost of Ownership (TCO) results will be.

The second important reason for performing a sizing is to set expectations for testing of the workload that is moved. This will provide a starting point that can be used for tuning in either Oracle10g, Linux, or z/VM, if necessary.

There is a service called SIZE390 that is available to provide pre-sale processor sizing estimates to IBM sales representatives and Business Partners for IBM zSeries and S/390 systems running z/OS or OS/390 and Linux. It is now available world wide.

SIZE390 provides a questionnaire that requests all the information needed to size workloads. The questionnaire can be obtained by IBM sales representatives or Business Partners through TechXpress.

You should be prepared to supply the following:

- Machine model and characteristics such as:
  - CPU MHz
  - Number of CPUs
  - Memory
- Type of application
  - DSS
  - OLTP
  - DNS
  - etc.
- Approximate number of users
- Utilization profile
  - VMSTATS should be provided for a peak period.

This sizing deliverable provides some information about sizing estimates for Linux. Keep in mind that to get exact sizing estimates you will need to test in each customer's environment, but this data is useful for measuring other possible consolidation efforts in the customer location.

Archived

2

Best practices for installing an Oracle Database 10g on Linux on zSeries

To obtain optimum performance from Oracle RDBMS when running in Linux on zSeries, the proper configuration of both Linux and z/VM are extremely important. Our experiences have demonstrated that the vast majority of performance concerns were solved by making changes to z/VM, Linux, or the I/O subsystem. This chapter discusses the major areas of concern.

- Sizing and tuning memory
- CPU allocation
- Paging, swap, and I/O considerations
- Monitoring performance
- Summary

For more information on VM tuning recommendations, see:

http://www.vm.ibm.com/perf/tips/

© Copyright IBM Corp. 2005. All rights reserved.

2.1 Memory sizing and configuration

The sizing and allocation of memory is one of the most critical areas of the implementing Oracle in Linux under z/VM. This was especially so with 31-bit Oracle9*i* and 31-bit Linux distributions. Even though Oracle Database 10g 64-bit runs in 64-bit Linux, memory sizing still affect performance.

When running under z/VM, there is a need for expanded storage. Even though z/VM is essentially 64-bit and supports 64-bit virtual machines, expanded storage must be configured when running Oracle. The need for a expanded storage still exists even with z/VM. When running under the current versions of z/VM, we found that you need expanded storage in the configuration. z/VM, even in a 64-bit environment, requires EStore to use as a paging device. Performance will not be acceptable unless you configure from 25 percent to 33 percent of the storage in the z/VM system as EStore.

This hierachy is needed because CP and the control blocks in z/VM must reside below 2 GB, or what is referred to as host memory, including a guest page being referenced for CP processing (I/O, IUCV, etc.). With high amounts of I/O (disk and network as well), this can create contention for storage below the 2 GB level. In z/VM 4.2.0 and earlier z/VM releases, when we steal a page below the 2 GB bar due to contention, we do not move it to central storage above the 2 GB; we page it out. If there is no expanded storage, it gets paged out to DASD. This can create a thrashing scenario. If you see paging to DASD, but lots of storage available above 2 GB, then there is probably contention for storage below 2 GB.

As a starting point, at least 25 percent of the amount of memory assigned to the LPAR that is running z/VM must be allocated to expanded storage. As the system performance characteristics become known, this number can be adjusted. Once configured, it is unlikely that the amount of expanded storage configured would be decreased.

When sizing the Linux virtual machine for Oracle, the virtual machine should be configured with the minimum amount of memory needed. Linux will set up cache buffers with all memory allocated to it. It is better to let z/VM handle paging and manage memory where possible. System performance can be adversely affected by the amount of memory allocated. This could be caused by the 2 GB line issued as outlined above.

The method to size a guest for Oracle is to add the Systems Global Area, the Program Global Area (if needed), and about 128 MB for Linux.

- The Systems Global Area is the memory requirement for Oracle. This is shared memory within Oracle to functions such as caching table data, parsing SQl, sorting, etc.

- The Program Global Area is memory outside the Oracle instance. This generally used by users who create connections to the database. Depending on the application, this can be either a small amount of memory per user or very large amounts (in excess of 10 MB per user) for applications like the Oracle E-Business Suite.

This tends to be the complete opposite of assigning memory to Linux in an Intel environment or even in UNIX. But it must be remembered that in this case, Linux is running under and being managed by another operating system.

2.2 CPU allocation

CPU (or CP) allocation can be viewed in a few perspectives:

- A MIPS requirement for the databases and applications to be run in Linux on zSeries
- Allocation of virtual CPs
- Setting shares or running multiple LPARS with CPUs either as shared or dedicated

2.2.1 Sizing

As part of the consolidation process, workloads should be sized not only to insure the correct amount of MIPS are determined, but also to aid in determining if this is a good candidate to move to Linux on zSeries. It can also help determine if this database should possibly be run in a Linux guest in an LPAR and not under z/VM with the development, test, etc. databases being run under z/VM. This is a very important part of the consolidation process, if not the most important part. Doing the right work here will save problems later after the workload is moved to Linux on zSeries.

There are several rules of thumb to do a sizing. While they may provide a quick number, none take into account the characteristics of the workload. These workload characteristics are extremely important in assessing MIPS requirements.

IBM can provide sizing estimates, and there is no charge for this. To do an estimate, IBM will need to following information:

- Make and model of the system the database currently runs on
- The number of CPs and the MHz rating of the CPs
- The peak utilization of each system

It is best if these are actual numbers from a system tool such as vmstat. Approximations can be provided, but the amount of utilization they are in error is directly proportional to the error in the MIPS estimate.

There are two methods of sizing: The Quick Sizer tool and Size390 team. The Quick Sizer tool takes your estimated data as input and provides a MIPS estimate. The Size390 team uses vmstat reports from peak periods to perform a sizing estimate. The Size390 team can also take vmstat reports taken through peak periods and perform a sizing estimate from that. If a Quick Sizer is done and the numbers warrant proceeding, then working with the Size390 team is the best approach.

2.2.2 CP allocation

The amount of virtual CPs allocated (using the CP **`define cpu`** command) to a Linux machine is important. In general, one virtual CP is a good starting point unless the sizing points to something much larger.

In a constrained CPU environment (that is, more CPU cycles are needed to complete the work), you should give the Linux virtual machine all the CPs it needs to do its work efficiently. However, there should not be more CPs allocated than the number of physical CPUs that are assigned to this LPAR when running under z/VM. If this were to happen (more virtual CPs than physical CPUs), the transaction rates would decrease and the cost to execute these transactions would increase. This is due to increased scheduling overhead in z/VM's control program.

CPU time is limited to the number of CPUs installed. Therefore it would be prudent to reduce CPU usage where possible. This would be in areas such as:

- Eliminate any unnecessary services that might install with the Linux guest.
- Eliminate any unnecessary cron tasks.
- Reduce unnecessary work.
  - Ensure timer patch is turned on and timer pops are disabled. This is the default with SLES8.
  - Eliminate using "r-u-there" pings to determine if the virtual machines are there and up.
  - Do not measure idle guests. Measuring takes cycles.
- Network consideration
  - Use Guest Lan
  - Use Vswitch

2.2.3 Setting shares

The workload of Linux virtual machines can be managed or controlled by setting of the SHARE value of a virtual machine. The first choice of SHARE is to use ABSOLUTE or RELATIVE. The second choice is the size of the SHARE. The simplest way to decide which to use for a specific server is to determine whether as more users log on to this system, should this service machine get more CPU or less CPU?

A relative share says this server should get a relative share of the processor, relative to all virtual machines in the dispatch and eligible lists. As more users log on, the share will drop.

An absolute shares remain fixed up to the point where the sum of the absolute shares is 100 percent or more, a rather confused state of configuration. Servers might have a requirement that increases as the level of work increases. These servers should have ABSOLUTE shares. All other users should use RELATIVE.

Size of share is both a business decision and a performance decision. For example, if one server is assigned a very high share, which might be as much of the system as the rest combined, one would expect this server to be absolutely critical to your business. This server has the capability of taking resources whenever it needs them, but if this server starts looping, it would easily consume all the resources allocated. But in general, this is not a likely situation for production.

A simple way of looking at share values is that if there is a heavy contention for the processor, what servers would you like to run? The production Oracle databases are an obvious choice. Required servers should have absolute values

Archived

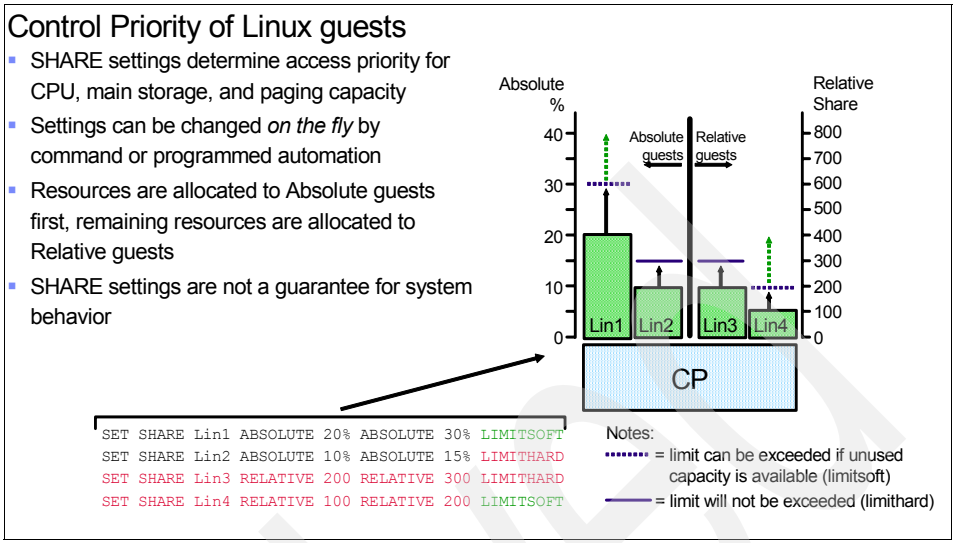


Figure 2-1 Using shares to manage guest priorities

2.3 Paging, swap, and I/O considerations

z/VM uses paging, while Linux primarily uses swapping but performs some paging as well. The term swap is a carry over from early days of Linux when it did swap an address space to reclaim memory if needed. However, today Linux does page and uses its own paging algorithm. This means that double paging can occur and this is not a new concept in z/VM.

One case may be where z/VM needs to page out and selects a page from the LRU. Hopefully the system is properly configured and the paging is to expanded storage. Now a Linux virtual machine needs to page and looks at its LRU to page. Most likely this could be the page that z/VM just paged out. If we assume this is the case, a page fault now occurs and the page must be brought back into central storage. Now Linux will complete its paging by moving the page to its swap space or paging device. In this case the system paged so that a system *guest* can page.

It is best to keep this type of activity to a minimum. The approach to this is to keep the memory in Linux virtual machines small enough so that z/VM does not do a lot of paging and large enough so that Linux does not have to do excessive paging. And all paging at the first level should be to a memory device such as a VDISK for Linux or expanded storage for z/VM.

2.3.1 z/VM paging

One of the common mistakes with new VM customers is to not allocate paging space (along with not configuring expanded storage). The installation process configures enough paging space to complete an installation. This paging space on the sysres pack is small and can handle a small amount of tasks. However, you should remove the paging space from the sysres pack and add DASD page space to do real work. The VM Planning and Administration manual has details on determining how much space is required. Here are a few thoughts:

- If the system is not paging, you may not care where you put the page space. However, it has been our experience that sooner or later the system grows to a point where it pages and then you will wish you had thought about it.
- z/VM paging is most optimal when it has large contiguous available space on volumes that are dedicated to paging. Therefore, do not mix page space with other space (user, tdisk, spool, Linux guests, etc.).
- Set up VM paging to many disks. The more subchannels z/VM can use the faster the paging will take place.
- A rough starting point for page allocation is to add up the virtual machine sizes of virtual servers running and multiply by 2. Keep an eye on the allocation percentage and the block read set size.

2.3.2 Swap space for Linux

Try to avoid swapping in Linux whenever possible. It adds pathlength and significant hit to response time. However, sometimes swapping is unavoidable. If you have to swap, these are some of your best choices.

- Dedicated volume - If the storage load on your Linux guest is large, the guest might need a lot of room for swap. One way to accomplish this is simply to ATTACH or DEDICATE an entire volume to Linux for swapping. If you have the DASD to spare, this can be a simple and effective approach.
- Traditional minidisk - Using a traditional minidisk on physical DASD requires some setup and formatting the first time and whenever changes in size of swap space are required. However, the storage burden on z/VM to support minidisk I/O is small, the controllers are well-cached, and I/O performance is generally very good. If you use a traditional minidisk, you should disable z/VM Minidisk Cache (MDC) for that minidisk (use the MINIOPT NOMDC statement in the user directory).
- VM VDISK - A VM virtual disk in storage (VDISK) is transient like a t-disk is. However, VDISK is backed by a memory address space instead of by real DASD. While in use, VDISK blocks reside in central storage (which make them very fast). When not in use, VDISK blocks can be paged out to

expanded storage or paging DASD. The use of VDISK for swapping is sufficiently complex that we have written a separate tips page for it.

Linux assigns priorities to swap extents. So, for example, you could set up a small VDISK with higher priority (higher numeric value), and it would be selected for swap as long as there was space on the VDISK to contain the process being swapped. Swap extents of equal priority are used in round-robin fashion. Equal prioritization can be used to spread swap I/O across chpids and controllers, but if you are doing this, be careful not to put all the swap extents on minidisks on the same physical DASD volume, for if you do, you will not be accomplishing any spreading. Use **`swapon -p`** ... to set swap extent priorities.

Setting up two VDISKS of different priorities or a VIDSK and a minidisk with the higher priority on the VDISK can provide a tuning toll or method for properly sizing your memory needs on the Linux virtual machine. Monitor the paging at peak times. If there is paging to the second paging disk, then you should consider adding memory to Linux.

2.3.3 I/O considerations

One of the biggest performance issues we ran into during some testing, both on z/OS and Linux on zSeries, was when we placed the database on a single rank in the ESS800. This will cause many performance-related issues that mostly involve cache utilization in the ESS800. The best practice for the ESS system is to distribute data across as many ranks (arrays) as possible. This made a dramatic difference in our testing, even though this was a small database.

It is also important to stripe the logical volumes that are created. This enables the operating system to issue multiple I/Os against the LVM. Striping a logical volume is not the same as any striping or RAID that is done inside the disk controller. The guidelines for striping on a IBM Total Storage ESS and DS8000 are as follows;

- ESCON® - One stripe per channel
- FICON® - One stripe per physical volume
- FCP - One stripe per volume or LUN

When the logical volume is created one of the parameters is the stripe size. The best stripe size is dependent on the workload characteristics. The IBM Web site recommends 16 K or 32 K stripe sizes. In testing typical OLTP type applications either 4 K or 8 K worked better. Applications such as Samba that may serve large files do better with a 64 K stripe. So the message here is that if you do not already know the optimal stripe size for your application, it may take some testing to determine what that size might be.

A final thought on this is that faster and newer is always better. Some implementations have experienced up to a 50 percent performance improvement just going from ESCON to FICON.

Parallel Access Volumes can be used with z/VM and Linux. This lets the ESS800 write from parallel processes to the same device. The writes will occur in different domains within the volume. This feature is only available on the IBM ESS800.

2.3.4 Monitoring the system

z/VM provides extensive virtualization techniques. A virtual machine running under z/VM acts as if it has all the resources allocated to it in the user direct profile that was created for the virtual machine. This means that monitoring what a virtual machine is doing by looking at resource utilization with a tool or monitor running in that virtual machine may not always be accurate.

Since monitoring a virtual machine from the virtual machine is not always accurate, it is extremely important to use a tool to monitor the z/VM system and all the virtual machines from a system perspective. There are two tools that we have experince with and both provide value:

- IBM Performance Toolkit for VM - Licensed through IBM
- ESAMON from Velocity Software (`http://www.velocity-software.com`)

While using one of these tools is absolutely necessary for performance monitoring and tuning, do not overlook using tools like **`sar`** to **`vmstat`** monitor events and resource utilization such as swapping in a Linux virtual machine.

Yet another way to monitor performance, especially during times when problems are suspected, is the use of CP commands such as **`cp indicate load`**. These are very valid to use for a current point in time but do not provide the ability to go back to review past performance, as does one of the tools mentioned above.

The purpose of this chapter is to describe the configuration of Linux and z/VM. However, this is for an Oracle database, and using a tool such as Statspack (or DB Control for 10g) is still an important way to monitor and tune the Oracle database server.

2.4 Summary

The following is a summary of the key points needed to get the best performance from Oracle.

- Memory - This is likely the most critical area of tuning Linux virtual machines. The mindset here is the exact opposite of Linux on Intel. The less memory the

better. Memory assigned to Linux should be enough to cover the needs of the Oracle database (SGA + PGA) and about 128 MBs more for Linux.

- Monitor resources - The more virtual machines installed under z/VM the more you will need to use a system monitor such as the z/VM Toolkit. Trying to understand performance problems and determine capacity needs without this type of monitor is impossible.
- Paging and swap space - Paging at the z/VM level and Linux level (swapping) is most practically unavoidable. You must allow for this. Expanded storage is an absolute must, as is swap space (paging device) on Linux. The best paging device for Linux is a VDISK.
- Avoid I/O bottlenecks - Use a best practice for configuring and using disk. Data must be distributed as well as possible across different ranks. The logical volumes should be striped, with the number of stripes being dependent on the technology used.

Archived

3

Installing Oracle 10g single instance

This chapter describes the steps we executed to install Oracle database 10g for Linux on zSeries in 4Q 2004.

The following topics are covered:

- Setting up the xWindows connection
- Preparing for the installation
- Running the Oracle Universal Installer
- Post-installation verification
- Setup on VM and Linux

© Copyright IBM Corp. 2005

3.1 Installing Oracle Database 10g on zSeries with Linux

Oracle Database 10g for Linux on zSeries is 64-bit and can be set up using a SLES8 64-bit Linux guest under VM in an LPAR on zSeries or directly on a LPAR running SLES8 64-bit Linux. Oracle 10g is also supported on SLES9, and there are plans to support it on RedHat 4.0 U1 in 3Q 2005.

The version released in October 2004 was Oracle Database 10g (10.1.0.3). This paper is based on the installation experiences gained when installing this code at IBM and Oracle locations.

This book assumes that you have a functional Linux image. It describes the steps we performed to prepare the environment and to run the installation process. In the attached appendix, we describe how we set up our VM and Linux guests.

3.2 Preparing the system environment

We did our testing on Linux guests on IBM z990 systems at IBM sites in Poughkeepsie and Montpellier, along with a Linux guest on a z990 at Oracle Headquarters in Redwood Shores, California. This book is based on a Linux guest named LINUX1. The file system we used was a 7 GB logical file mounted as /oracle.

The Linux environment

Have your systems administrator set up the Linux guest with the resources you need such as CPU, memory, disk, and network connectivity. Have SLES8 Linux installed with at least kernel level 2.4.21-251.

> **Note:** Oracle certified on the 112 level, but we recommend using the latest level, at least 251 or higher.
>
> SLES9 should at least be at kernel level 2.6.4-151.

Definitions used for VM

This is the Linux definition we used under VM in Montpellier:

The setup is SuSE 2.4.21-251 guest Linux system (64-bit), running under z/VM 4.4 on a z/900 LPAR:

```
Storage size is 1 GB
IP address is 9.x.x.x
System name is LINUX1
```

```
 Root password is root1
Each system has these file spaces:
Filesystem Size Used Avail Use% Mounted on
/dev/dasdb1 2.1G 73M 1.9G 4% /
/dev/dasdc1 2.3G 2.0G 153M 93% /usr
/dev/dasdd1 2.3G 2.0G 222M 90% /opt
/dev/grp1/vol1 6.8G 20k 6.4G 1% /oracle this is 3 3390-3 vols in LVM
/dev/dasdh1 2.3G 20k 2.1G 1% /oracle2
```

You can have multiple ORACLE_HOMEs or multiple databases in a single home in one guest. We recommend only having one Oracle production instance per VM/Linux guest. While not absolutely necessary, we also suggest that this same strategy be adopted if running development and test databases under z/VM. Running multiple homes or multiple databases in an LPAR can be implemented assuming you assign the right resources to provide the expected performance.

Setting up the group and user ID for Oracle

First, create two groups, DBA and OINSTALL, by using the YAST command. Then create a user ID called oracle with the primary group as DBA.

Ensure the user ID oracle can write to the /oracle directory. We created the mount point as part of the installation as root. As our directory had been created by the user ID root, we had to logon as root and issue the command:

```
chown oracle:dba oracle
```

If you have already created directories under this and want to change the ownership to oracle, you can use the -R switch to it recursively.

3.2.1 Setting up the fixed buffer option with ECKD™ disks

If you are using ECKD disks on SLES9 or RedHat 4.0, make sure you have the Fixed Buffer Option enabled. If you are using FCP, this does not apply.

Information on this can be found at:

http://awlinux1.alphaworks.ibm.com/developerworks/linux390/perf/tuning_rec_fixed_io_buffers.shtml#begin

And at:

http://awlinux1.alphaworks.ibm.com/developerworks/linux390/perf/tuning_how_fixed_io_buffers.shtml#begin

The second Web site is linked from the first at the bottom of the page.

Note: As of 3Q 2005, the Fixed Buffer Option is presently being back-ported to SLES8.

For example, in our testing we:

1. Added the fixed buffer parameter in /etc/zipl.conf in SLES9.

```
parameters = dasd=200-20f,fixedbuffers root=/dev/dasda1 elevator=deadline
selinux=0
       TERM=dumb
```

2. Added the fixed buffer option in /etc/modprobe.conf in RedHat 4.0.

```
options dasd_mod dasd=200-20f,fixed buffers

and /etc/zipl.conf   for other module options
e.g. parameters='root=LABEL=/'
```

   And in addition executed the command:

```
mkinitrd
```

3.3 Setting up an xWindows interface using VNC

The Oracle Universal Installer (OUI) requires that you have xWindows interface. To enable xWindows we used the VNC server that comes with SuSE Linux. We downloaded the VNC viewer from the following Web site:

http://www.realvnc.com/download.html

You can download either a Linux or Windows® version, depending on the client that you chose for the installation. Once you have a VNC client installed you can follow the next set of steps to set up the viewer.

Using PuTTY, log on using the secure shell. We logged in to the Linux guest as root. To start the vnc server, we issued the command;

```
vncserver
```

This will start the vnc server for the root user. You will be asked for a password and to verify the password you entered. This will start a session in a format known as the tiny window manager. We chose to use the motif window manager, which is a bit easier to use. This is done by executing the following steps;

1. **`vi /root/.vnc/xstartup`**
   a. Go to the line with twm and place the cursor under the letter `t`.
   b. Type `r` (to change character).

c. Type the letter `m`.

d. Press the Esc key

e. Type `:wq` and `enter`.

f. Now kill the session you started by entering `vncserver -kill :1` (or the session number you were given when you started).

g. Start the vncserver with **`vncserver`**. It will start with the same session as before and retain your password.

Note: The letter m option may not be available in all Linux distributions.

You can run multiple VNC sessions on multiple user IDs (not multiple sessions on a single user ID though). You can perform this procedure to use the motif viewer for oracle or any other user ID. The .vnc directory and associated files are located in /home/*userid*/.vnc.

Note that different users will have different ports. In our case root was:

```
9.12.4.53:1
```

The oracle user ID was:

9.12.4.53:2

Later in the process, when we create the oracle ID, we use two VNC sessions, one for oracle and one for root. This seems simpler to us than switching users (**`su`**) within the same session.

Once we started the session, we did a right-click. This presented a work menu from which we selected the X-Terminal to open a second window for the user ID *oracle*.

Archived

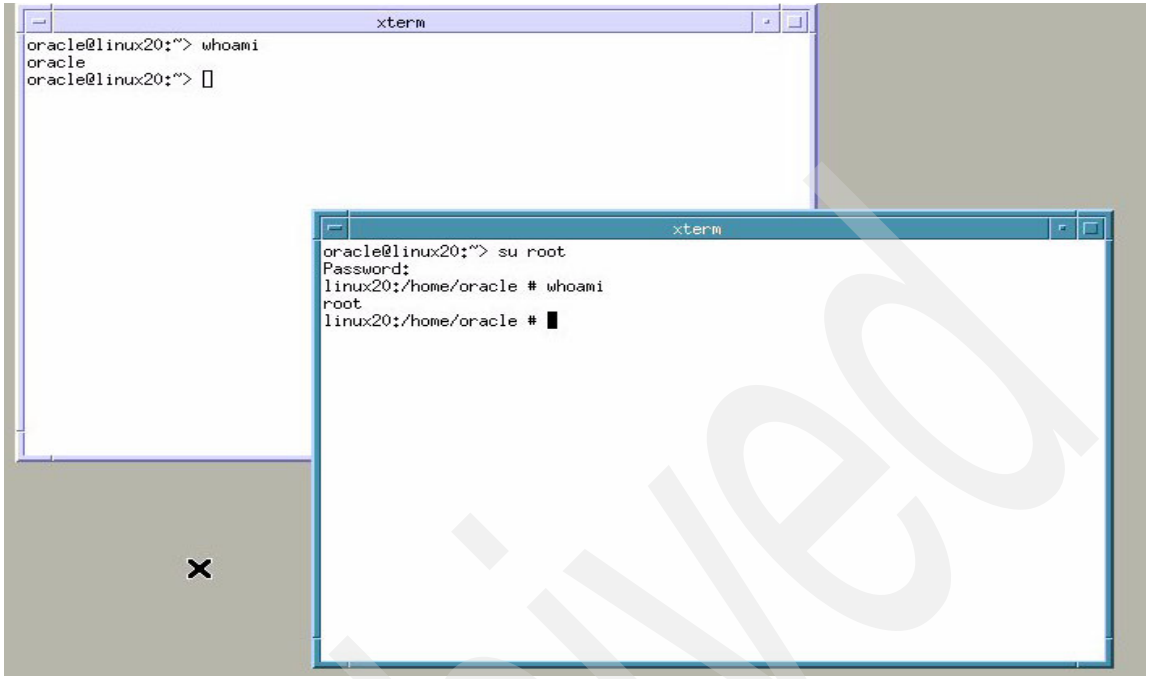


Figure 3-1 VNC Viewer with a window for oracle and root

3.4 Downloading the code

In the following steps, we logged into our guest as oracle.

The steps to download the code are:

1. Go to:

   http://www.otn.oracle.com

2. Choose **Download** on the right.
3. Choose **Oracle Database 10g**.
4. Choose Oracle database 10g for Linux on zSeries NEW.
5. You may have to complete a series of questions about licensing to proceed.

   Once the code is downloaded, go to the directory where you want to store the images and complete the following commands:

   - **cd /oracle**
   - **mkdir images**
   - **cd images**

Archived

– FTP to the Linux guest

We received the following files:

```
-rw-------    1 oracle dba 557448320 Oct 26 17:27 Linux_ship.ccd.cpio.gz
-rw-------    1 oracle dba 514335522 Oct 26 16:16 Linux_ship.client.cpio.gz
-rw-------    1 oracle dba 579717912 Oct 26 15:13
Linux_ship.db.disk1.cpio.gz
-rw-------    1 oracle dba 178013002 Oct 26 15:40
Linux_ship.db.disk2.cpio.gz
-rw-------    1 oracle dba 237219293 Oct 26 15:40 Linux_ship.crs.cpio.gz
```

The content of the files are:

– Linux_ship_ccd.cpio.gz: The companion CD
– Linux_ship_client.cpio.gz: Client code such as the precompilers
– Linux_ship_crs.cpio.gz: The cluster-ready services software for RAC
– Linux_ship_db.disk*n*.cpio.gz: The two CDs of code to create database

The **`ls -la`** command was used to check that we received the same size files as on the host.

6. Uncompress the database file in the /oracle/images directory:

```
gunzip Linux_ship.db.disk1.cpio.gz
gunzip Linux_ship.db.disk2.cpio.gz
mkdir db
cd db
cpio -idcmv<../Linux_ship.db.disk1.cpio
cpio -idcmv<../Linux_ship.db.disk2.cpio
```

These are large files. The **`ftp`** and **`cpio`** commands can take 10 to 30 minutes to complete.

The reason we created the db directory is to save the files. Otherwise, if we were to run another **`cpio`** command on the other cpio file such as CRS, Client, or CCD, that cpio would overlay the previous one and you would no longer have the files for the db. We ended up with the following directories under /oracle/images:

```
db
crs
ccd
client
```

3.4.1 Finding the documentation

Documentation is found at:

http://www.otn.oracle.com

Click **Documentation** at the bottom of the page on the right side, then click **Database**.

Make sure you obtain the release notes for Oracle Database 10g for Linux on zSeries—*Oracle Database Release Notes for Linux on zSeries,* B13964.

For Oracle Database 10g, we used the following generic documentation:

- *Oracle Database Installation Guide for UNIX Systems*, B10811
- *Oracle Database Administrator's Reference for UNIX Systems*, B10812

The release notes contain important information not included in the generic documentation.

3.4.2 Checking the Linux kernel settings

Check the kernel parameters as per Section 2 of the installation guide beginning on page 2-44. The most important parameter is shmmax. This limits the largest shared memory segment allowed and therefore can limit the amount of memory for Oracle and the Systems Global Area (SGA). It can be set as large as you wish, even beyond the amount of memory on the system. But if you set it too small, then Oracle will not start. Follow these steps:

1. Log on as root.
2. **`cd /proc/sys/kernel`**
3. **`cat sem`**

   Our entry was 250 2560000 32 1024.
4. **`cat shmmax`**

   Our entry was 33554432.
5. **`cat shmmni`**

   Our entry was 4096.

The Oracle doc recommends that the sem value be set to 100. In our case, the default was 32. It is the third parameter in the set when you issue a **`cat`** command such as **`cat /proc/sys/kernel/sem`**.

We increased the shmmax as well as semopm. For this, we created the file /etc/sysctl.conf and added only the following entries:

```
kernel.sem=250 32000 100 28
kernel.shmmax=2147483648
```

Then we executed the following commands. The first command activates the change, while the second one makes the changes permanent across boots:

```
/sbin/sysctl -p
/sbin/chkconfig boot.sysctl on
```

Important: If you plan to have a very large SGA, you may want to increase the size of shmmax. The database will not start if shmmax is smaller than the SGA.

3.5 Running the Universal Installer

Choose the names for ORACLE_HOME, ORACLE_SID, and so on, as shown in Table 3-1.

Table 3-1 Names chosen for Oracle installation

Variable	Value	Comment
ORACLE_BASE	/oradbf	Highest level directory for this Oracle installation
ORACLE_SID	ora1	Database name
ORACLE_HOME	/oradbf/o10g	Where the binaries are located
ORACLE_INVENTORY	/oradbf/oraInventory	Must not be in ORACLE_HOME but under ORACE_BASE
Images directory	/oradbf/images/	Where we put the cpio files
Oracle libraries	$ORACLE_HOME/lib:$ORACLE_HOME/rdbms/bin	
Oracle db files	/oradbf/oradata	

We chose the directories above. To facilitate the install, we put the following entries in the oracle .profile:

```
export ORACLE_BASE=/oradbf
export ORACLE_HOME=/oradbf/O10g
export ORACLE_INVENTORY=/oradbf/oraInventory
export PATH=$PATH:$ORACLE_HOME/bin
export LD_LIBRARY_PATH=$ORACLE_HOME/lib:$ORACLE_HOME/rdbms/bin
export ORACLE_SID=ora1
```

And then executed the profile by entering:

```
. .profile (that's dot space dotprofile).
```

Archived

Note: In one test, we used the orarun package from the SuSE Web site. This automated tool set all the kernel config values and set the .profile shell scripts.

3.5.1 Starting the OUI

We started the OUI with the oracle user. With the vncserver started from the oracle user ID, we started the vncviewer with 9.4.12.153:2, as seen in Figure 3-2, and then our password (Figure 3-3) to start the vnc session.

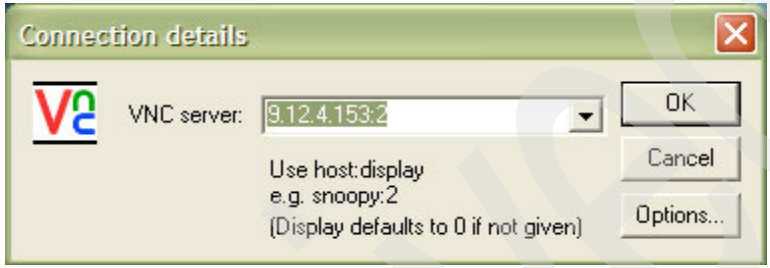


Figure 3-2 VNC server IP address

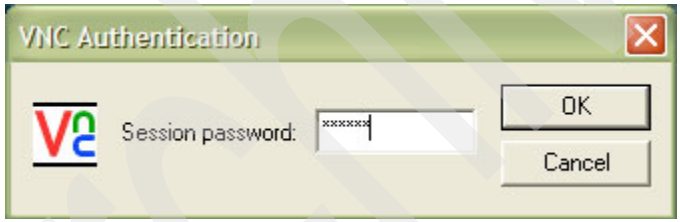


Figure 3-3 VNC password

If the vncserver is not started in oracle, then you must PUTTY into the user ID oracle and start it using the procedure described in “Setting up an xWindows interface using VNC” on page 24.

We also ensured that the directories listed above have the ownership of oracle. If you try this from user ID root, you will get the following message:

```
The user is root. Oracle Universal Installer cannot continue installation
if the user is root.
: No such file or directory
```

We went to the directory where the CD-ROM images were stored, then ran the installer as the oracle user:

```
cd /oradbf/images/Disk1
./runInstaller
```

This takes a few minutes. Then a screen appears with the name of the Oracle Universal Installer, followed by the Welcome panel. These messages will appear on the telnet window.

Start the Oracle Universal Installer:

```
No pre-requisite checks found in oraparam.ini, no system pre-requisite
checks will be executed.
Preparing to launch Oracle Universal Installer from
/tmp/OraInstall2004-07-19_04-28-20PM. Please wait ...Entering the
Code---------
After sprintf Code---------
Entering the Code---------
After sprintf Code---------
Entering the Code---------
After sprintf Code---------
```

Note: You should not have a problem starting the OUI if you have started the vncserver as user ID oracle. However, if you started vncserver as user ID root, you will have to set the DISPLAY command in your user ID oracle session and issue the **`xhost +`** command as user ID root.

This is because if you log on as a user that is not oracle and start the Xserver (vnc, cygwin, or any other) and then you use the **`su -l oracle`** command you will need to execute the **`xhost +`** command to allow access to the Xserver from the Oracle session so the display will show up. The reason for this is that the Xserver is owned by the user that started it and not by oracle.

The xhost program is used to add and delete host names or user names to the list allowed to make connections to the X server. There are several notes in metalink to further explain this.

The Oracle Universal Installer presents a series of panels that enable you to choose the appropriate options and to enter the information required for the installation process. After successfully starting the OUI, Figure 3-4 on page 32 appears.

3.5.2 Initial OUI panels

Here we review the initial OUI panels.

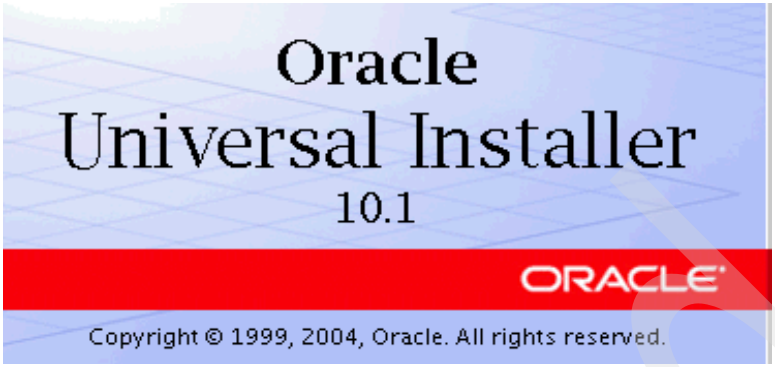


Figure 3-4 First panel

The first panel remains for several seconds, followed by the welcome panel.

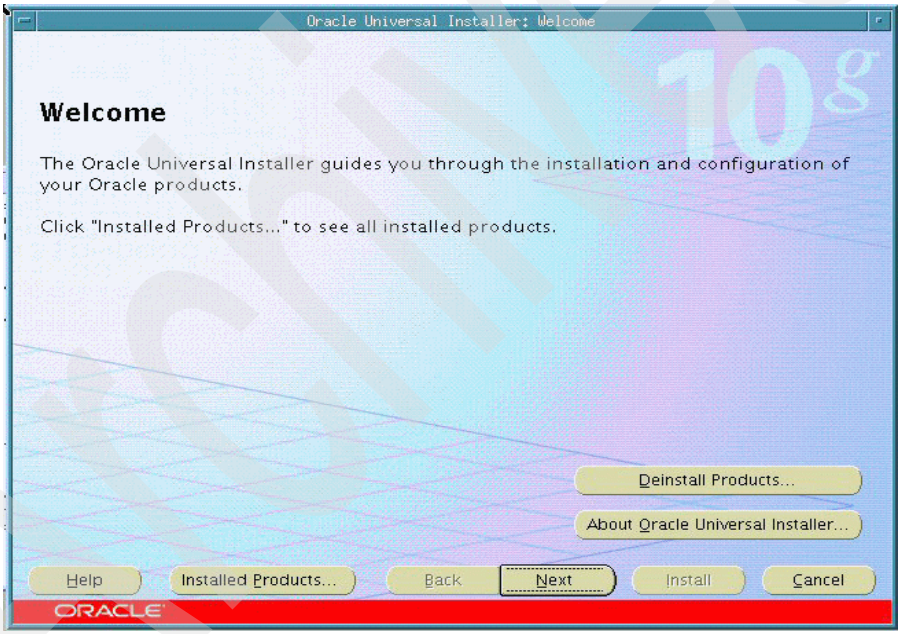


Figure 3-5 Welcome panel

3.5.3 Inventory directory panel

At this point you are asked to indicate your inventory location directory and the UNIX group name. You will create the Oracle Inventory file /var/opt/oracle/oraInst.loc. This is the default location. It is recommended that this location be changed.

Figure 3-6 Specify Inventory directory panel

Tab to the Specify Group Name and click to get the cursor at the beginning of the line. Enter `dba` (in lowercase).

If you have completed a previous Oracle installation and an inventory file already exists, then you will not receive this panel. The OUI will use the values already in the file. When you select **Next**, you will see Figure 3-7 on page 34.

3.5.4 Changing to root screen for oraInstroot script

Here we review the steps involved in changing to the root screen for the oraInstroot script.

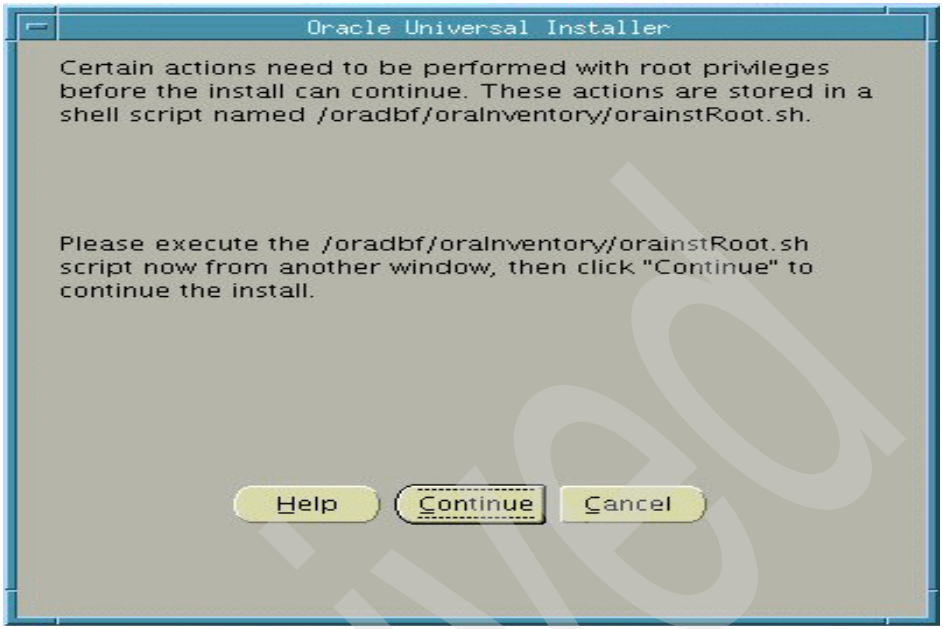


Figure 3-7 orainstRoot script

At this point, you have to go the window that is logged on as user root to complete the next step. We opened another window and executed the command:

```
su - root
```

Then we executed the command:

```
./oraninstRoot.sh .
```

You can see from the output in Figure 3-8 on page 35 that the owning group for the Oracle software is dba.

Archived

```
X Desktop
drwxr-xr-x    6 oracle   dba          176 2004-09-29 08:56 .
drwxr-xr-x   21 root     root        4096 2004-09-23 16:57 ..
drwxr-xr-x    2 oracle   dba           80 2004-09-29 09:43 images
-rwxr-xr-x    1 oracle   users     2625063 2004-08-03 11:19 swingbench21f.zip
drwxrwxrwx    2 oracle   users        448 2004-09-10 16:32 transport
linux20:/oradbf # cd imagesw
bash: cd: imagesw: No such file or directory
linux20:/oradbf # cd images
linux20:/oradbf/images # ls
.  ..  ship.db.cpio
linux20:/oradbf/images # ls -l
total 732831
drwxr-xr-x    2 oracle   dba           80 2004-09-29 09:43 .
drwxr-xr-x    6 oracle   dba          176 2004-09-29 08:56 ..
-rw-------    1 oracle   dba     749686272 2004-09-29 09:25 ship.db.cpio
linux20:/oradbf/images # whoami
root
linux20:/oradbf/images # cd ../oraInventory
linux20:/oradbf/oraInventory # ls
.  ..  orainstRoot.sh
linux20:/oradbf/oraInventory # ./orainstRoot.sh
Creating the Oracle inventory pointer file (/var/opt/oracle/oraInst.loc)
Changing groupname of /oradbf/oraInventory to dba.
linux20:/oradbf/oraInventory #
```

Figure 3-8 Results from running orainstRoot script

Selecting **Continue** again takes you to the following screen.

3.5.5 File location panel

Figure 3-9 on page 36 shows the screen that enables you to specify your ORACLE_HOME and gives you the ability to change its name as well. Our ORACLE_HOME is /oradbf/O10g. This is the value we placed in the oracle user ID profile earlier.

Archived

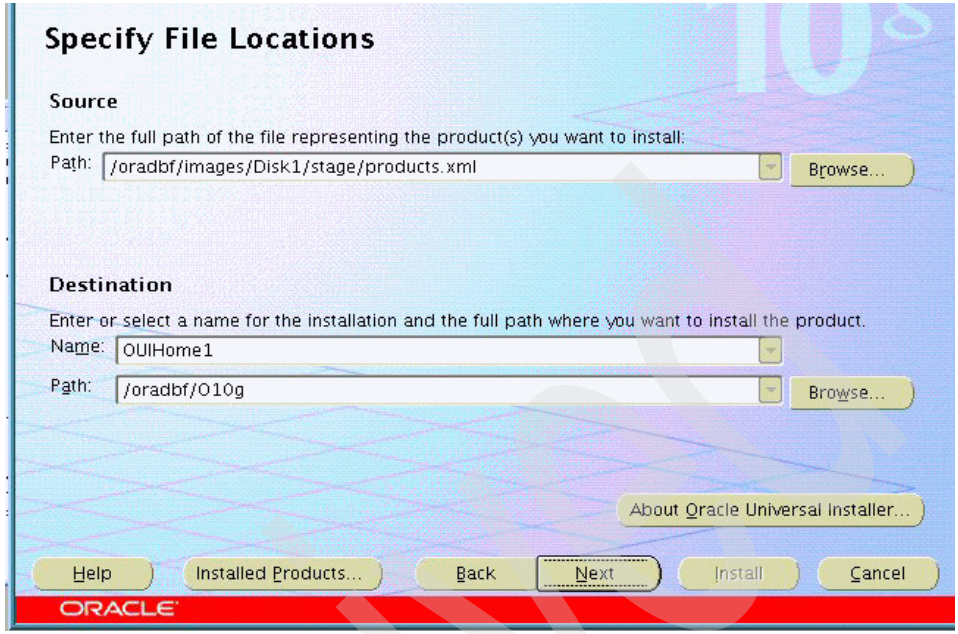


Figure 3-9 Specify FIle Locations panel

Ensure that oracle has write permission to the /oradbf directory (or whatever you used) or the installation will end here. If, for some reason, this belongs to root, you will need to **chown** (change ownership of) the directory to oracle.

3.5.6 Installation type

We chose to install the Enterprise Edition version of the database; see Figure 3-11 on page 37.

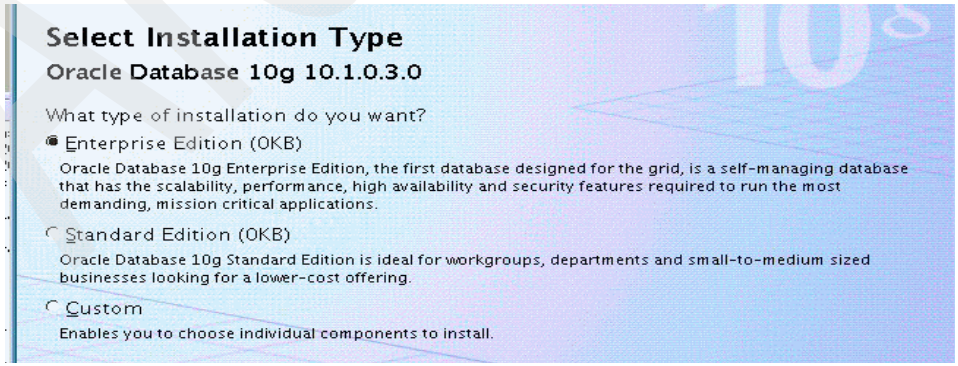


Figure 3-10 Specify Installation Type panel

3.5.7 Selecting database configuration

We chose to do a general purpose install. This installs the binaries and creates a database.

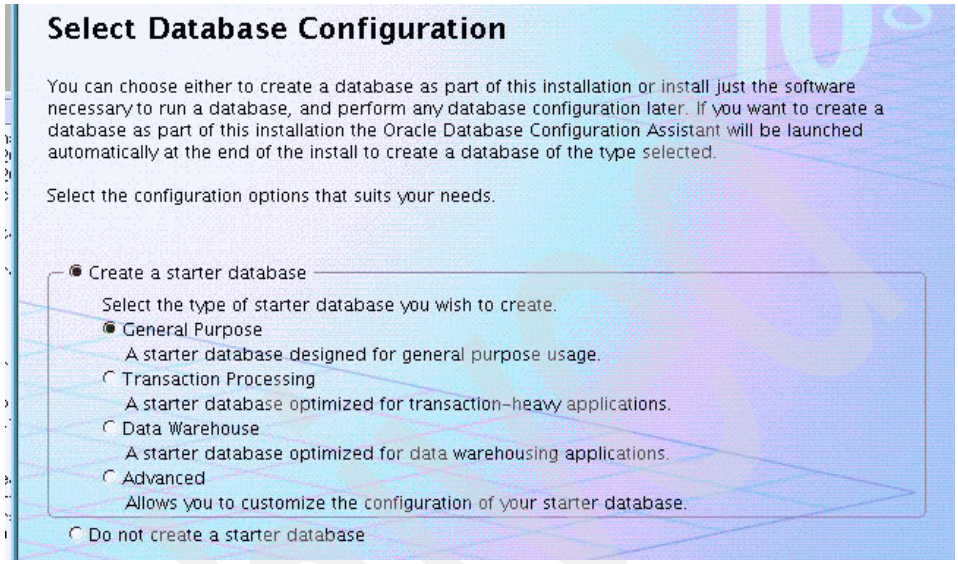


Figure 3-11 Select Database Configuration panel

The next several panels ask you to provide information about the database that will be created.

3.5.8 Database configuration options

We created the database ora1. We used the unicode standard character set UTF-8 AL32UTF8 and checked the box to create the sample schemas for our database; see Figure 3-12 on page 38.

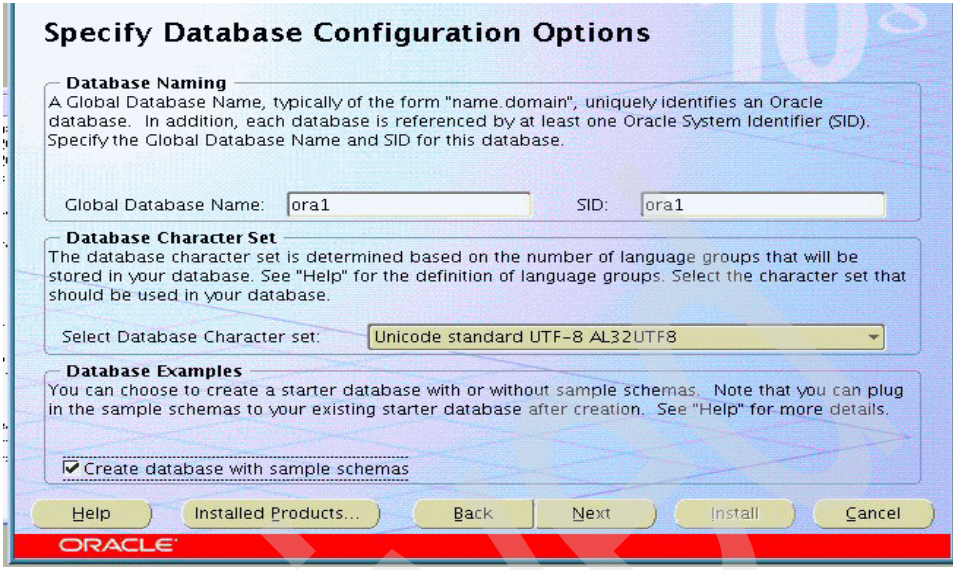


Figure 3-12 Specify Database Configuration Options panel

The OUI creates a database with an SGA that is 40 percent of the size of your Linux guest. Running the OUI in a 1 GB guest will give you an SGA of 400 MB. You can change this by modifying the init.ora parameters later.

3.5.9 Selecting the database management options

We used the defaults shown in Figure 3-13 on page 39.

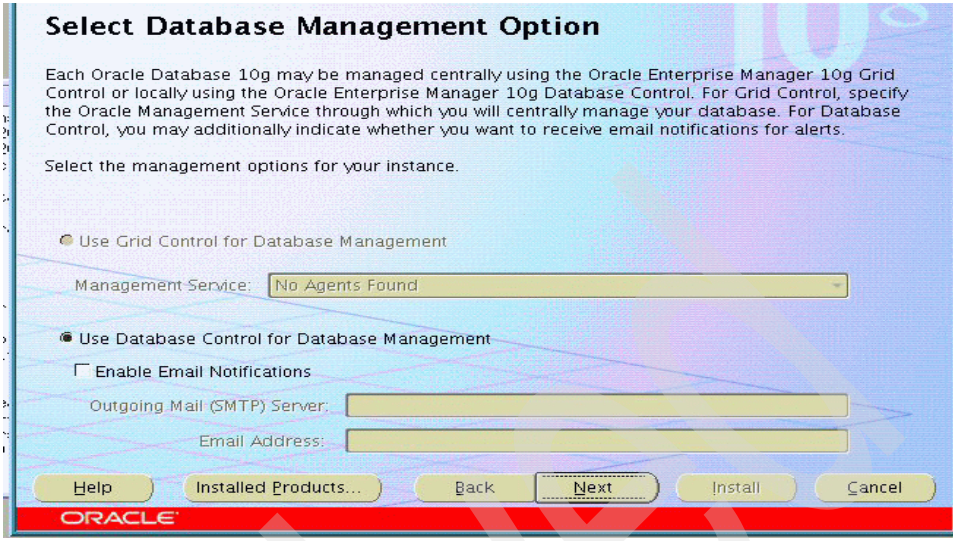


Figure 3-13 Select Database Management Option panel

3.5.10 Selecting the database file storage

We used the defaults shown in Figure 3-14 on page 40. The location for the data files is /oradbf/oradata/ora1. A new directory is now created with the name of the oracle SID for each database.

Archived

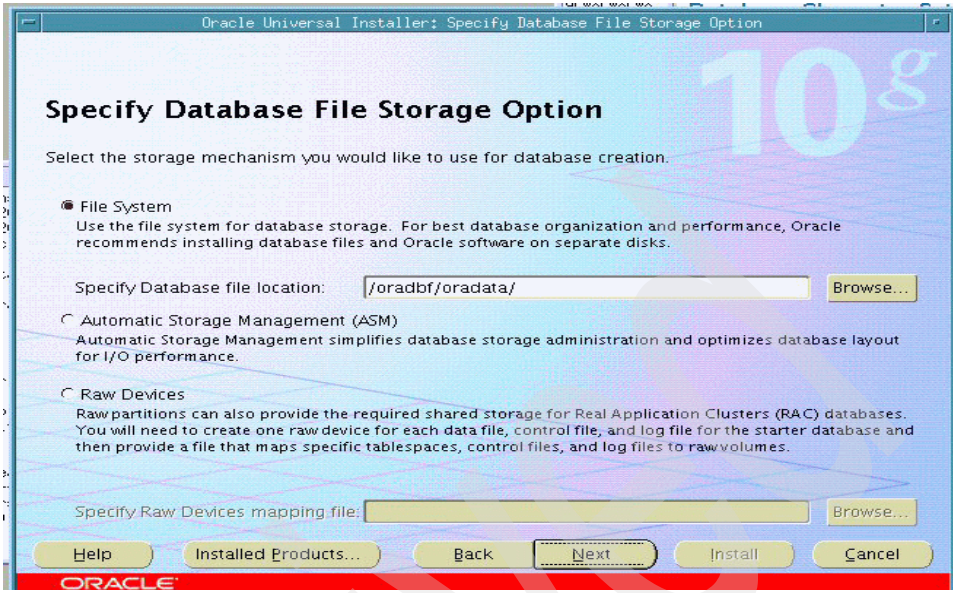


Figure 3-14 Select Database File Storage Option panel

3.5.11 Selecting the backup and recovery options

In Figure 3-15 on page 41, we chose **Do not enable automatic backups**, the default value.

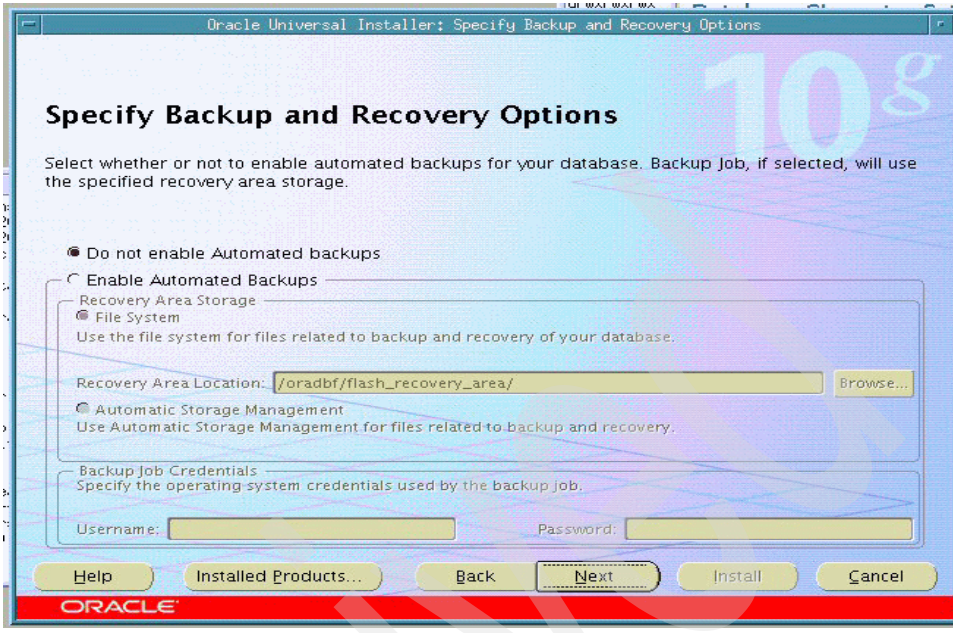


Figure 3-15 Select Backup and Recovery Options panel

3.5.12 Choose the database passwords

You must choose the passwords for your database instance. Since this is not a production environment, we chose to use the same password for all database schema passwords.

Archived

Figure 3-16 Specify Database Schema Passwords panel

3.5.13 Summary

At the end of the series of questions, you will receive a summary of your choices to review.

Note the space requirements section in Figure 3-17 on page 43. If you do not have sufficient space, it will prompt you to resolve this before proceeding.

Figure 3-17 Summary panel

When you are ready to proceed, click the **Install** button. See Figure 3-18 on page 44.

> **Note:** If there is a line in red, you could have a error message about not enough space. In one case, we had to remove old files from /tmp to be able to continue.

3.5.14 Install completing

Here we address the completion of the install.

Archived

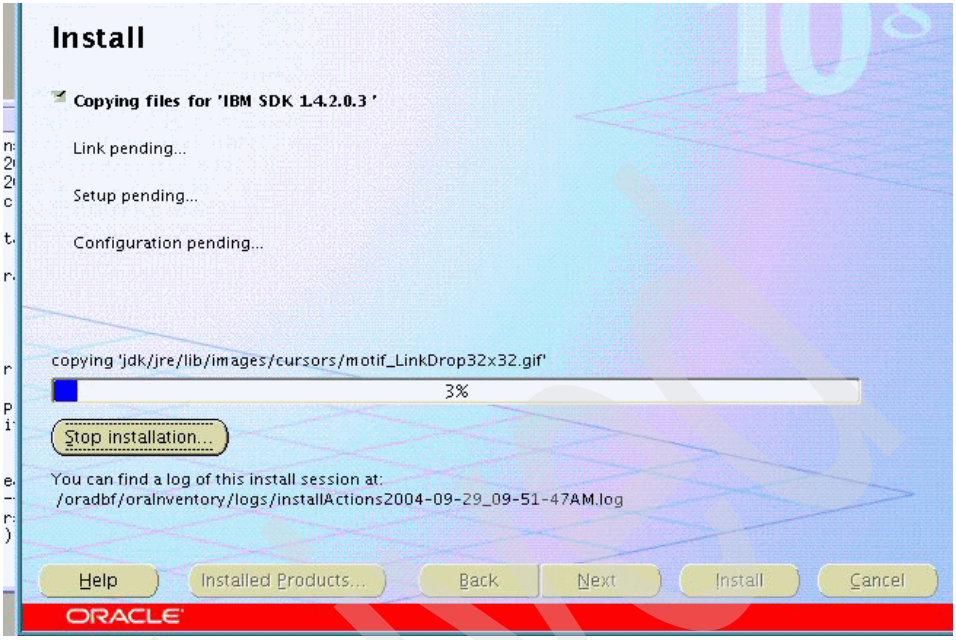


Figure 3-18 Installation progress panel

This part of the installation takes a while, as it is placing all the components of the database in the libraries.

During the remainder of the installation and creation of the database, the following panels will appear that provide the progress.

3.5.15 Configuration Assistant panel

Figure 3-19 on page 45 appears when the OUI is ready to configure the database.

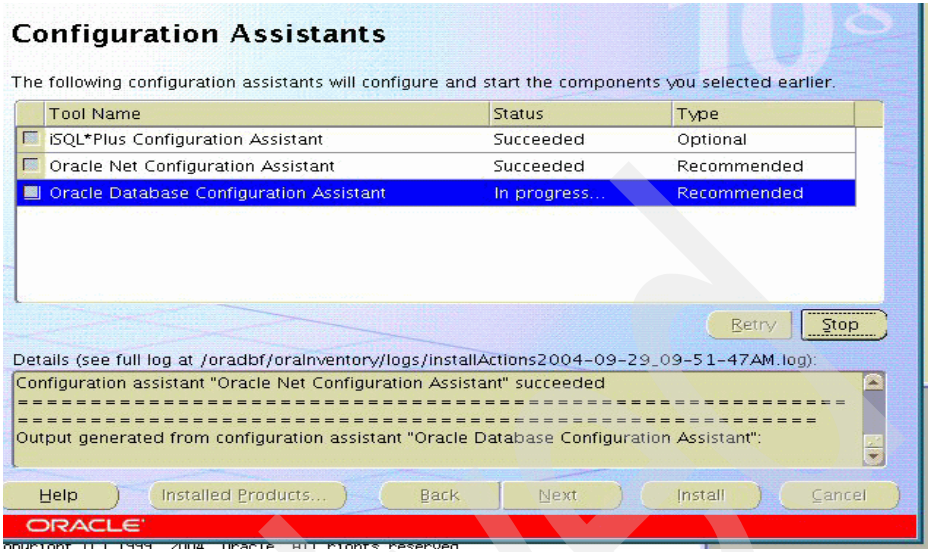


Figure 3-19 Configuration Assistants panel

If you chose only to install the Oracle code and not to create a database, you will go to the end of installation screen, as shown in Figure 3-25 on page 49.

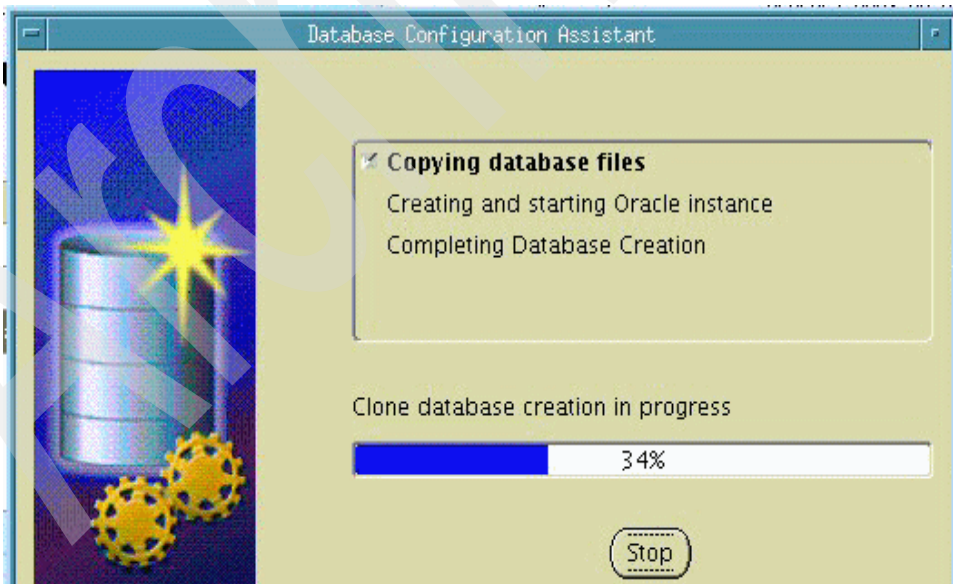


Figure 3-20 Database creation panel

When the installation process and the database creation are complete, Figure 3-21 appears. You will again be asked to go to a window where the user ID root is logged on to complete the next step.

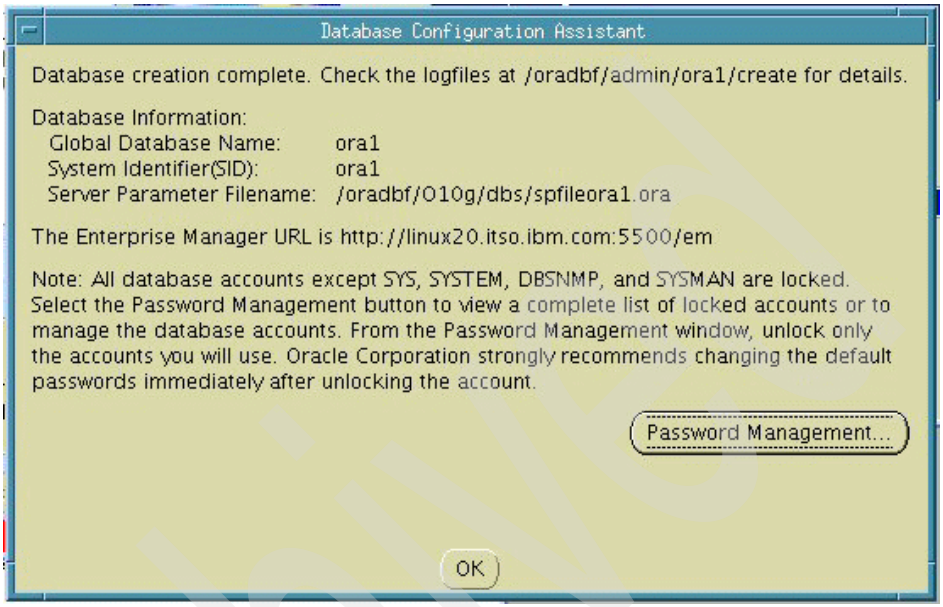


Figure 3-21 Password management panel

At this point we selected the **Password Management** button to unlock the Scott ID, as we used it later for other testing. After unlocking the account, the password for Scott will expire when the ID logs on. At that point you can enter the old password of tiger and use tiger as the new password as well.

Archived

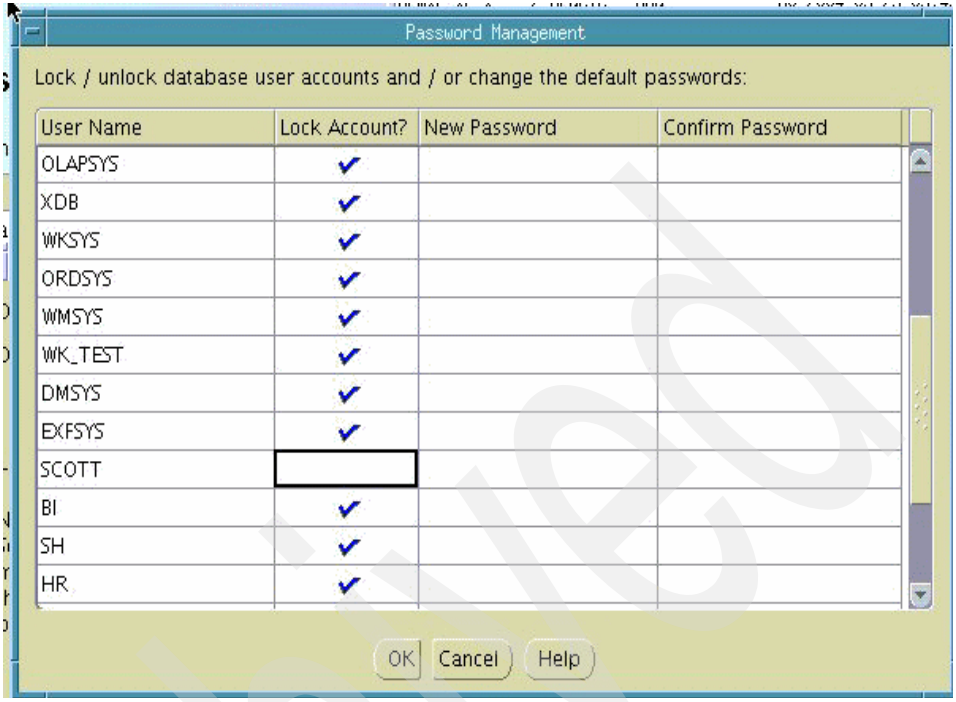


Figure 3-22 Unlock Scott panel

When the installation process and the database creation is complete, Figure 3-23 appears. You will again be asked to go to a window where the user ID root is logged on to complete the step.

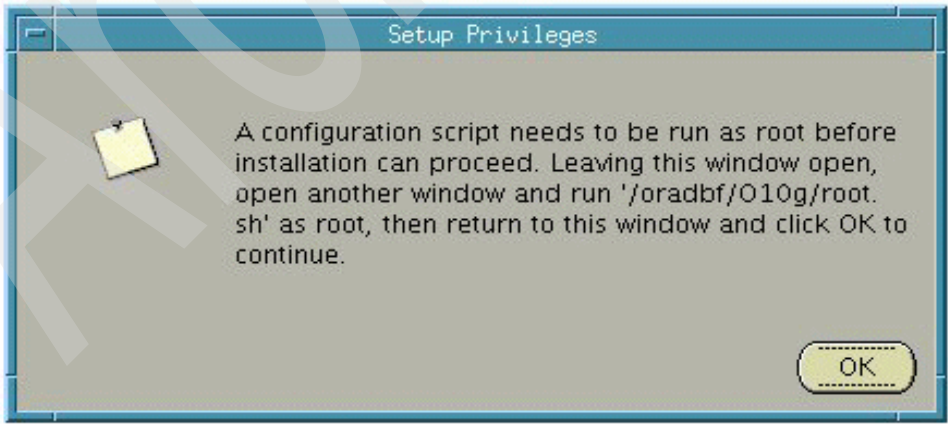


Figure 3-23 Setup Privileges panel

Archived

We went to the window with the root ID and executed the root.sh script. We received the messages in Figure 3-24.

```
xterm
linux20:/oradbf/O10g # ./root.sh
Running Oracle10 root.sh script...
\nThe following environment variables are set as:
    ORACLE_OWNER= oracle
    ORACLE_HOME=  /oradbf/O10g

Enter the full pathname of the local bin directory: [/usr/local/bin]:
   Copying dbhome to /usr/local/bin ...
   Copying oraenv to /usr/local/bin ...
   Copying coraenv to /usr/local/bin ...

\nCreating /etc/oratab file...
Adding entry to /etc/oratab file...
Entries will be added to the /etc/oratab file as needed by
Database Configuration Assistant when a database is created
Finished running generic part of root.sh script.
Now product-specific root actions will be performed.
/etc/oracle does not exist. Creating it now.
Successfully accumulated necessary OCR keys.
Creating OCR keys for user 'root', privgrp 'root'..
Operation successful.
Oracle Cluster Registry for cluster has been initialized

Adding to inittab
Checking the status of Oracle init process...
Expecting the CRS daemons to be up within 600 seconds.
CSS is active on these nodes.
        linux20
CSS is active on all nodes.
Oracle CSS service is installed and running under init(1M)
linux20:/oradbf/O10g #
```

Figure 3-24 Results of running root.sh script

Archived

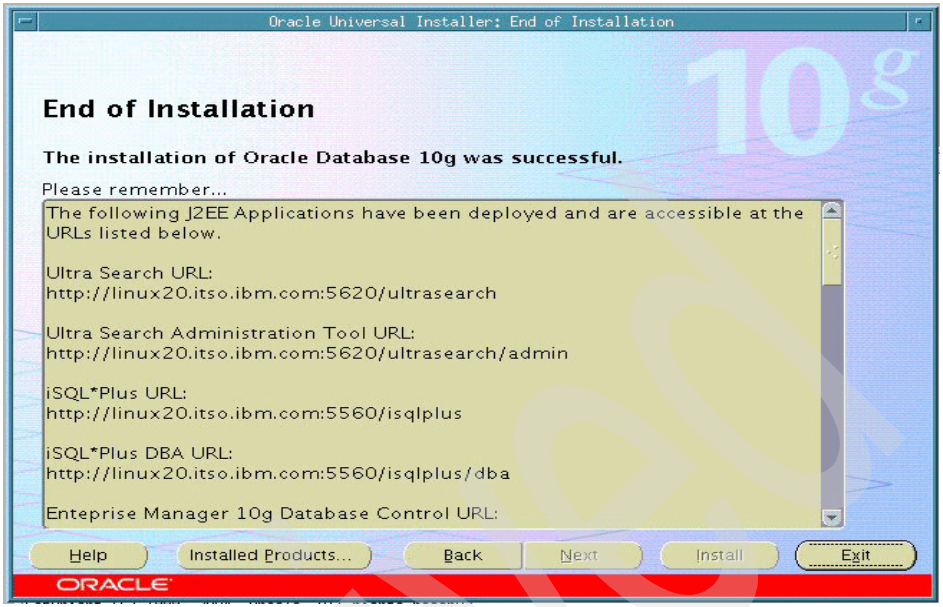


Figure 3-25 End of Installation panel

At this point, you have installed the Oracle binaries and created a database instance, so you can exit from the Oracle Universal Installer by clicking **Exit**. If you scroll down the page, you will see the URLs for the Oracle Enterprise Manager.

3.6 Verifying that the database is running

Once the process completes, exit the OUI. Now verify that the database is up. From the oracle user ID, enter:

```
sqlplus /'as sysdba'
```

This will give you command line access to the database with DBA privileges. Enter the following SQL command:

```
select name from v$database;
```

It will return the SID you gave it during the installation, orcl.

By logging into to Oracle with sqlplus as stated above, you can start and stop the database with the command startup and shutdown. As long as the ORACLE_SID

variable is set, no parameters are needed. When using the **shutdown** command, please use this version:

```
shutdown immediate
```

To exit it out of sqlplus:

```
sqlplus> exit
```

The Oracle Listener should also be running at the completion of the install. The listener listens on a port for users who want to come into the database from OracleNet (that is, SQL*Net, Net8, tns, etc.), the Oracle protocol that sits on top of TCP/IP as the communication link to the database. Enter the command **lsnrctl status** and you should get a message indicating the listener is up for the database. The listener can be started and stopped by entering the following command:

```
lsnrctl {start, stop}
```

To restart the database, you can issue the following commands:

```
sqlplus /'as sysdba'
startup
```

At this point, you have a running database.

3.7 Enabling Async IO

Oracle Database 10g does not install asyncio by default. To enable asyncio, you will need to relink oracle. After the installation process has successfully completed, shut down the database and stop the listener, then execute the following procedure to enable asyncio.

Example 3-1 Make command to enable asyncio in Oracle

```
Linux20:~> cd $ORACLE_HOME/rdbms/lib
Linux20:~/oradbf/O10g/rdbms/lib> make -f ins_rdbms.mk async_on
Linux20:~/oradbf/O10g/rdbms/lib> make -f ins_rdbms.mk  ioracle
```

Attention: Use **make -f ins_rdbms.mk async_off** to switch back in case of problems.

After we completed the above we made changes to the init.ora. We included the following two parameters in our init.ora to enable asyncio and direct I/O.

```
DISK_ASYNCH_IO=TRUE
FILESYSTEMIO_OPTIONS=SETALL
```

There is a combination of parameters to use, depending on what you want to do.

- DISK_ASYNC_IO=TRUE is all that is needed for asyncio for raw devices.
- FILESYSTEMIO_OPTIONS=ASYNC is required for asyncio only for file systems.

3.8 Using the LOCK_SGA parameter

If you decide you would like to use the init.ora parameter to lock the sga in memory you can follow these instructions:

- On SLES9
  - Issue the command **sysctl -w vm.disable_cap_mlock=1**.
  - Add lock_sga = true to the init.ora.
  - Start the Oracle database instance.

  You receive this error message if it is not set up correctly:

  ```
  ORA-27126: unable to lock shared memory segment in core
  Linux-s390x Error: 1: Operation not permitted
  ```

- On RedHat 4
  - Add these two lines to the /etc/security/limit.conf file:

    ```
    oracle          soft    memlock 4096000
    oracle          hard    memlock 4096000
    ```

  - Add the init.ora parameter - lock_sga = true.

    With RH without making any changes you get this message:

    ```
    SQL> startup
    ORA-27102: out of memory
    Linux-s390x Error: 12: Cannot allocate memory
    ```

- On SLES8

  We do not recommend this, but you can change the owner of oracle binary to root and add user ID root to group dba, then start the database with lock_sga = true in init.ora.

In each case, the startup of the database confirmed success:

```
SQL> startup
ORACLE instance started.

Total System Global Area 1241513984 bytes
Fixed Size                  1321512 bytes
Variable Size             333698520 bytes
Database Buffers          905969664 bytes
Redo Buffers                 524288 bytes
```

Archived

```
Database mounted.
Database opened.
SQL> show parameter lock_sga;

NAME                                 TYPE        VALUE
------------------------------------ -----------
lock_sga                             boolean     TRUE
SQL>
```

Note that the first two options do not work on SLES8.

3.9 Using OEM to manage an Oracle database

You can use OEM through DB Control to manage an Oracle database. After the successful completion of the installation of the database you can invoke Oracle Enterprise Manager using the URLs that are displayed by the OUI, as shown in Figure 3-25 on page 49. The OEM Login screen is the result of entering the URL (host:port/em) in an IE browser.

This shows we logged on with user ID SYS and selected SYSDBA from the pull-down option.

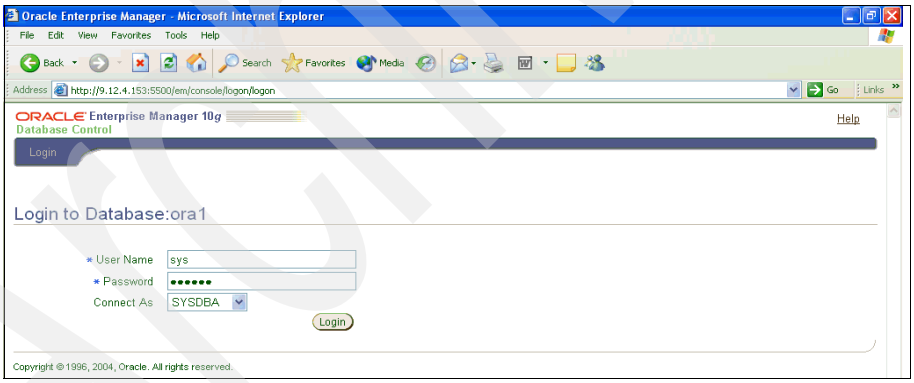


Figure 3-26 Logon screen to OEM

This is the OEM Home screen after the logon is complete. It is a scrollable screen. This only shows the top part of the functions available.

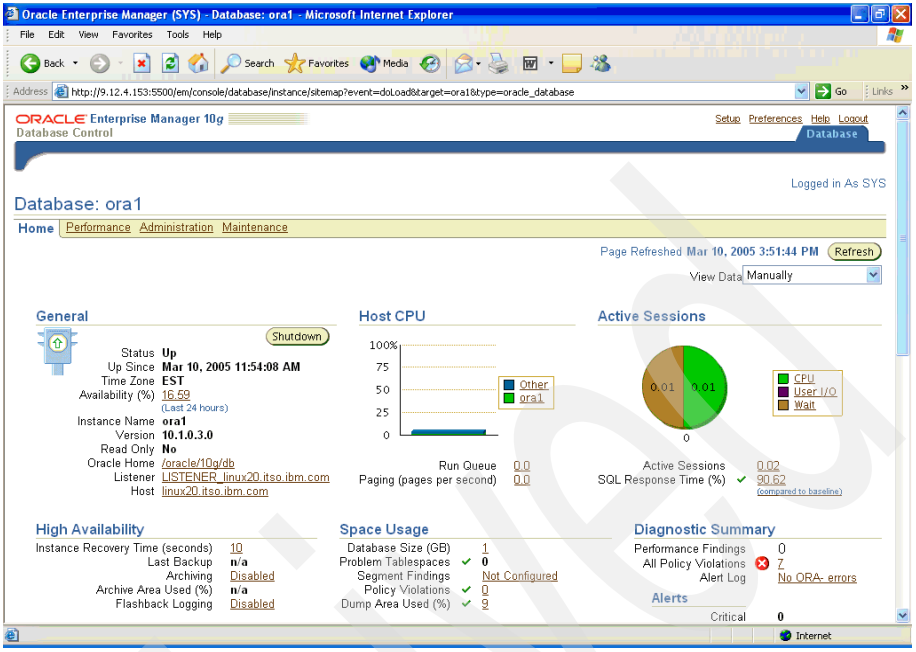


Figure 3-27 Home screen for Oracle Enterprise Manager

You can use this to manage your single instance database such as checking for patches, running Statspack, and so on.

Archived

Archived

4

Installing an Oracle 10g Database with ASM

This chapter describes the new Automated Storage Manager (ASM) that is available with Oracle Database 10g.

This is an example of the creation of a single database instance choosing the ASM option for the disk storage for the database files.

© Copyright IBM Corp. 2005. All rights reserved.

4.1 ASM overview

One of the new features coming with Oracle 10g is the Automatic Storage Management (ASM). It integrates into the database product the capabilities of both a Volume Manager and a file system, and relieves the DBA of most of the painful tasks of storage administration. Through simplified interfaces, ASM makes storage management and provisioning easy, flexible, and dynamic.

Key features of ASM are summarized in the Oracle white paper *Oracle 10g ASM technical best practices* by Nitin Vengurlekar. The full document can be found on metalink at:

http://metalink.oracle.com/metalink/plsql/showdoc?db=NOT&id=265633.1

As described in this document, the key ASM features are:

- I/O is spread evenly across all available disk drives to prevent hot spots and maximize performance.
- Inherent large file support.
- Performs automatic online redistribution after the incremental addition or removal of storage capacity.
- Maintains redundant copies of data to provide fault tolerance, or it can be built on top of vendor-supplied reliable storage mechanisms.
- Provides database storage management for single SMP machines, or across multiple nodes of a cluster for Oracle Real Application Clusters (RAC) support.
- Leverage redundancy from intelligent storage arrays.
- For simplicity and easier migration to ASM, 10g will allow the co-existence of ASM and non-ASM files. Any new files can be created as ASM files while existing files can also be migrated to ASM.
- New RMAN commands enable non-ASM managed files to be relocated to an ASM diskgroup.
- 10g Enterprise Manager can manage most ASM disk and file management activities.

4.2 Setting up ASM

On Linux for Intel, Oracle offers two alternatives to use ASM. The first is to use ASM as provided with Oracle 10g code. It implies the use of raw disks (character devices) to store the data. The second does not require raw devices, since the disks being accessed as block devices. It allows for more advanced performance

options, but it requires some external components. The ASM.LIB driver and utilities and OCFS2, which are downloadable from the OTN Web site, should be available late in 2005 for zSeries. So far, these components have not been made available on Linux on zSeries.

4.3 Binding disks to raw devices

Using raw disks is therefore the only option available for ASM on Linux on zSeries. The first step is to configure some raw devices to be bound with real disks at boot time. SLES8 provides a configuration file for that purpose: /etc/raw (RedHat uses /etc/sysconfig/rawdevices).

Example 4-1 Raw devices definitions in /etc/raw

```
# /etc/raw
#
# sample configuration to bind raw devices
# to block devices
#
# The format of this file is:
# raw<N>:<blockdev>
#
# example:
# ---------
# raw1:hdb1
#
# this means: bind /dev/raw/raw1 to /dev/hdb1
#
# ...
raw1:dasdc1
raw2:dasdd1
```

The above definition will bind disks dasdc1 and dasdd1, respectively, with raw devices /dev/raw/raw1 and /dev/raw/raw2.

The raw devices binding can be activated dynamically using the *raw* script in /etc/init.d, and the devices status can be checked with the **raw -qa** command.

Example 4-2 Raw devices activation and status

```
oracle3:/etc # /etc/init.d/raw start
bind /dev/raw/raw1 to /dev/dasdc1...                                done
bind /dev/raw/raw2 to /dev/dasdd1...                                done
oracle3:/etc # raw -qa
/dev/raw/raw1:  bound to major 94, minor 9
```


Archived

```
/dev/raw/raw2:  bound to major 94, minor 13
```

Tip: To activate raw devices binding at boot time, use the following command:

```
/sbin/chkconfig raw on
```

The last step is to make sure that the Oracle user is authorized to access the raw devices. By default, raw devices are owned by user *root* and group *disk*. It is therefore necessary to change the ownership using the **chown** command:

```
chown oracle:disk /dev/raw/raw*
```

4.4 Configuring ASM instance using DBCA

Prior to creating a database using ASM, we need to create an ASM instance. It can be seen as a unique *storage service provider* for all the database instances running on the same Linux system. An ASM instance does not contain any datafiles, and requires only a few parameters in the init.ora file. In the case of several systems operating within a RAC cluster, then one ASM instance is required on each of the systems.

The DBCA utility can be used to create an ASM instance. Launch the DBCA utility and start the dialog to create a new database until step 6 - Storage Options.

Archived

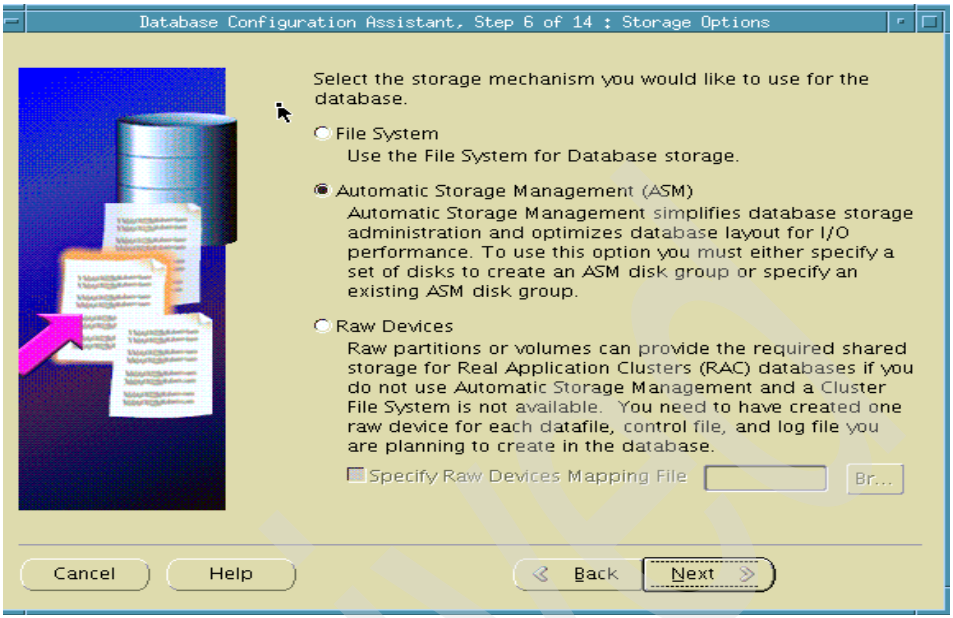


Figure 4-1 Step 6 - Storage options

In the Storage Options window select **ASM** and click **Next**. If this is the first time ASM is used, the Create ASM instance panel is presented.

Archived

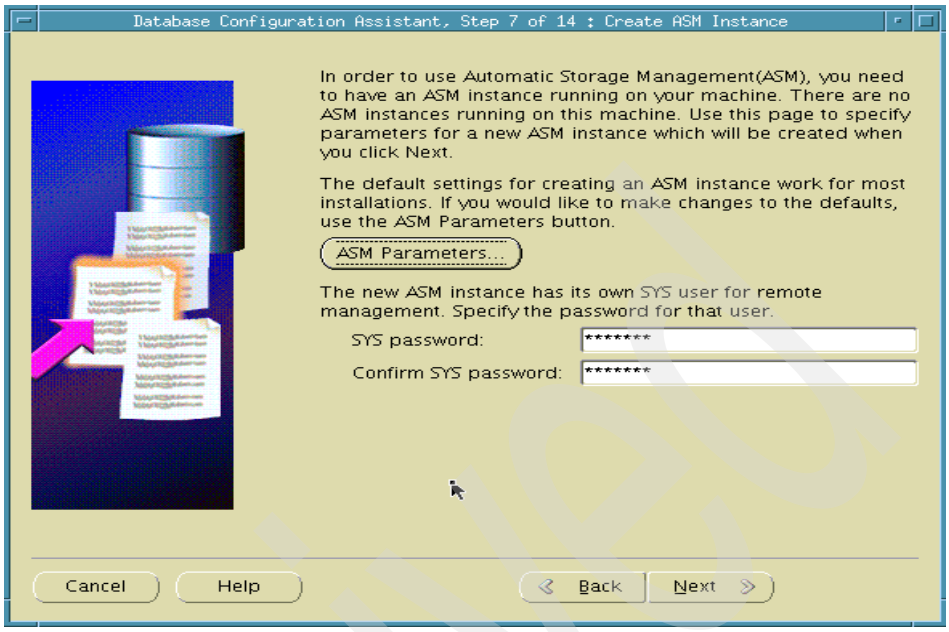


Figure 4-2 Step 7 - Create ASM instance

Filling the password field and then clicking the **Next** button will initiate the creation of the ASM instance, and a pop-up asking for confirmation will appear.

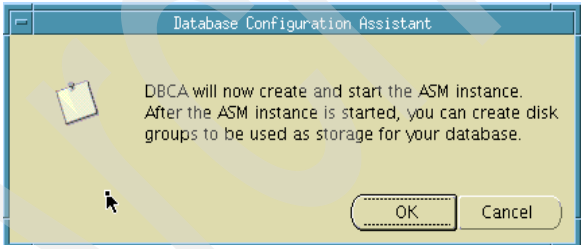


Figure 4-3 ASM instance creation confirmation

Attention: Even when working in a single instance (non-clustered), ASM requires that the Cluster Synchronization Services (CSS) are installed and working properly. CSS is used to preserve synchronization between ASM and databases instances.

The ASM parameters should be left to defaults in most cases. However, if some changes are needed then the ASM parameters button in the previous window will display a panel with all parameters listed as below.

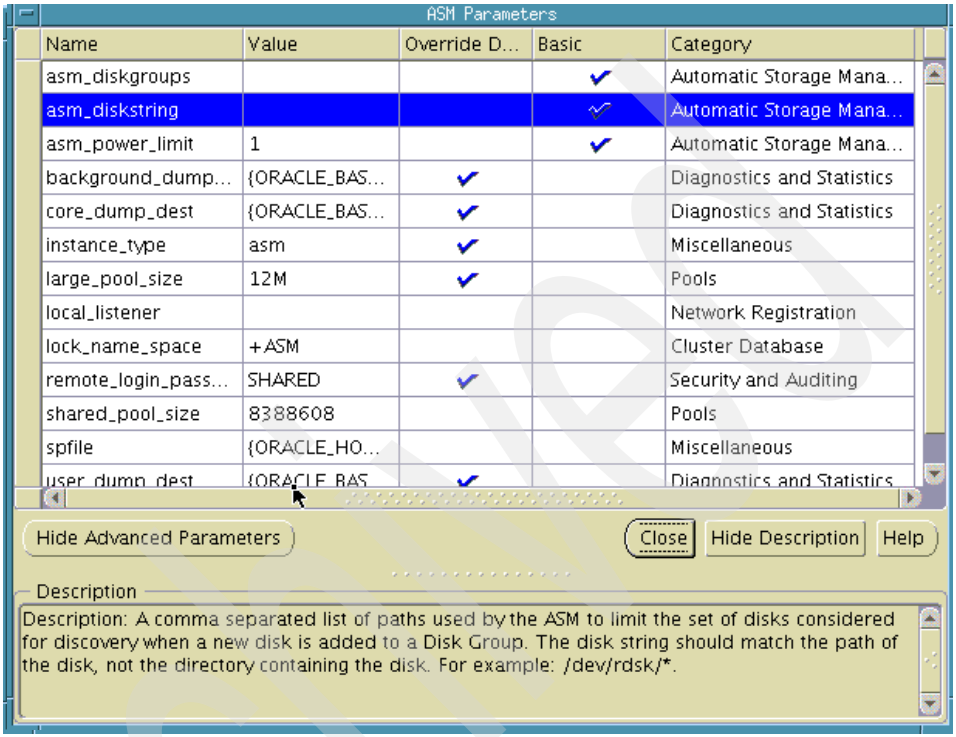


Figure 4-4 ASM parameters

Once the ASM instance has been started successfully, it is time to define a diskgroup and to include some disks into it. A diskgroup is the highest level storage structure in ASM; it can be compared to a Volume Group in LVM. Oracle recommends using only two diskgroups per ASM instance: One for the database files and one for the Flash Recovery Area, containing all backup and recovery-related files. This is done in the next window.

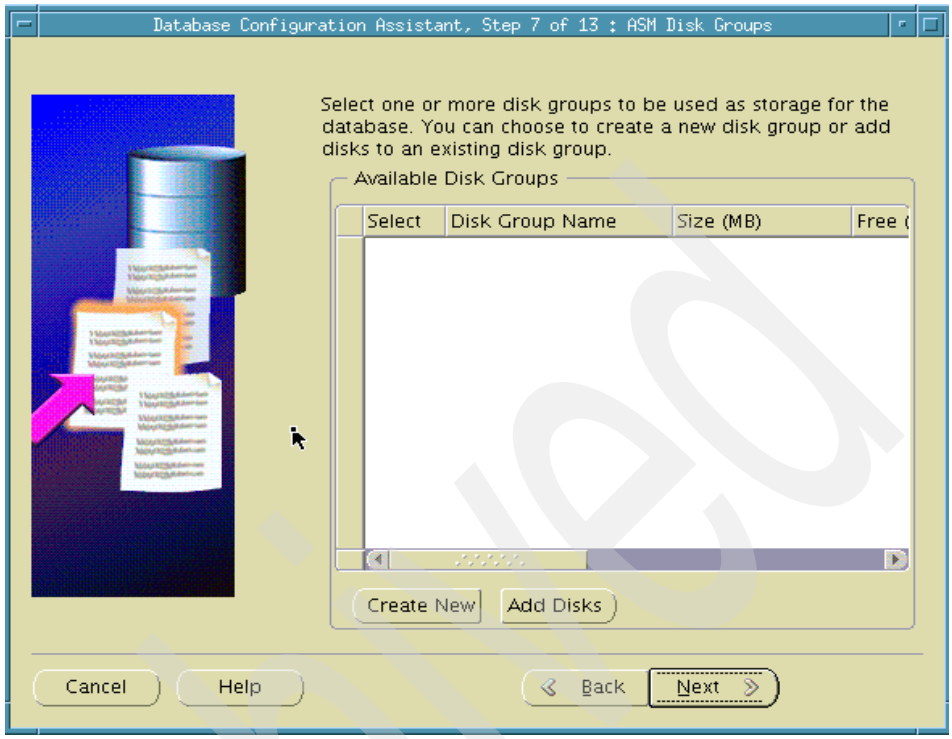


Figure 4-5 ASM diskgroups

Click the **Create New** button in the ASM Disk Group window.

ASM will scan the defined discovery path to check which disks are configured for use. On Linux, this path is set by default to /dev/raw/* in order to discover all the raw devices. The next window should show the raw devices that have been previously defined, which ones are available (candidate) for use, and which ones are already in use.

Archived

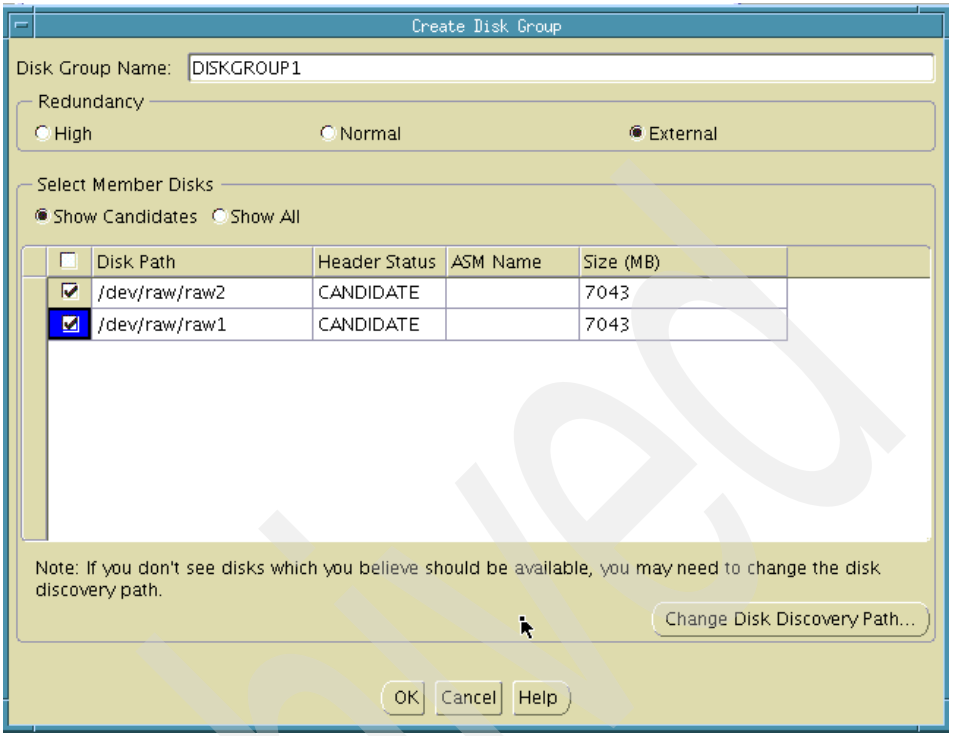


Figure 4-6 Create Disk Group

After a name is entered for the diskgroup, the redundancy level must be chosen, and the disks to be included in that diskgroup must be selected.

ASM provides three levels of redundancy. Redundancy can be seen as mirroring, the base unit for mirroring being the block of data, not the physical disk. This ensures better performance, and a more rationale use of space—that is, there is no one single pool of disks used as base, and another pool as mirror image; each disk can receive any kind of block. (ASM must make sure a block and its mirror images never end up onto the same disks.)

- *Normal* is the default and provides in two ways software mirroring (each block of data is mirrored once).
- *High* provides for 3-way mirroring.
- *External* ensures no mirroring at the software level, and assumes it is provided externally. It may be accurate on a zSeries platform, as lots of effort has been made on storage subsystems like ESS to provide hardware reliability: No single points of failure and the ability to configure disks in RAID-5 or RAID-10. See ESS redbooks.

Tip: If some of the configured raw devices do not show up on the previous window, the cause may be:

- The binding has not been done correctly. Use the **`raw -qa`** command to check.
- The discovery path is wrong. It should be set to /dev/raw/* on SLES8. It may have to be changed depending on Linux distributions and releases.
- The ownership of the raw device has not been changed from root to oracle.

The next window is presented for diskgroup selection.

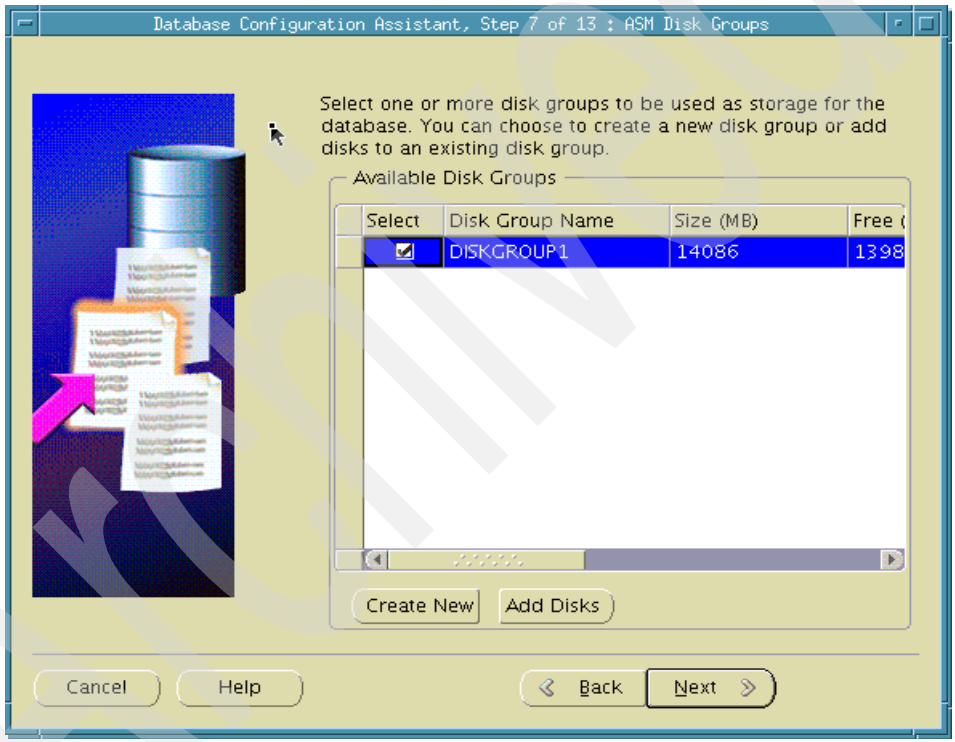


Figure 4-7 Disk group selection

Eventually this step could also be used to define new diskgroups, or to include new disks in a diskgroup. Select the diskgroup on which the database is to be created, and click **Next** to go to the database file location options.

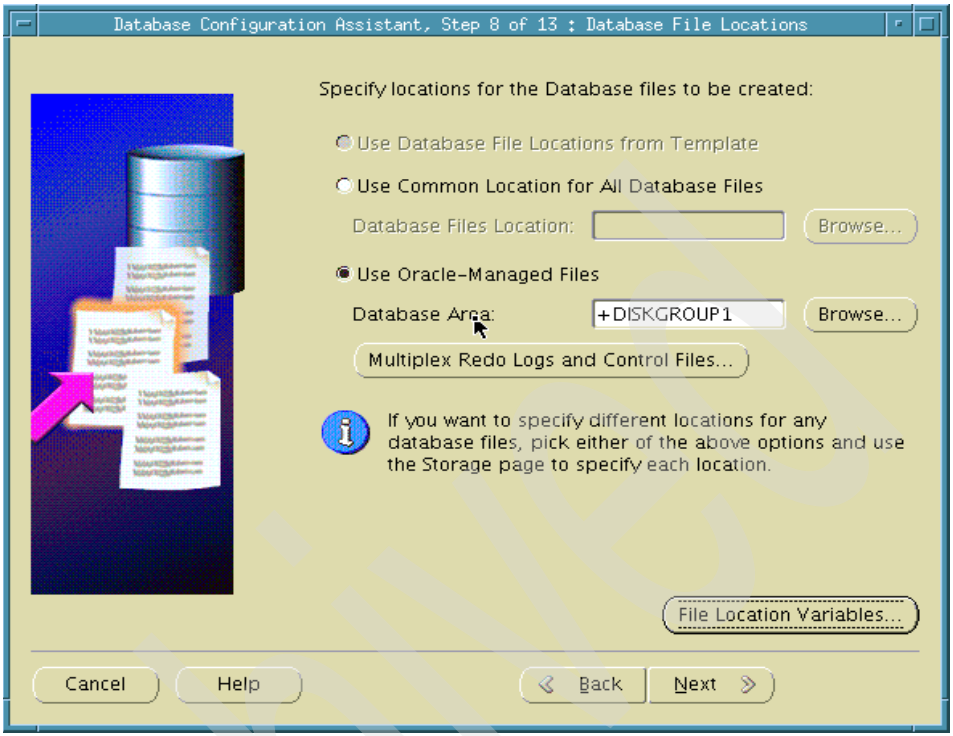


Figure 4-8 Database files location

This step allows for more details on where to put the DB files (if, for example, you prefer to put the redo logs on separate devices) as well as some multiplexing options (click **HELP** for more details).

Next we present the recovery options.

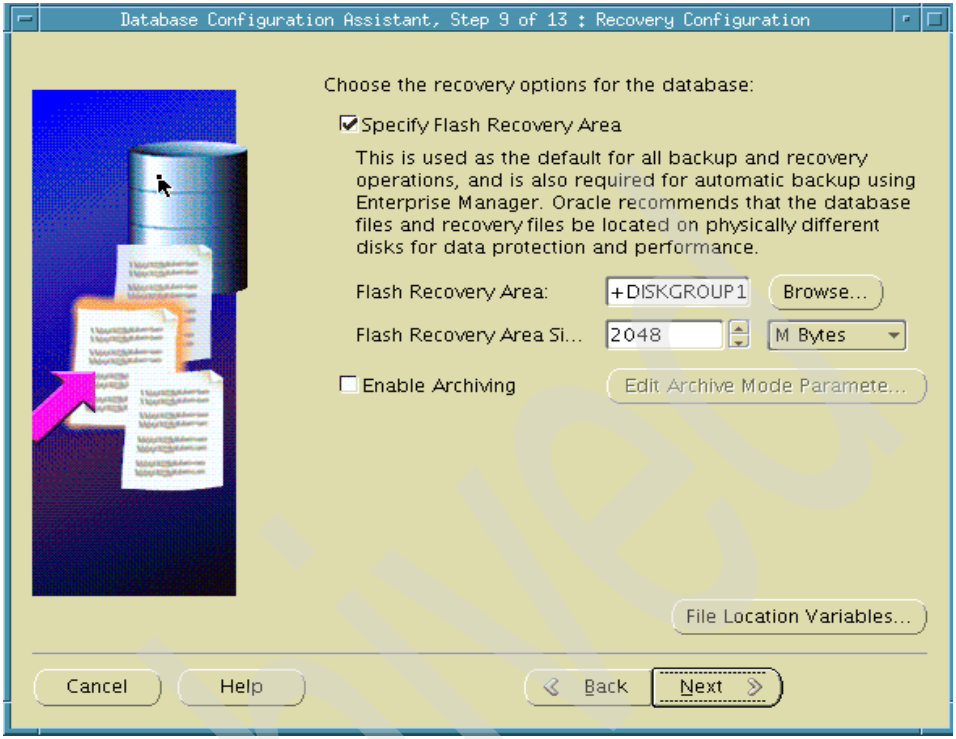


Figure 4-9 Recovery configuration

Oracle recommends putting the Flash Recovery Area on a different diskgroup other than the DB files, to ensure that active data and backup/recovery data will never be on the same physical disk.

This is the last ASM-related panel. The next steps are normal database creation steps using DBCA (database content, initialization parameters, storage options) and have been covered in 4.4, “Configuring ASM instance using DBCA” on page 58.

4.5 Managing ASM using SQL commands

Instead of using DBCA, the DBA now has a convenient option to manage ASM and disk space using SQL orders. We give some examples of such commands (in bold case); refer to *Oracle Administrator’s Guide* for more details.

4.5.1 Connect to the ASM instance

> **Tip:** The default name for the ASM instance is +ASM, as shown in the /etc/oratab entry generated by the DBCA dialog:
>
> `+ASM:/opt/oracle/product/10.1.0/db_1:N`
>
> Generally, all parameter names related to ASM that are generated by the system begin with a plus sign (+).

Example 4-3 Connecting to ASM

```
oracle@oracle3:~> export ORACLE_SID=+ASM
oracle@oracle3:~> scalpels "/ as sysdba"

SQL*Plus: Release 10.1.0.3.0 - Production on Tue Nov 30 11:15:25 2004

Copyright (c) 1982, 2004, Oracle.  All rights reserved.


Connected to:
Oracle Database 10g Enterprise Edition Release 10.1.0.3.0 - 64bit Production
With the Partitioning, OLAP and Data Mining options

SQL>  select instance_name from v$instance;

INSTANCE_NAME
----------------
+ASM
```

4.5.2 Creating a new diskgroup

The first step before creating a new diskgroup is to query ASM about the current disks allocations, using the v$asm_disk view.

Example 4-4 Displaying disks

```
SQL> select name,  path, disk_number, group_number from v$asm_disk;

NAME                PATH                DISK_NUMBER GROUP_NUMBER
-----------------   -----------------   ----------- ------------
                    /dev/raw/raw4             0            0
                    /dev/raw/raw3             1            0
DISKGROUP1_0000     /dev/raw/raw2             0            1
DISKGROUP1_0001     /dev/raw/raw1             1            1
```

Archived

The above command scans the ASM disk discovery path. It shows that raw devices raw1 and raw2 are currently in use by diskgroup DISKGROUP1, and devices raw3 and raw4 are defined and available for use (diskgroup NAME is null and GROUP_NUMBER is 0).

Now we can create a new diskgroup, and allocate raw devices, raw3 and raw4, to that diskgroup. We define this diskgroup with a redundancy type *external*.

Example 4-5 Creating diskgroup

```
SQL> create diskgroup DISKGROUP2 external redundancy disk
'/dev/raw/raw3','/dev/raw/raw4';
Diskgroup created.
```

We can query the result using the v$asm_diskgroup view.

Example 4-6 Displaying diskgroups

```
SQL>  select name,state,type,group_number,total_mb,free_mb from v$asm_diskgroup;

NAME                           STATE       TYPE   GROUP_NUMBER   TOTAL_MB    FREE_MB
------------------------------ ----------- ------ ------------ ---------- ----------
DISKGROUP1                     MOUNTED     NORMAL            1      14086      12246
DISKGROUP2                     MOUNTED     EXTERN            2      14086      14034
```

4.5.3 Modifying an existing diskgroup

One of the most interesting features of ASM is that it makes it possible for a DBA to dynamically increase the size of storage available for database use, without having to stop Oracle and use complex resizing commands. Furthermore, it also allows for not only preserving the stripping of data over the disks, but also for redistributing the data in such a way that all the disks (including the newly allocated ones) end up having the same amount of file extents and an identical usage pattern.

From the following SQL command, we can see that DISKGROUP1 is made up of two disks of 7 GB each, 1 GB on each disks being used by a database previously created.

Example 4-7 Displaying disks

```
SQL> select name,group_number,total_mb,free_mb,path from v$asm_disk;

NAME                           GROUP_NUMBER   TOTAL_MB    FREE_MB PATH
------------------------------ ------------ ---------- ----------
                                          0       7043          0 /dev/raw/raw6
                                          0       7043          0 /dev/raw/raw5
DISKGROUP1_0000                           1       7043       6043 /dev/raw/raw2
```

```
DISKGROUP1_0001                          1       7043       6043 /dev/raw/raw1
DISKGROUP2_0001                          2       7043       7018 /dev/raw/raw4
DISKGROUP2_0000                          2       7043       7016 /dev/raw/raw3

6 rows selected.
```

We are going to insert a new disk in that diskgroup and ask ASM to distribute the data evenly amongst the three disks.

Example 4-8 Adding a new disk

```
SQL> alter diskgroup diskgroup1 add disk '/dev/raw/raw5' rebalance power 1;

Diskgroup altered.

SQL> select group_number,operation,state,est_work,sofar,est_rate,est_minutes
from v$asm_operation;

GROUP_NUMBER OPERA STAT   EST_WORK      SOFAR   EST_RATE EST_MINUTES
------------ ----- ---- ---------- ---------- ----------
           1 REBAL RUN         715         64        428           1
```

The above alter command, used to add a new disk, also allows us to specify a rebalance power. This power will influence how fast and impacting for the system the data redistribution will be. The default is specified in init.ora with the asm_power_limit parameter. It can vary from 1 (low speed, low impact on performances) to 11 (full throttle, high impact). Optionally, the **alter diskgroup** command can override the default rebalance power, and also specify a 0 value to prevent rebalancing. The v$asm_operation view can then be used to check the processing of the rebalancing operation. A query command will show the new data distribution on disk after rebalancing.

Example 4-9 Disks view

```
SQL> select name,group_number,total_mb,free_mb,path from v$asm_disk;

NAME                           GROUP_NUMBER   TOTAL_MB    FREE_MB PATH
------------------------------ ------------ ---------- -----------
                                          0       7043          0 /dev/raw/raw6
DISKGROUP1_0000                           1       7043       6342 /dev/raw/raw2
DISKGROUP1_0001                           1       7043       6324 /dev/raw/raw1
DISKGROUP2_0001                           2       7043       7018 /dev/raw/raw4
DISKGROUP2_0000                           2       7043       7016 /dev/raw/raw3
DISKGROUP1_0002                           1       7043       6390 /dev/raw/raw5
6 rows selected.
```

The output of the previous command shows that some data has been moved from raw2 and raw1 to raw5, so that the three disks show an almost identical amount of used space.

4.6 Managing ASM using OEM

It is also possible to manage ASM through the Enterprise Manager Web interface.

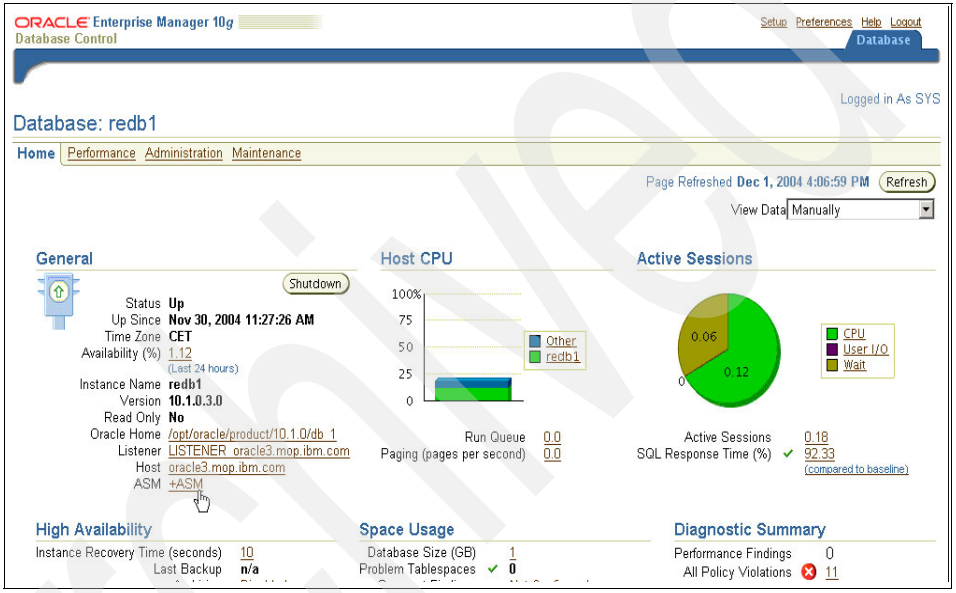


Figure 4-10 OEM primary panel

From the OEM primary panel above, select the ASM option.

Archived

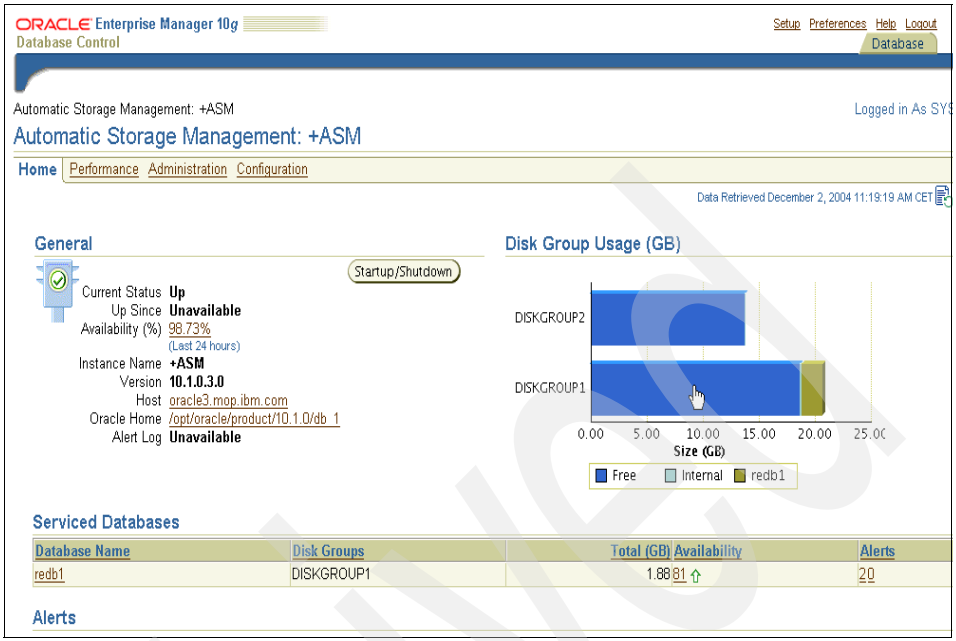


Figure 4-11 ASM main window

From the ASM window, select a diskgroup for a detailed view.

Archived

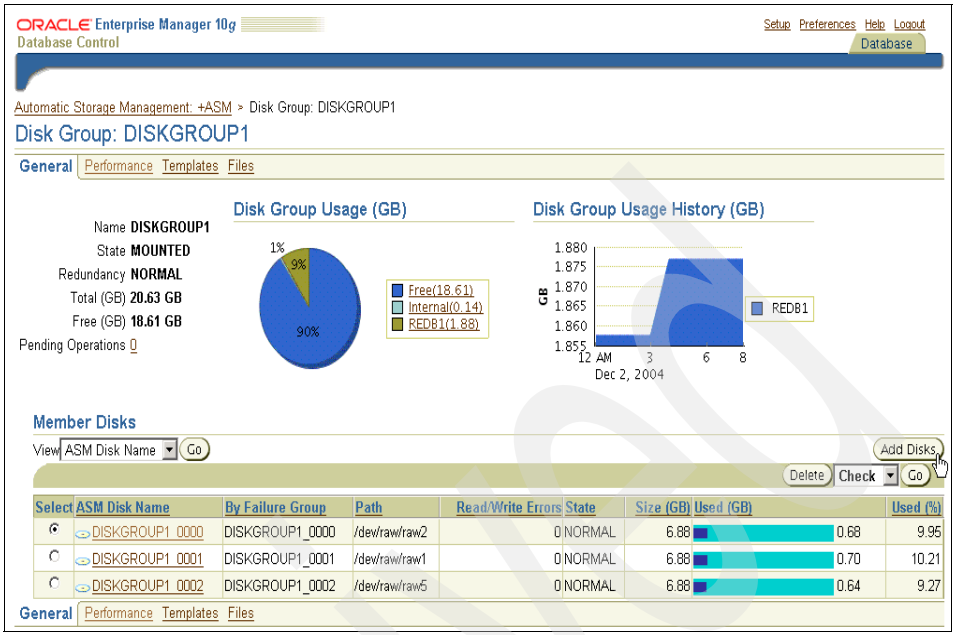


Figure 4-12 Diskgroup view

From here, the main possibilities are:

- To click the ADD DISK button to add new disks to the diskgroup.

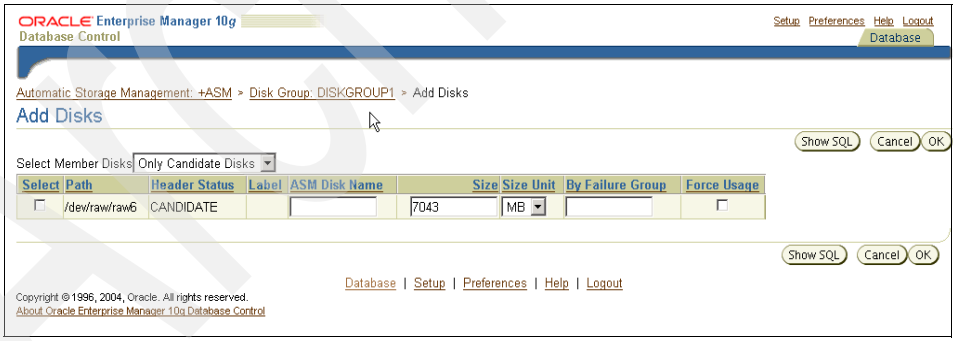


Figure 4-13 Add Disk window

- To select the FILES option to check how database files are spread over the diskgroup.

Archived

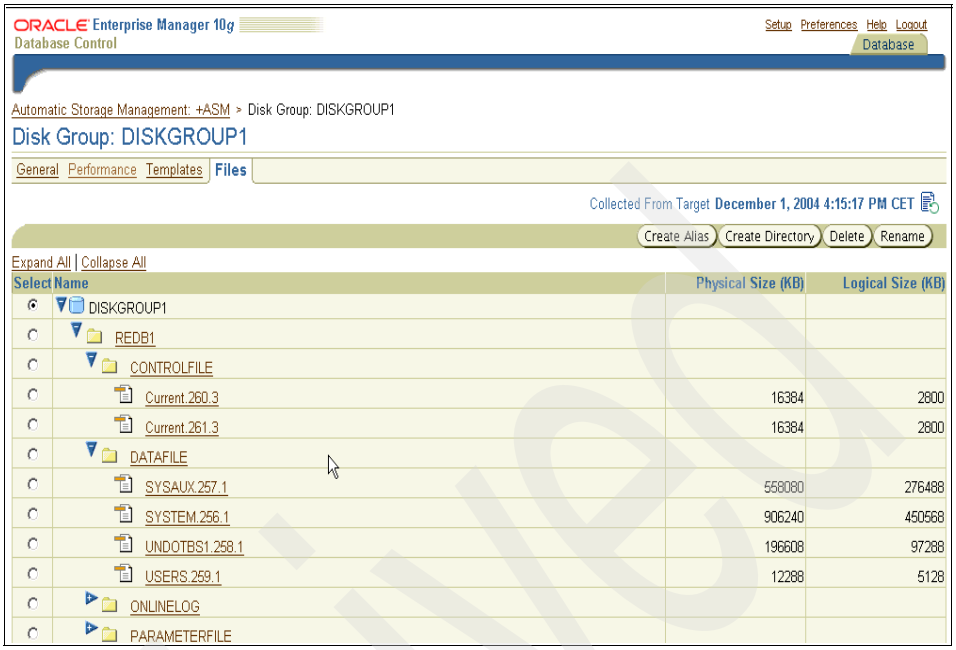


Figure 4-14 Files window

- Use the PERFORMANCE option to display various I/O statistics.

Archived

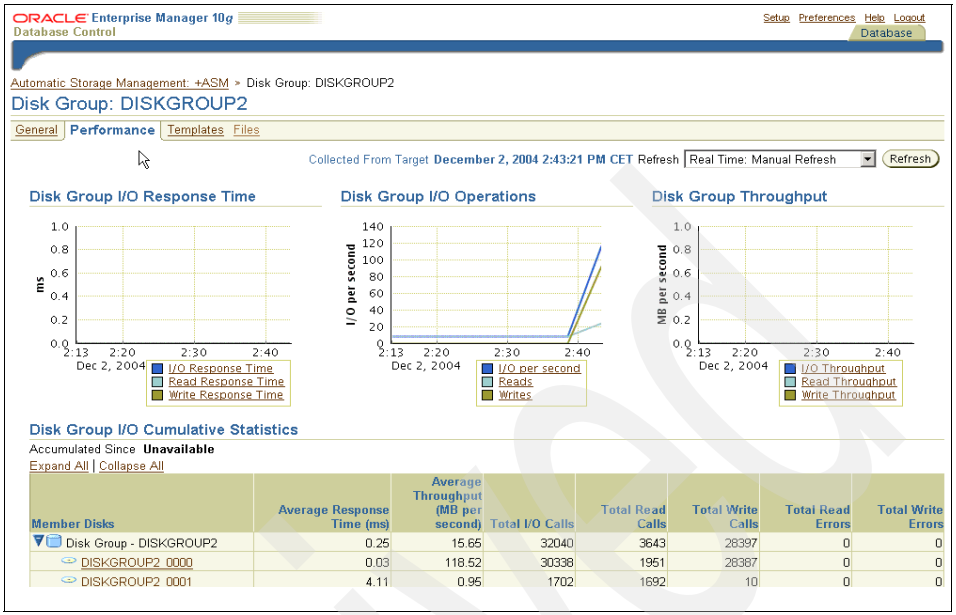


Figure 4-15 Performance window

4.7 ASM best practices

ASM best practices follow.

- All disks in a diskgroup should have the same characteristics (performance and size) to ensure consistent performance. If several types of disks are used, they should be placed in different diskgroups.
- ASM will ensure the data is striped evenly amongst all disks. The storage administrator still has to ensure that these disks are spread evenly within the storage subsystem. In the case of an ESS, make sure the disks are well balanced over several control units.
- ASM disks should not be managed by another volume manager, like LVM, especially when some striping mechanism could interfere with ASM.
- Oracle suggests that, for ease of management, no more than two diskgroups should be defined per ASM instance—one for the database files, one for the Flash Recovery Area.
- The following files cannot reside on ASM-managed storage: Oracle binaries (executable), CRS files (OCR, voting disks), trace files, and non-database files in general.

- Although ASM does not provide imbedded multipathing recognition, it has the capability to handle it when it is provided by third-party software. On Linux on zSeries, such a mechanism has been implemented through LVM. LVM allows the operating system, and thus ASM, to increase I/O parallelism and failover capabilities. For more details, see the multipathing chapters in the Fiber Channel Protocol redbook at http://www.redbooks.ibm.com/redbooks/pdfs/sg246344.pdf/ and also the Parallel Access Volumes paper at http://www10.software.ibm.com/developerworks/opensource/linux390/docu/lx24pav00.pdf/. The way to implement this with ASM is to define one logical volume per disk, mapping the whole disk, and to integrate this logical disk and its aliases (pseudo devices) into an ASM diskgroup.

Archived

Archived

5

Installing CRS and RAC

We installed Oracle Cluster Ready Services (CRS) and Oracle 10g Database Version for Linux with Real Application Clusters (RAC) for zSeries under VM. Then we created a database using Database Creation Assistant (DBCA).

Real Application Clusters is an Oracle feature for sharing a database instance between two or more nodes. The reasons for using RAC are:

- Increased availability
- Increased scalability

RAC is easy to implement on zSeries because of the existing mainframe infrastructure features such as:

- Shared DASD is a feature of zSeries and z/VM.
- High Speed Interconnect for z/Linux using channel-to-channel connection, or ICUV or Hipersockets or Ethernet (Gigabit is preferred).

This chapter describes our:

- VM and Linux guest setup
- CRS installation
- Oracle 10g RAC installation
- Use of DBCA to create the shared database

© Copyright IBM Corp. 2005. All rights reserved.

5.1 VM set up

By sharing the database files, Oracle 10g RAC can be deployed in several ways on zSeries:

- Two or more instances in one LPAR
- Two or more LPARs in one system, each with an Oracle instance
- Two or more systems in the same sysplex, each with an Oracle instance

In our case, we used VM 4.4 and two Linux guests in one LPAR, as shown on the right-hand side of Figure 5-1.

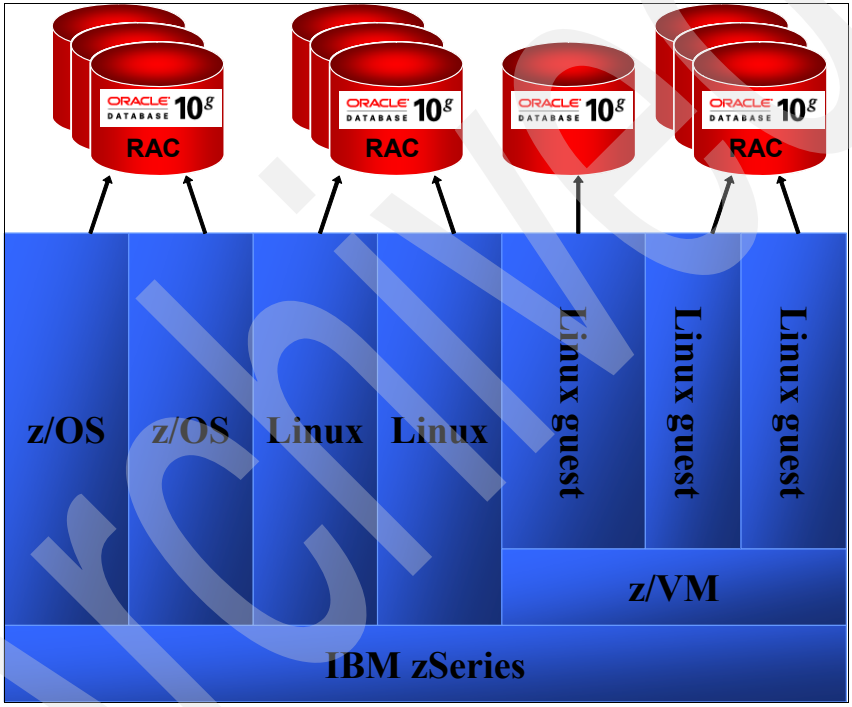


Figure 5-1 Possible RAC setup on one zSeries system

Software

The environment we used for the installation of Oracle10g was Linux SLES8 kernel 2.4.21.-261 running under VM 4.4. Recently we tested the installation with SLES9 kernel 2.6.5.-151.

Archived

Documentation

Oracle documentation can be obtained at:

http://www.oracle.com/technology/documentation/database10g.html

We used the following Oracle documentation for this installation.

Oracle Real Application Clusters Installation and Configuration Guide 10g Release 1 (10.1) for AIX®-Based Systems, hp HP-UX PA-RISC (64-bit), hp Tru64 UNIX, Linux, Solaris™ Operating System (SPARC 64-bit), and Windows Platforms Part No. B10766-08.

There are several metalink notes on CRS that we found to be very helpful:

- 259301.1 - CRS and 10g Real Application Clusters
- 265769.1 - 10g RAC: Troubleshooting CRS Reboots
- 239998.1 - 10g RAC: How to Clean Up After a Failed CRS Install

Virtual machine setup

The Linux virtual machines are defined in the user directory of VM. Example 5-1 shows the entries of our Linux guests pazxxto5 and pazxxt06.

Example 5-1 Directory entries for Linux guests

```
USER PAZXXT06 xxxxxxx 2G 6G G 64
   ACCOUNT CO000620 LINUX
   CPU 00 BASE
   IPL CMS PARM AUTOCR
   IUCV *IDENT GATEANY GATEWAY REVOKE
   IUCV ALLOW
   OPTION TODENABLE LNKNOPAS
   POSIXINFO UID 997
   CONSOLE 001F 3215 A
   SPOOL 000C 2540 READER *
   SPOOL 000D 2540 PUNCH A
   SPOOL 000E 1403
*
   LINK MAINT 019E 019E RR
   LINK MAINT 019D 019D RR
   LINK MAINT 0190 0190 RR
   LINK VMLINUX 0104 0191 RR
*
   MDISK 0200 3390 0001 00020 LXC05A MR BOOT
   MDISK 0201 3390 0001 03338 LXC0BE MR ROOT
   MDISK 0204 3390 5839 00800 LXC15A MR TMP
   MDISK 0205 3390 3339 00200 LXC108 MR HOME
   MDISK 0206 3390 1311 01500 LXC385 MR USR LIB64
*  swap space definitions
   MDISK 0202 FB-512 V-DISK 204800 MR
```

```
   MDISK 0203 FB-512 V-DISK 204800 MR
*  local disks
   MDISK 0300 3390 0001 10016 LXC02E MR
   MDISK 0301 3390 0001 10016 LXC21D MR
*  raw devices
   MDISK 0400 3390 3359 3000 LXC04F MW
   MINIOPT NOMDC
   MDISK 0401 3390 0001 3000 LXC2E4 MW
   MINIOPT NOMDC
   MDISK 0402 3390 3001 3000 LXC0BB MW
   MINIOPT NOMDC
   MDISK 0403 3390 7201 2800 LXC32C MW
   MINIOPT NOMDC
   MDISK 0404 3390 3838 3000 LXC11F MW
   MINIOPT NOMDC
   MDISK 0405 3390 4001 3000 LXC294 MW
   MINIOPT NOMDC
   MDISK 0406 3390 0101 3000 LXC0D0 MW
   MINIOPT NOMDC
   MDISK 0407 3390 6001 3000 LXC293 MW
   MINIOPT NOMDC
   MDISK 0408 3390 4339 3000 LXC0BC MW
   MINIOPT NOMDC
   MDISK 0409 3390 3201 3000 LXC253 MW
   MINIOPT NOMDC
   MDISK 040A 3390 6339 3000 LXC0BE MW
   MINIOPT NOMDC
   MDISK 040B 3390 6701 3000 LXC2C8 MW
   MINIOPT NOMDC
   MDISK 040C 3390 6539 3000 LXC0D5 MW
 MINIOPT NOMDC
 MDISK 040D 3390 7001 3000 LXC294 MW
 MINIOPT NOMDC
 MDISK 040E 3390 4601 3000 LXC0D7 MW
 MINIOPT NOMDC


USER PAZXXT06 xxxxxxx 2G 6G G 64
   ACCOUNT C0000620 LINUX
   CPU 00 BASE
*  CPU 01
   IPL CMS PARM AUTOCR
   IUCV *IDENT GATEANY GATEWAY REVOKE
   IUCV ALLOW
   OPTION TODENABLE LNKNOPAS
   POSIXINFO UID 998
   CONSOLE 001F 3215 A
   SPOOL 000C 2540 READER *
   SPOOL 000D 2540 PUNCH A
```

```
   SPOOL 000E 1403
*
   LINK MAINT 019E 019E RR
   LINK MAINT 019D 019D RR
   LINK MAINT 0190 0190 RR
   LINK VMLINUX 0104 0191 RR
*
   MDISK 0200 3390 0601 00020 LXC2B8 MR BOOT
   MDISK 0201 3390 3339 03338 LXC206 MR ROOT
   MDISK 0204 3390 8977 00800 LXC32B MR TMP
   MDISK 0205 3390 0811 00200 LXC38B MR HOME
   MDISK 0206 3390 4339 01500 LXC309 MR USR LIB64
*  swap space definitions
   MDISK 0202 FB-512 V-DISK 204800 MR
   MDISK 0203 FB-512 V-DISK 204800 MR
 *  local disks
    MDISK 0300 3390 0001 10016 LXC1AE MR
    MDISK 0301 3390 0001 10016 LXC24A MR
 *  raw devices
    MDISK 0400 3390 3359 3000 LXC04F MW
    MINIOPT NOMDC
    MDISK 0401 3390 0001 3000 LXC2E4 MW
    MINIOPT NOMDC
    MDISK 0402 3390 3001 3000 LXC0BB MW
    MINIOPT NOMDC
    MDISK 0403 3390 7201 2800 LXC32C MW
    MINIOPT NOMDC
    MDISK 0404 3390 3838 3000 LXC11F MW
    MINIOPT NOMDC
    MDISK 0405 3390 4001 3000 LXC294 MW
    MINIOPT NOMDC
    MDISK 0406 3390 0101 3000 LXC0D0 MW
    MINIOPT NOMDC
    MDISK 0407 3390 6001 3000 LXC293 MW
    MINIOPT NOMDC
    MDISK 0408 3390 4339 3000 LXC0BC MW
    MINIOPT NOMDC
    MDISK 0409 3390 3201 3000 LXC253 MW
    MINIOPT NOMDC
    MDISK 040A 3390 6339 3000 LXC0BE MW
    MINIOPT NOMDC
    MDISK 040B 3390 6701 3000 LXC2C8 MW
    MINIOPT NOMDC
    MDISK 040C 3390 6539 3000 LXC0D5 MW
 MINIOPT NOMDC
 MDISK 040D 3390 7001 3000 LXC294 MW
 MINIOPT NOMDC
 MDISK 040E 3390 4601 3000 LXC0D7 MW
```

Archived

```
MINIOPT NOMDC
```

Shared disk setup

Oracle RAC requires the sharing of disk between the nodes that will access the database. With IBM mainframes, the sharing of disk is part of the architecture. In our installation we used VM mini disks shared between two virtual machines.

These devices were previously formatted. When devices are not formatted you can used the command **dasdfmt** to do so.

The four lines in Figure 5-2 are the part of user directory definition for virtual machines pazxxt05 and pazxxt06 that correspond to the share devices we used.

These four lines are in each virtual machines definition.

Example 5-2 Example of definition of two shared disks

```
MDISK 0400 3390 3359 3000 LXC04F MW
MDISK 0401 3390 0001 3000 LXC2E4 MW
MINIOPT NOMDC
```

On the devices defined in Example 5-2 we created logical volumes and raw devices. See “Creating the raw devices” on page 84 and “Set up logical volumes” on page 84 for explanations on how we created the logical volumes and raw devices.

Attention: If you are using SLES9 SP2 or later or Redhat 4.0 you should edit the udev permissions file to add user oracle for /dev/raw nodes. It controls the permissions and ownership of device nodes. This is necessary so the /dev/raw nodes will still be owned by user oracle after reboot. The change we made for SLES9 in /etc/udev/udev.permissions was to add:

```
raw/*:oracle:dba:0775
```

In Redhat 4.0 the line should be added to the file /etc/udev/permissions.d/50-udev.permissions.

5.2 Linux setup

Attention: Make sure to start with a clean copy of SLES9 when you install CRS. Using a clone of a system that had Oracle installed before can cause installation problems.

The steps we completed to prepare our two Linux systems were:

- Setting the kernel values
- Moving the scripts to both nodes
- Creating the raw devices
- Creating an oracle user ID
- Setting up the logical volumes
- Making symbolic links
- Binding the raw devices
- Setting up the /etc/host file
- Setting up the ssh interface

5.2.1 Setting the kernel values

Using any text editor, create or edit the /etc/sysctl.conf file and add or edit lines similar to the following:

```
kernel.shmall = 2097152
kernel.shmmax = 2147483648
kernel.shmmni = 4096
kernel.sem = 250 32000 100 128
fs.file-max = 65536
net.ipv4.ip_local_port_range = 1024 65000
net.core.rmem_default = 262144
net.core.rmem_max = 262144
net.core.wmem_default = 262144
net.core.wmem_max = 262144
```

By specifying the values in the /etc/sysctl.conf file, the values persist when you restart the system.

Enter the following command to change the current values of the kernel parameters:

```
# /sbin/sysctl -p
```

Review the output from this command to verify that the values are correct. If the values are incorrect, edit the /etc/sysctl.conf file, then enter this command again.

5.2.2 Moving the scripts to both nodes

You will need to complete the following setup steps on both nodes. You can use scp or ftp to move the scripts to the second node. We used scp on pazxxt05, as shown in the following example:

```
scp rac_scripts pazxxt06:rac_myscripts
```

Archived

5.2.3 Creating the raw devices

In order to use raw devices you need to create the nodes "/dev/raw/raw*n*", where *n* is the raw device number. Some systems have these nodes; if you do not have them, run the sample script. This sample script creates 100 nodes starting with "/dev/raw/raw1" and ending with "/dev/raw/raw100”.

Example 5-3 Creating raw devices

```
::::::::::::::
rac_make_raw
::::::::::::::
i=1
while [ $i != 100 ]; do
echo "mknod /dev/raw/raw$i c 162 $i"
mknod /dev/raw/raw$i c 162 $i
i=`expr $i + 1`
done
```

5.2.4 Create Oracle account

To create the oracle user, you can run yast2 or use the command **adduser**. Make sure you create a group for dba; we used dba for our group name.

5.2.5 Set up logical volumes

Before you create the logical volumes you need to create a volume group and assign dasd to it. You can use yast2 or **vgcreate** to create the volume group.

Review the Oracle documentation Oracle Real Application Clusters Installation and Configuration Guide 10g Release 1 (10.1) for AIX-Based Systems, Apple Mac OS X, hp HP-UX, hp Tru64 UNIX, Linux, Solaris Operating System, and Windows Platforms, Part No. B10766-08.

The section on Configuring Raw Partitions or Raw logical volumes on Linux in Chapter 5, “Pre-Installation Tasks for RAC on Linux,” describes the step for Linux on zSeries. After creating the volume groups on one node, we need to run vgchange and vgscan on the other cluster nodes.

Example 5-4 is a sample script.

Example 5-4 Script to create logical volumes

```
# lvcreate -L size -n lv_name vg_name

lvcreate -L 500M -n oracl_system_raw_500m VGraw
lvcreate -L 800M -n oracl_sysaux_raw_800m VGraw
```


Archived

```
lvcreate -L 500M -n oracl_undotbs1_raw_500m VGraw
lvcreate -L 500M -n oracl_undotbs2_raw_500m VGraw
lvcreate -L 250M -n oracl_temp_raw_250m VGraw
lvcreate -L 160M -n oracl_example_raw_160m VGraw
lvcreate -L 120M -n oracl_users_raw_120m VGraw
lvcreate -L 120M -n oracl_redo1_1_raw_120m VGraw
lvcreate -L 120M -n oracl_redo1_2_raw_120m VGraw
lvcreate -L 120M -n oracl_redo2_1_raw_120m VGraw
lvcreate -L 120M -n oracl_redo2_2_raw_120m VGraw
lvcreate -L 110M -n oracl_control1_raw_110m VGraw
lvcreate -L 110M -n oracl_control2_raw_110m VGraw
lvcreate -L 5M -n oracl_spfile_raw_5m VGraw
lvcreate -L 5M -n oracl_pwdfile_raw_5m VGraw
lvcreate -L 100M -n ora_ocr_raw_100m VGraw
lvcreate -L 100M -n ora_vote_raw_100m VGraw
lvcreate -L 500M -n oracl_asm1_500m VGraw
lvcreate -L 500M -n oracl_asm2_500m VGraw
lvcreate -L 500M -n oracl_asm3_500m VGraw
lvcreate -L 500M -n oracl_asm4_500m VGraw
```

5.2.6 Making symbolic links

When using raw devices for your database you need to create symbolic links to files on a file system on both nodes. The links will be used by DBCA when you create a database; look at dbca_map.

You must create the symbolic links on all nodes.

Example 5-5 Setting up symbolic links for raw devices

```
::::::::::::::
rac_link
::::::::::::::

ln -s /dev/raw/raw1 /oracle/10g/oradata/oracl_system_raw_500m
ln -s /dev/raw/raw2 /oracle/10g/oradata/oracl_sysaux_raw_800m
ln -s /dev/raw/raw3 /oracle/10g/oradata/oracl_undotbs1_raw_500m
ln -s /dev/raw/raw4 /oracle/10g/oradata/oracl_undotbs2_raw_500m
ln -s /dev/raw/raw5 /oracle/10g/oradata/oracl_temp_raw_250m
ln -s /dev/raw/raw6 /oracle/10g/oradata/oracl_example_raw_160m
ln -s /dev/raw/raw7 /oracle/10g/oradata/oracl_users_raw_120m
ln -s /dev/raw/raw8 /oracle/10g/oradata/oracl_redo1_1_raw_120m
ln -s /dev/raw/raw9 /oracle/10g/oradata/oracl_redo1_2_raw_120m
ln -s /dev/raw/raw10 /oracle/10g/oradata/oracl_redo2_1_raw_120m
ln -s /dev/raw/raw11 /oracle/10g/oradata/oracl_redo2_2_raw_120m
ln -s /dev/raw/raw12 /oracle/10g/oradata/oracl_control1_raw_110m
ln -s /dev/raw/raw13 /oracle/10g/oradata/oracl_control2_raw_110m
ln -s /dev/raw/raw14 /oracle/10g/oradata/oracl_spfile_raw_5m
```

Archived

```
ln -s /dev/raw/raw15 /oracle/10g/oradata/oracl_pwdfile_raw_5m
ln -s /dev/raw/raw16 /oracle/10g/oradata/ora_ocr_raw_100m
ln -s /dev/raw/raw17 /oracle/10g/oradata/ora_vote_raw_100m
```

Example 5-6 is the map that will be used by DBCA. You need this on the node where you run the DBCA.

Example 5-6 Map to be used by DBCA for data files

```
=============
rac_dbca_map
=============
system=/oracle/10g/oradata/oracl_system_raw_500m
sysaux=/oracle/10g/oradata/oracl_sysaux_raw_800m
undotbs1=/oracle/10g/oradata/oracl_undotbs1_raw_500m
undotbs2=/oracle/10g/oradata/oracl_undotbs2_raw_500m
temp=/oracle/10g/oradata/oracl_temp_raw_250m
example=/oracle/10g/oradata/oracl_example_raw_160m
users=/oracle/10g/oradata/oracl_users_raw_120m
redo1_1=/oracle/10g/oradata/oracl_redo1_1_raw_120m
redo1_2=/oracle/10g/oradata/oracl_redo1_2_raw_120m
redo2_1=/oracle/10g/oradata/oracl_redo2_1_raw_120m
redo2_2=/oracle/10g/oradata/oracl_redo2_2_raw_120m
control1=/oracle/10g/oradata/oracl_control1_raw_110m
control2=/oracle/10g/oradata/oracl_control2_raw_110m
spfile=/oracle/10g/oradata/oracl_spfile_raw_5m


pwdfile=/oracle/10g/oradata/oracl_pwdfile_raw_5m
```

Because of an Oracle bug in DBCA, we removed this last line (the pwdfile line). The password file is created in $ORACLE_HOME/dbs on both nodes.

Using ASM

If you use ASM you will need to add lines to your link script:

```
ln -s /dev/raw/raw18 /oracle/10g/oradata/oracl_asm1_500m
ln -s /dev/raw/raw19 /oracle/10g/oradata/oracl_asm2_500m
ln -s /dev/raw/raw20 /oracle/10g/oradata/oracl_asm3_500m
ln -s /dev/raw/raw21 /oracle/10g/oradata/oracl_asm4_500m
```

5.2.7 Binding the raw devices

To use raw devices you have to bind the node to the actual device or the logical volume.

Archived

The binding of the raw devices should be put in /etc/init.d so they are bounded before the CRS initialization happens on reboot.

You have to do this on each node.

Example 5-7 Binding the raw device

```
::::::::::::::
rac_rawbind
::::::::::::::
# raw /dev/raw/raw1 /dev/rawvg1/lv_sys

raw /dev/raw/raw1 /dev/VGraw/oracl_system_raw_500m
raw /dev/raw/raw2 /dev/VGraw/oracl_sysaux_raw_800m
raw /dev/raw/raw3 /dev/VGraw/oracl_undotbs1_raw_500m
raw /dev/raw/raw4 /dev/VGraw/oracl_undotbs2_raw_500m
raw /dev/raw/raw5 /dev/VGraw/oracl_temp_raw_250m
raw /dev/raw/raw6 /dev/VGraw/oracl_example_raw_160m
raw /dev/raw/raw7 /dev/VGraw/oracl_users_raw_120m
raw /dev/raw/raw8 /dev/VGraw/oracl_redo1_1_raw_120m
raw /dev/raw/raw9 /dev/VGraw/oracl_redo1_2_raw_120m
raw /dev/raw/raw10 /dev/VGraw/oracl_redo2_1_raw_120m
raw /dev/raw/raw11 /dev/VGraw/oracl_redo2_2_raw_120m
raw /dev/raw/raw12 /dev/VGraw/oracl_control1_raw_110m
raw /dev/raw/raw13 /dev/VGraw/oracl_control2_raw_110m
raw /dev/raw/raw14 /dev/VGraw/oracl_spfile_raw_5m
raw /dev/raw/raw15 /dev/VGraw/oracl_pwdfile_raw_5m
raw /dev/raw/raw16 /dev/VGraw/ora_ocr_raw_100m
raw /dev/raw/raw17 /dev/VGraw/ora_vote_raw_100m
raw /dev/raw/raw18 /dev/VGraw/oracl_asm1_500m
raw /dev/raw/raw19 /dev/VGraw/oracl_asm2_500m
raw /dev/raw/raw20 /dev/VGraw/oracl_asm3_500m
raw /dev/raw/raw21 /dev/VGraw/oracl_asm4_500m
raw /dev/raw/raw1 /dev/VGraw/oracl_system_raw_500m
```

5.2.8 Set up the /etc/host file

You should have three IP addresses for each node. In our case, we used an alias for the third IP address. This was our /etc/host file/.

Example 5-8 Sample of /etc/host entries

```
130.35.52.159   pazxxt06.us.oracle.com  pazxxt06 prv_pazxxt06
130.35.52.157   pazxxt05.us.oracle.com  pazxxt05 prv_pazxxt05
130.35.55.5     vip-pazxxt05
130.35.55.6     vip-pazxxt06
```

Archived

For simplicity in our testing we used the same IP for public and private addresses. However, this is not recommended for a production environment. You should use separate IP addresses for the public and private.

Review the network setup in Part II Section 5 in the Oracle RAC book - *Oracle Real Application Clusters Installation and Configuration Guide 10g Release 1 (10.1) for AIX-Based Systems, hp HP-UX PA-RISC (64-bit), hp Tru64 UNIX, Linux, Solaris Operating System (SPARC 64-bit), and Windows Platforms* Part No. B10766-08.

5.2.9 Set up ssh to work without password

Before you install and use Oracle Real Application clusters, you must configure secureshell (SSH) for the oracle user on all cluster nodes. The Oracle Installer uses the **ssh** and **scp** commands during installation to run remote commands on and copy files to the other cluster nodes. You must configure SSH so that these commands do not prompt for a password.

To configure SSH, complete the following steps on each cluster node:

1. Log in as the oracle user.
2. If necessary, create the .ssh directory in the oracle user's home directory and set the correct permissions on it:

```
$ mkdir ~/.ssh
$ chmod 755 ~/.ssh
```

Enter the following commands to generate an RSA key for Version 2 of the SSH protocol:

```
$ /usr/bin/ssh-keygen -t rsa
```

At the prompts, just press Enter.

Enter the following commands to generate a DSA key for Version 2 of the SSH protocol:

```
$ /usr/bin/ssh-keygen -t dsa
```

At the prompts, press Enter to accept the default.

This command writes the public key to the ~/.ssh/id_dsa.pub file and the private key to the ~/.ssh/id_dsa file. Never distribute the private key to anyone.

Copy the contents of the ~/.ssh/id_rsa.pub and ~/.ssh/id_dsa.pub files to the ~/.ssh/authorized_keys file on this node and to the same file on all other cluster nodes.

Archived

Change the permissions on the ~/.ssh/authorized_keys file on all cluster nodes:

```
$ chmod 644 ~/.ssh/authorized_keys
```

Note: The ~/.ssh/authorized_keys file on every node must contain the contents from all of the ~/.ssh/id_rsa.pub and ~/.ssh/id_dsa.pub files that you generated on all cluster nodes.

To test the SSH configuration, enter the following commands from the same terminal session, testing the configuration of each cluster node:

```
$ ssh nodename1 date
$ ssh nodename2 date
```

These commands should display the date set on each node. If any node prompts for a password or pass phrase, verify that the ~/.ssh/authorized_keys file on that node contains the correct public keys.

Note: The first time you use SSH to connect to a node from a particular system, you might see a message stating that the authenticity of the host could not be established. Enter `yes` at the prompt to continue. You should not see this message again when you connect from this system to that node.

If you see any other messages or text, apart from the date, the installation might fail. Make any changes required to ensure that only the date is displayed when you enter these commands.

You must make sure the disk can be written to from both nodes. Otherwise, installation on the remote nodes will fail. No error message will indicate this failure.

5.3 Preparation review

Follow Chapter 5 in O*racle Real Application Clusters Installation and Configuration Guide10g Release 1 (10.1) for AIX-Based Systems, hp HP-UXPA-RISC (64-bit), hp Tru64 UNIX, Linux, Solaris Operating System (SPARC 64-bit), and Windows Platforms,* Part No. B10766-05.

The steps are:

1. Log in to the system as root.
2. Check the hardware requirements.
3. Check the network requirements.
4. Check the software requirements.
5. Create required UNIX groups and user.

6. Configure Kernel parameters and shell limits.
7. Identify required software directories.
8. Identify or create an Oracle base directory.
9. Create the CRS home directory.
10. Choose a storage option for Oracle CRS, database, and recovery files.
11. Create directories for Oracle CRS, database, or recovery files.
12. Configure disks for automatic storage management.
13. Configure raw partitions.
14. Verify that the required software is running.
15. Stop existing Oracle processes.
16. Configure the Oracle users environment.

5.4 Oracle CRS installation

Oracle RAC is installed in several steps. The first is to install Cluster Ready Services (CRS). This is distributed as a separate CD image. We downloaded the file Linux_ship.crs.cpio.gz.

To expand the file we issued the commands:

```
gunzip Linux_ship.crs.cpio.gz
```

Extract the cpio archives with the command:

```
cpio -idmv < Linux_ship.crs.cpio
```

We found it best not to put this in the same directory as the database files.

Follow the steps in Chapter 9 in *Oracle Real Application Clusters Installation and Configuration Guide 10g Release 1 (10.1) for AIX-Based Systems, hp HP-UX PA-RISC (64-bit), hp Tru64 UNIX, Linux, Solaris Operating System (SPARC 64-bit), and Windows Platforms* Part No. B10766-08.

Check that the **ssh** command on each node is functioning by issuing the following commands on each node:

```
ssh pazxxt05 date
ssh pazxxto6 date
ssh prv_pazxxxt05 date
ssh prv_pazxxt06 date
```

Set up the DISPLAY variable

We used vncserver and viewer as our Xserver. Log on to pazxxt05 and run the vncserver as described in “Setting up an xWindows interface using VNC” on page 24, and as shown here:

```
orainst@pazxxt05:~> vncserver :4
```

```
New 'X' desktop is vmlinux9:4

Starting applications specified in /home/orainst/.vnc/xstartup
Log file is /home/orainst/.vnc/pazxxt05:4.log
```

Then set the DISPLAY variable by entering:

```
orainst@pazxxt05:~> export DISPLAY=pazxxt05:4
```

To install Oracle Cluster Ready Services we started the Oracle Universal Installer in the same way as if we were installing an Oracle10g Database. Refer to Chapter 3, “Installing Oracle 10g single instance” on page 21.

We needed the following information in the installation of CRS:

- Public node names - pazxxt05 and pazxxt06
- Private node names
- VIP node names
- Directory for the Oracle software /oracle/images/crs/Disk1
- Directory for OraInventory - /oracle/10g/OraInventory
- IP addresses from /etc/hosts file
- Location of the voting disk - /dev/raw/raw17
- Location of the OCR disk - /dev/raw/raw16

Note: We found that the voting disk must be larger than the documentation indicates. We made it 100 MB.

Go to the directory named CRS.

```
cd /oracle/crs
cd Disk1
./runInstaller
```

After you invoke the Oracle Universal Installer (OUI) using the command **./runInstaller**, the Welcome screen appears, and then on the following screens we entered our information for the CRS install.

Archived

Figure 5-2 For the first installation entry the inventory path

At this point you are asked to run the first of the root.sh scripts. You must go to a window that is logged on with the root user ID and run the root.sh script. This creates an Oracle Inventory Pointer File in /var/opt/oracle/oraInst.loc. This time the script is only run on the installing node.

Figure 5-3 Request to run root.sh

After the root.sh completes, return to the OUI and click **Next** to get the Specify File Locations panel. Check the values and click **Next**.

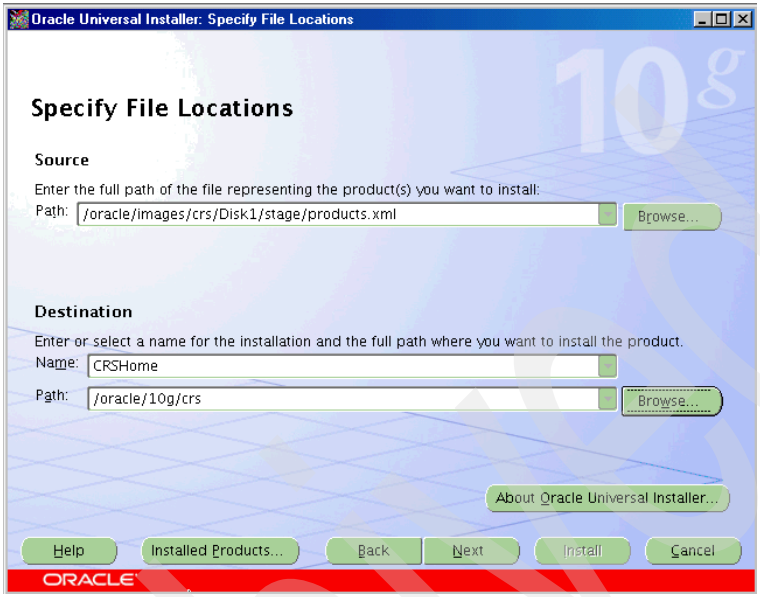


Figure 5-4 Enter the destination path

We chose the default of English.

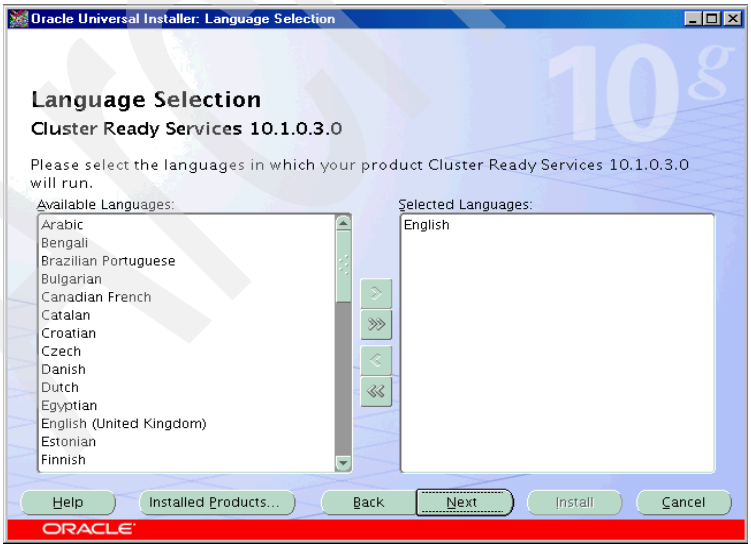


Figure 5-5 Specify language

Click **Next**.

On the next panel you enter the node names and IP addresses. This was our /etc/host file:

```
130.35.52.159   pazxxt06.us.oracle.com  pazxxt06 prv_pazxxt06
130.35.52.157   pazxxt05.us.oracle.com  pazxxt05 prv_pazxxt05
130.35.55.5     vip-pazxxt05
130.35.55.6     vip-pazxxt06
```

We entered the node name and the alias.

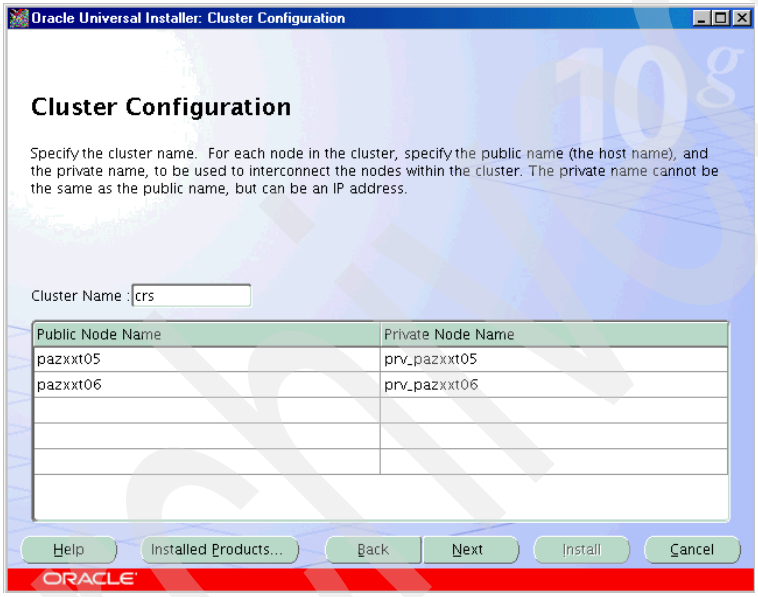


Figure 5-6 Enter the cluster nodes

Archived

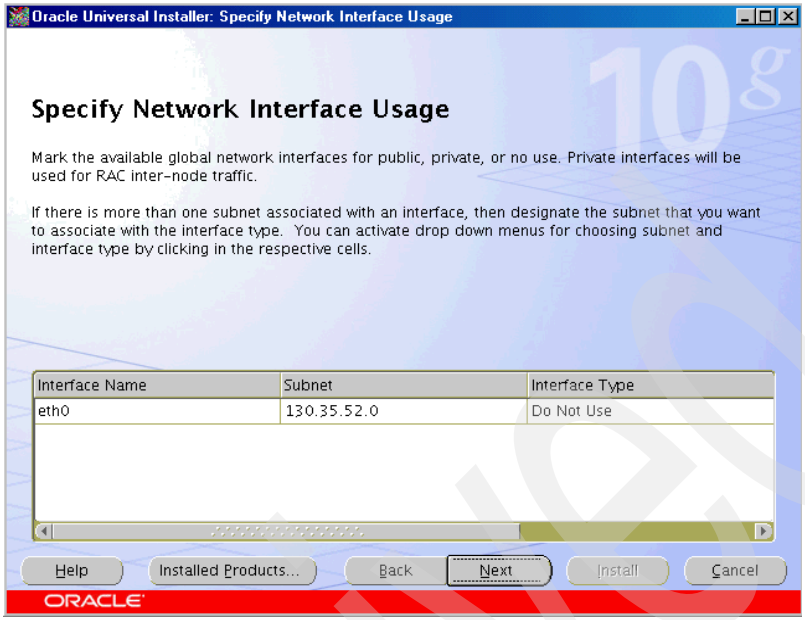


Figure 5-7 Specify the network interface usage

To verify the disks we could use, we issued the **lvscan** command.

Example 5-9 Output from LVSCAN

```
pazxxt05:~ # vgscan
vgscan -- reading all physical volumes (this may take a while...)
vgscan -- found active volume group "VG1"
vgscan -- found active volume group "VGraw"
vgscan -- "/etc/lvmtab" and "/etc/lvmtab.d" successfully created
vgscan -- WARNING: This program does not do a VGDA backup of your volume
groups

pazxxt05:~ #

pazxxt05:~ # lvscan
lvscan -- ACTIVE            "/dev/VG1/local" [13 GB]
lvscan -- ACTIVE            "/dev/VGraw/oracl_system_raw_500m" [500 MB]
lvscan -- ACTIVE            "/dev/VGraw/oracl_sysaux_raw_800m" [800 MB]
lvscan -- ACTIVE            "/dev/VGraw/oracl_undotbs1_raw_500m" [500 MB]
lvscan -- ACTIVE            "/dev/VGraw/oracl_temp_raw_250m" [252 MB]
lvscan -- ACTIVE            "/dev/VGraw/oracl_example_raw_160m" [160 MB]
lvscan -- ACTIVE            "/dev/VGraw/oracl_users_raw_120m" [120 MB]
lvscan -- ACTIVE            "/dev/VGraw/oracl_undotbs2_raw_500m" [500 MB]
lvscan -- ACTIVE            "/dev/VGraw/oracl_control1_raw_110m" [112 MB]
lvscan -- ACTIVE            "/dev/VGraw/oracl_control2_raw_110m" [112 MB]
```

```
lvscan -- ACTIVE            "/dev/VGraw/oracl_spfile_raw_5m" [8 MB]
lvscan -- ACTIVE            "/dev/VGraw/oracl_pwdfile_raw_5m" [8 MB]
lvscan -- ACTIVE            "/dev/VGraw/ora_ocr_raw_100m" [100 MB]
lvscan -- ACTIVE            "/dev/VGraw/ora_vote_raw_20m" [100 MB]
lvscan -- ACTIVE            "/dev/VGraw/oracl_redo1_1_raw_120m" [120 MB]
lvscan -- ACTIVE            "/dev/VGraw/oracl_redo1_2_raw_120m" [120 MB]
lvscan -- ACTIVE            "/dev/VGraw/oracl_redo2_1_raw_120m" [120 MB]
lvscan -- ACTIVE            "/dev/VGraw/oracl_redo2_2_raw_120m" [120 MB]
lvscan -- ACTIVE            "/dev/VGraw/oracl_asm1_500m" [500 MB]
lvscan -- ACTIVE            "/dev/VGraw/oracl_asm2_500m" [500 MB]
lvscan -- ACTIVE            "/dev/VGraw/oracl_asm3_500m" [500 MB]
lvscan -- ACTIVE            "/dev/VGraw/oracl_asm4_500m" [500 MB]
lvscan -- ACTIVE            "/dev/VGraw/oracl_asm5_500m" [500 MB]
lvscan -- ACTIVE            "/dev/VGraw/oracl_asm6_500m" [500 MB]
lvscan -- ACTIVE            "/dev/VGraw/oracl_asm7_500m" [500 MB]
lvscan -- ACTIVE            "/dev/VGraw/oracl_asm8_500m" [500 MB]
lvscan -- 26 logical volumes with 20.57 GB total in 2 volume groups
lvscan -- 26 active logical volumes

pazxxt05:~ #
```

We used raw17 for the voting disk and raw16 for the OCR disk.

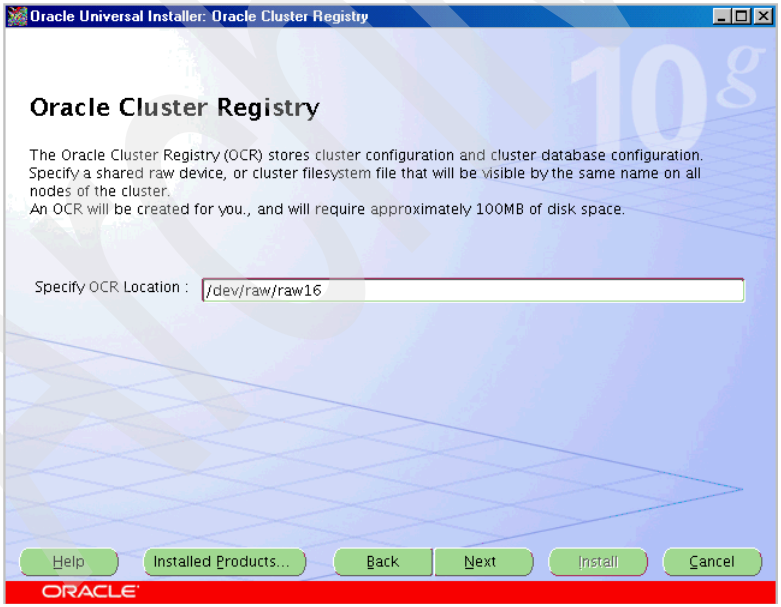


Figure 5-8 Cluster registry location

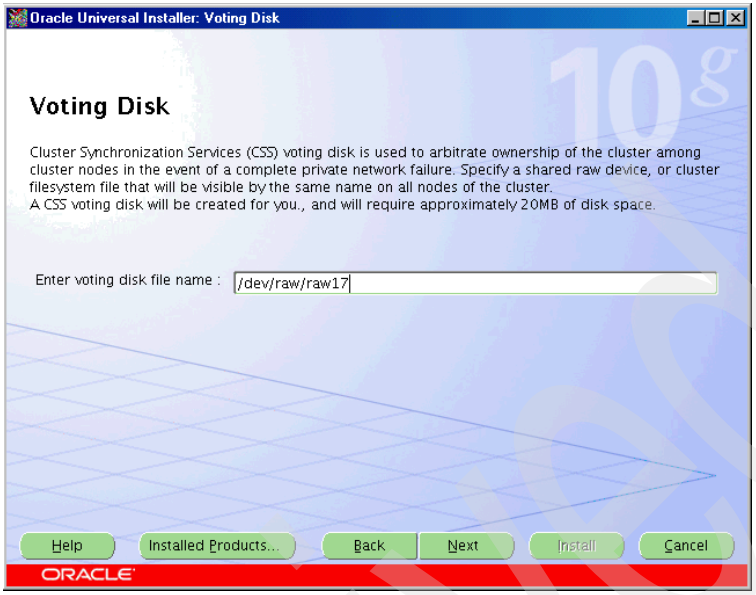


Figure 5-9 File name for voting disk

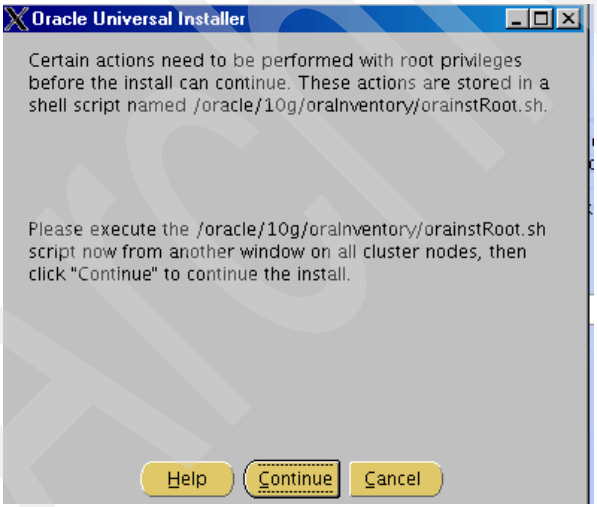


Figure 5-10 Request to run root.sh

Go to the telnet window that is logged on as root and run the orainstRoot.sh script. This time the root.sh must be run on both nodes.

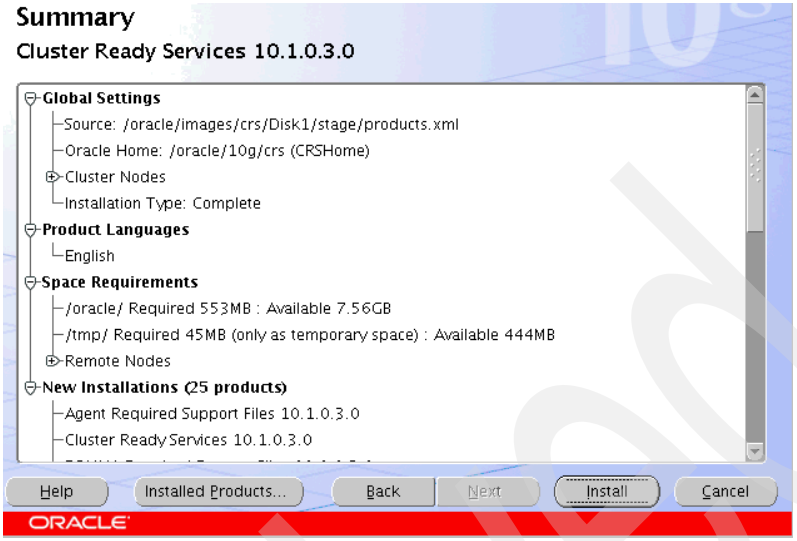


Figure 5-11 Summary for CRS

If you have any error messages on this panel you must resolve the errors before proceeding.

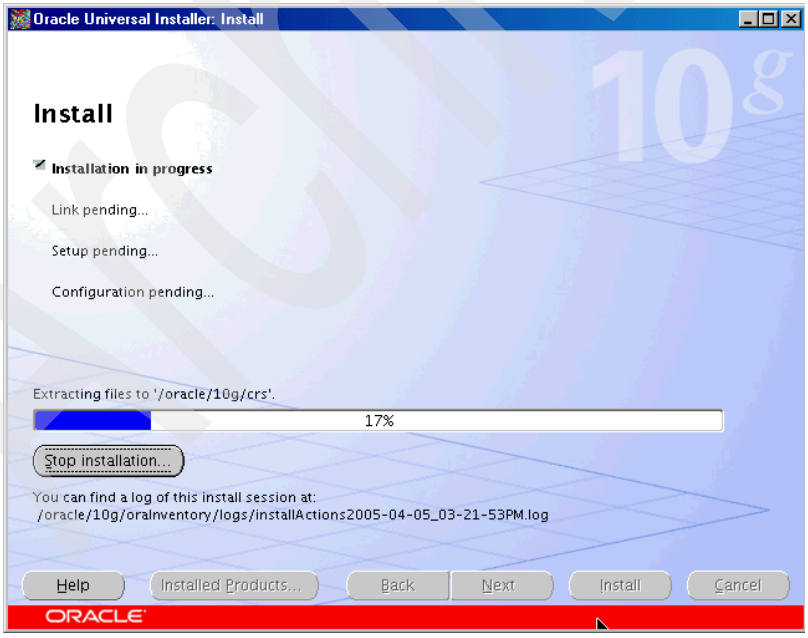


Figure 5-12 Installation of CRS

This installation takes a while to complete.

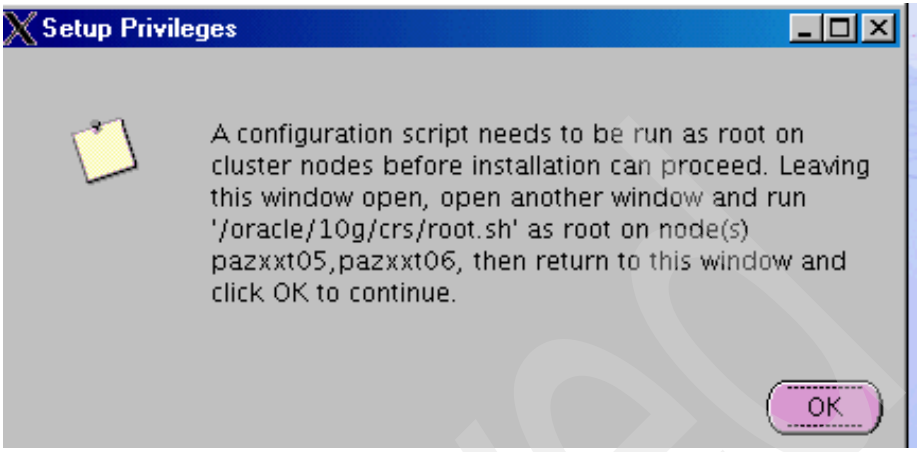


Figure 5-13 Configuration script

We started the root.sh script in parallel on both nodes and then we got the message that the CRS is active on both nodes.

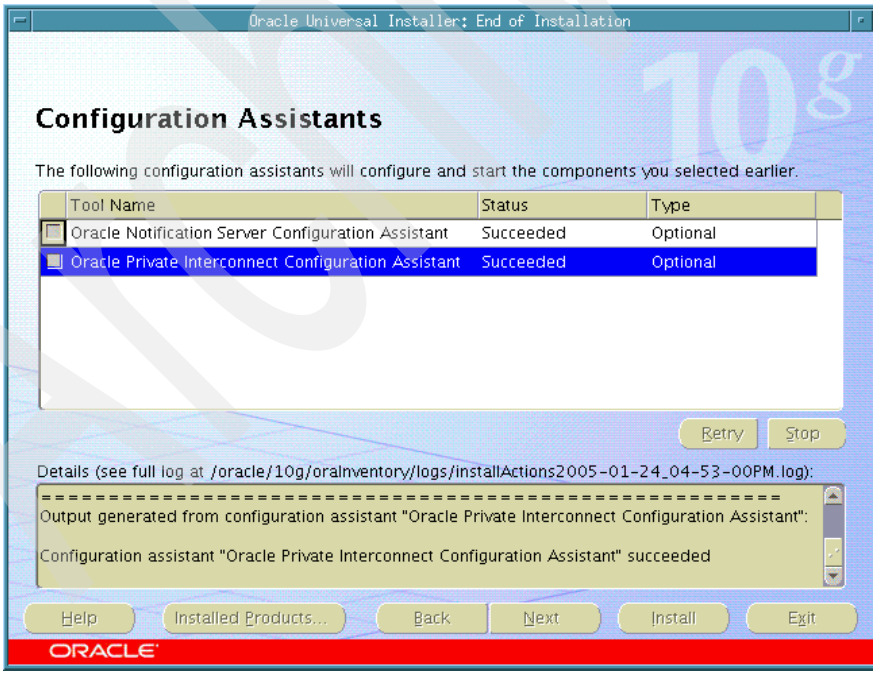


Figure 5-14 Configuration assistants

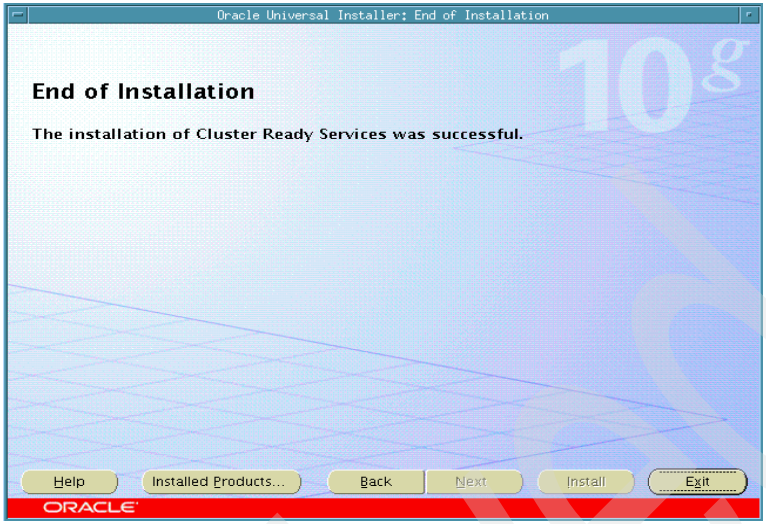


Figure 5-15 End of CRS installation

At this point, the CRS installation was complete and successful. We were ready to move on to the installation of RAC in “Oracle RAC installation” on page 101.

5.4.1 Cleaning up CRS if you need to reinstall

There is a metalink note on how to clean up your CRS installation if you need to restart. Review this for the latest information. It is Doc ID: Note:239998.1, Subject: 10gRAC: How to Clean Up After a Failed CRS Install.

We did a CRS deinstall using the OUI, then ran the following script (Example 5-10) to remove other files.

> **Note:** Do not include crs in the name of the script or it will kill itself. Do not remove the /var/opt/oracle directory if you have other Oracle instances on this Linux guest.

We found that we had to run a cleanup in two steps. If we ran the kill of the processes before we deleted the crs initialization scripts we would have system hang-ups. You must run these on both nodes.

Example 5-10 First script to clean up CRS

```
::::::::::::::
rac_remove 1
::::::::::::::
```

```
rm -f /etc/init.d/init.cssd
rm -f /etc/init.d/init.crs
rm -f /etc/init.d/init.crsd
rm -f /etc/init.d/init.evmd
rm -f /etc/init.d/rc2.d/K96init.crs
rm -f /etc/init.d/rc2.d/S96init.crs
rm -f /etc/init.d/rc3.d/K96init.crs
rm -f /etc/init.d/rc3.d/S96init.crs
rm -f /etc/init.d/rc5.d/K96init.crs
rm -f /etc/init.d/rc5.d/S96init.crs
rm -Rf /etc/oracle/scls_scr
rm -f /etc/inittab.crs
cp /etc/inittab.orig /etc/inittab
rm -rf /etc/oracle
rm -rf /var/opt/oracle
rm -rf /oracle/10g/crs/*
rm -rf /oracle/10g/db/*
rm -rf /oracle/10g/oraInventory/*
dd if=/dev/zero of=/dev/raw/raw17 bs=8192 count=2560
dd if=/dev/zero of=/dev/raw/raw16 bs=8192 count=12800
```

Example 5-11 Second script to clean up CRS

```
::::::::::::::
rac_remove 2
::::::::::::::
ps -ef|grep crs|grep -v grep|awk '{print $2}'|xargs kill -9
ps -ef|grep evm|grep -v grep|awk '{print $2}'|xargs kill -9
ps -ef|grep css|grep -v grep|awk '{print $2}'|xargs kill -9
```

Check that /etc/oracle/ocr.loc has been removed. In one case, we got an error message, as it had not been removed correctly.

After deinstalling and cleaning up these files and directories we were able to restart the installation of CRS.

5.5 Oracle RAC installation

Follow the steps in Chapter 10 in *Oracle Real Application Clusters Installation and Configuration Guide 10g Release 1 (10.1) for AIX-Based Systems, hp HP-UX PA-RISC (64-bit), hp Tru64 UNIX, Linux, Solaris Operating System (SPARC 64-bit), and Windows Platforms*, Part No. B10766-08.

Archived

We need to have downloaded the two DB disk images from otn.oracle.com and run the **gzip** and **cpio** commands. The files are:

```
Linux_ship.db.disk1.cpio.gz
Linux_ship.db.disk2.cpio.gz
```

We used the same ones we have used for the single instance.

```
cd /oracle/images
cd/Disk1
```

You start the OUI by issuing the command:

```
./runInstaller
```

The first panels are the Welcome screen, and then the Inventory panel and the install panel.

The next panel for the RAC installation is shown in Figure 5-16.

Figure 5-16 Select the cluster nodes

Because CRS is already installed, the OUI recognizes that we are installing RAC and presents the names of the nodes. You have to select the second node and click the **Next** button.

On this next panel you choose the type of installation or license you want to install.

Figure 5-17 Select installation type

Then in Figure 5-18, we choose to not create a stater database at this point. We will use the DBCA command later to do this. We just want to install the Oracle RAC code at this point.

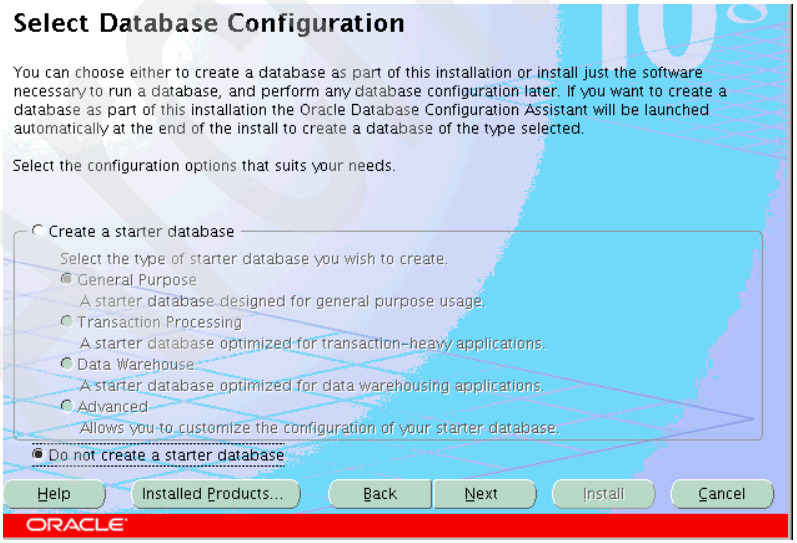


Figure 5-18 Select database configuration

If you choose to create a starter database at this time, you go through a series of panels that asks you to supply the information about the database instance, as we describe in “Oracle Database creation” on page 110.

On the Summary panel, check that there are no error messages before clicking the **Install** button.

Figure 5-19 Summary of the RAC installation

Click **Next** to proceed with the installation.

Archived

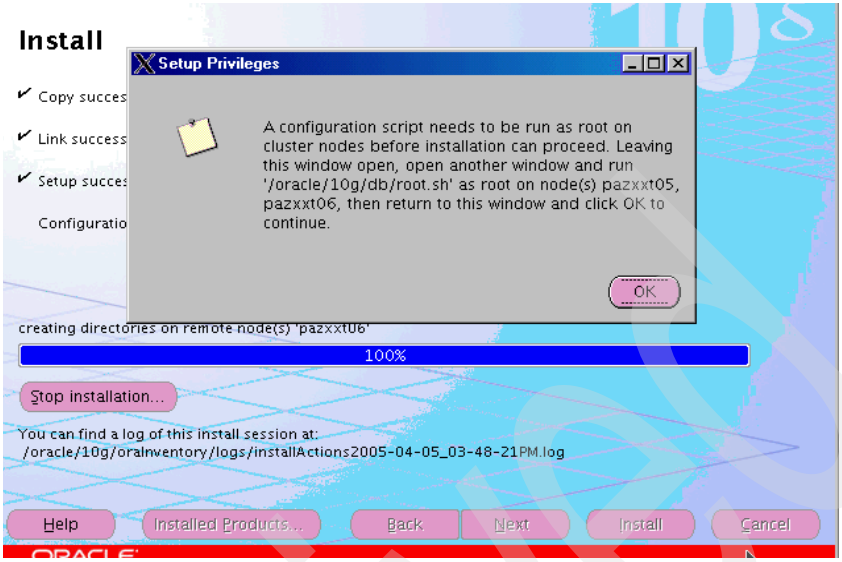


Figure 5-20 Running the root.sh script

The install panel will take some time to complete. During the installation, you are prompted to go the telnet session as user ID root and run the root.sh script on both nodes.

In this case, the DISPLAY variable must be set in the root telnet session. This must done on the vnc screen, not the putty screen, as it uses the xWindows interface. For example:

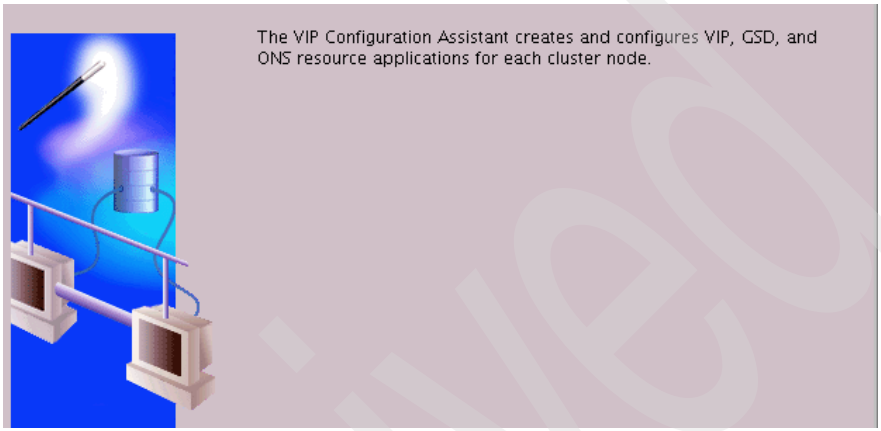
```
export DISPLAY=pazxxt05:4
xhost +
```

Note: Because we were logged on as user ID oracle when we started the vncserver session, we had to issue the **xhost +** command. If we log on as another user and then **su** to oracle and **su** to root you only do **xhost +** once at the beginning.

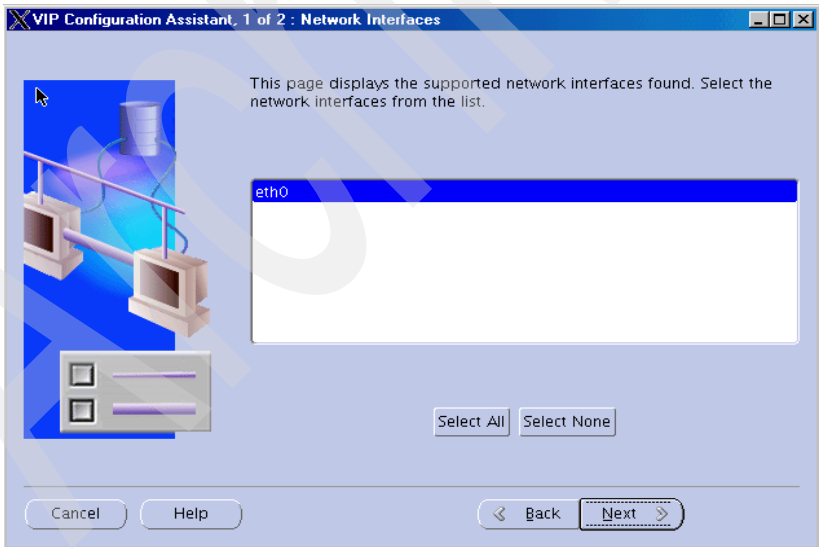
Let the root.sh script run to completion on the first node before starting to run it on the second node. During the running of the script on the first node, the VIP configuration script will be run, which is shown in “VIP configuration” on page 106.

5.5.1 VIP configuration

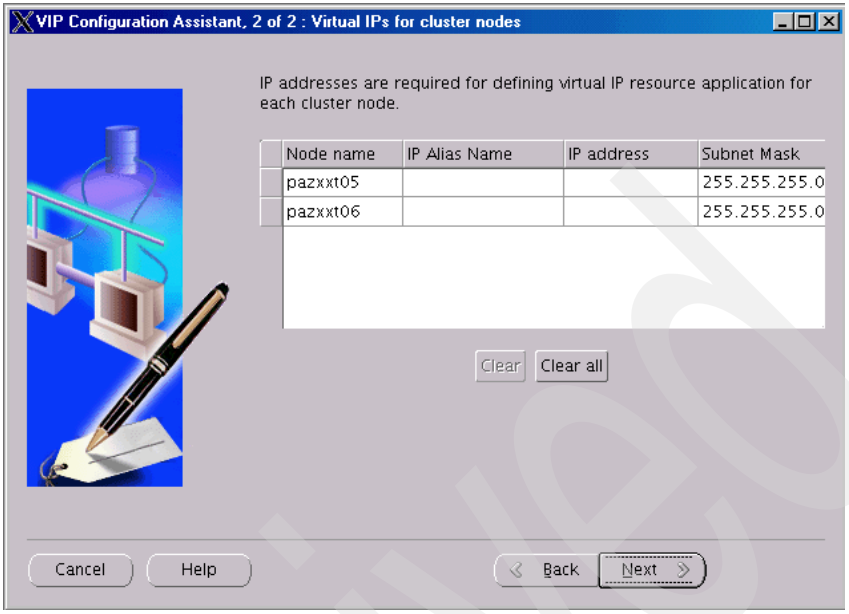
During the running of the root.sh script on the first node, the VIP configuration will be executed. If you wait for the root.sh to finish on the first node, this will only be run once. The following 10 panels show the steps of the VIP configuration process.



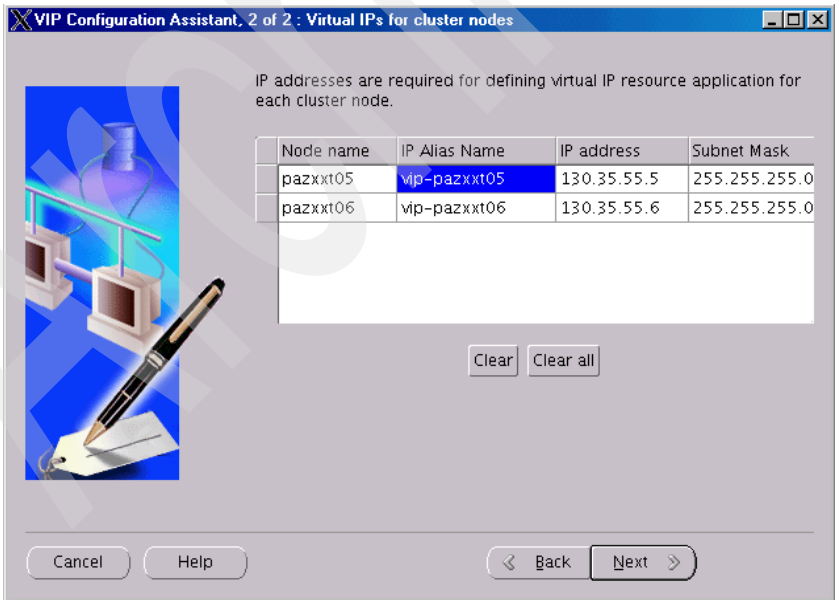
The VIP configuration will be started automatically.



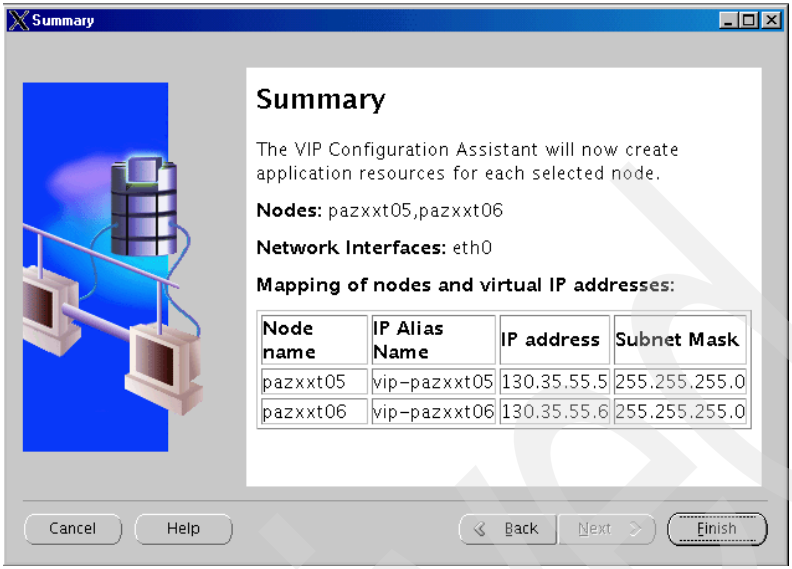
Archived

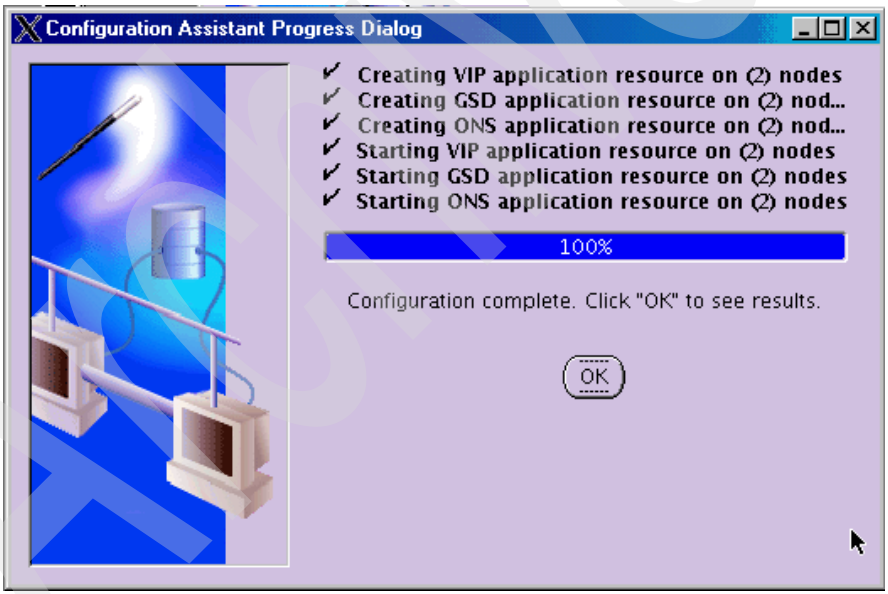


After entering the first IP alias name, it was able to pull out the rest of the information.

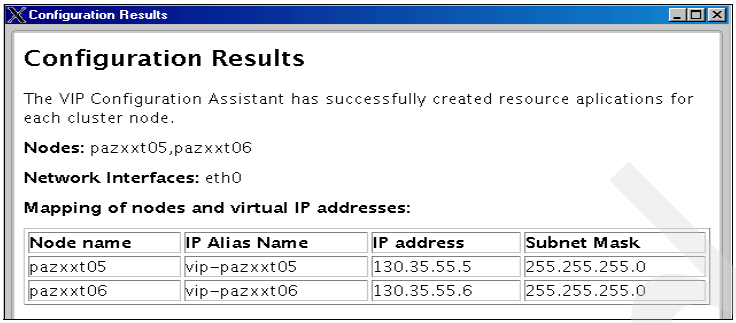


Archived


Summary
Summary
The VIP Configuration Assistant will now create application resources for each selected node.
Nodes: pazxxt05,pazxxt06
Network Interfaces: eth0
Mapping of nodes and virtual IP addresses:
Node name | IP Alias Name | IP address | Subnet Mask
pazxxt05 | vip-pazxxt05 | 130.35.55.5 | 255.255.255.0
pazxxt06 | vip-pazxxt06 | 130.35.55.6 | 255.255.255.0
Cancel
Help
Back
Next
Finish


Configuration Assistant Progress Dialog
Creating VIP application resource on (2) nodes
Creating GSD application resource on (2) nod...
Creating ONS application resource on (2) nod...
Starting VIP application resource on (2) nodes
Starting GSD application resource on (2) nodes
Starting ONS application resource on (2) nodes
100%
Configuration complete. Click "OK" to see results.
OK

Archived



This is the end of the VIP configuration. Click **Exit**. You will be taken back to the Installation process and receive the End of Installation panel.

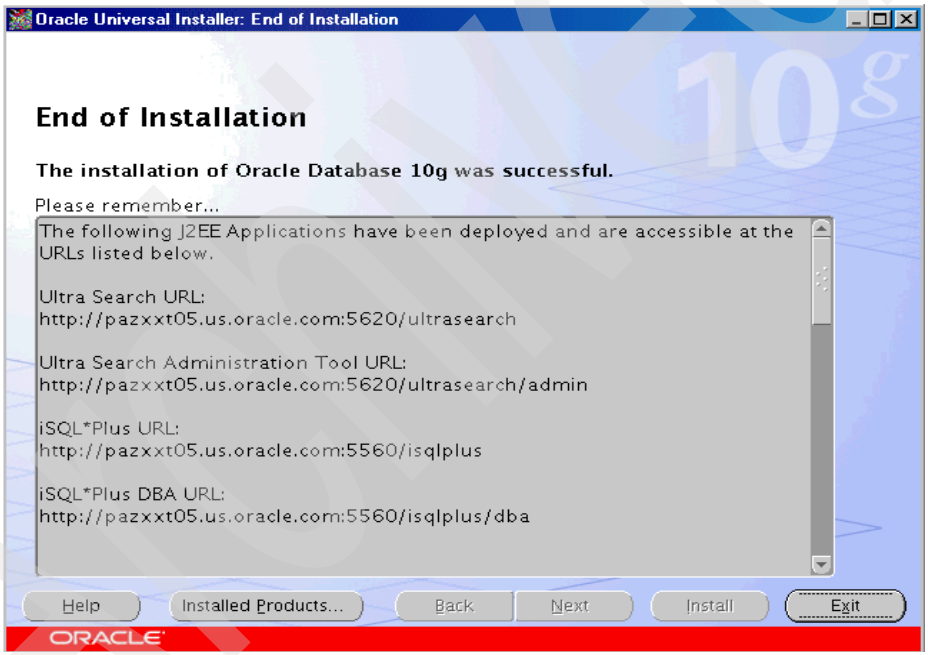


Figure 5-21 End of Oracle RAC Database installation

If you scroll down this panel, you will see the information about Oracle Enterprise Manager.

The RAC software in now installed. You are ready to create a database instance as described in “Oracle Database creation” on page 110.

5.6 Oracle Database creation

You are now ready to create a database using the **dbca** command.

First set up the .profile for user ID oracle.

```
export ORACLE_HOME=/oracle/O10g
export ORACLE_SID=orcl
export PATH=$PATH:$ORACLE_HOME/bin
echo ORACLE_HOME=$ORACLE_HOME
echo ORACLE_SID=$ORACLE_SID
export DISPLAY=pazxxt05:4
```

Follow the steps in Chapter 11 in *Oracle Real Application Clusters Installation and Configuration Guide 10g Release 1 (10.1) for AIX-Based Systems, hp HP-UXPA-RISC (64-bit), hp Tru64 UNIX, Linux, Solaris Operating System (SPARC 64-bit), and Windows Platforms* Part No. B10766-05

We chose orcl for our SID name. The instance names will be orcl1 and orcl2.

Before you issue the **dbca** command, ensure your DISPLAY variable is set for user ID oracle.

```
export DISPLAY=pazxxt05:4
dbca
```

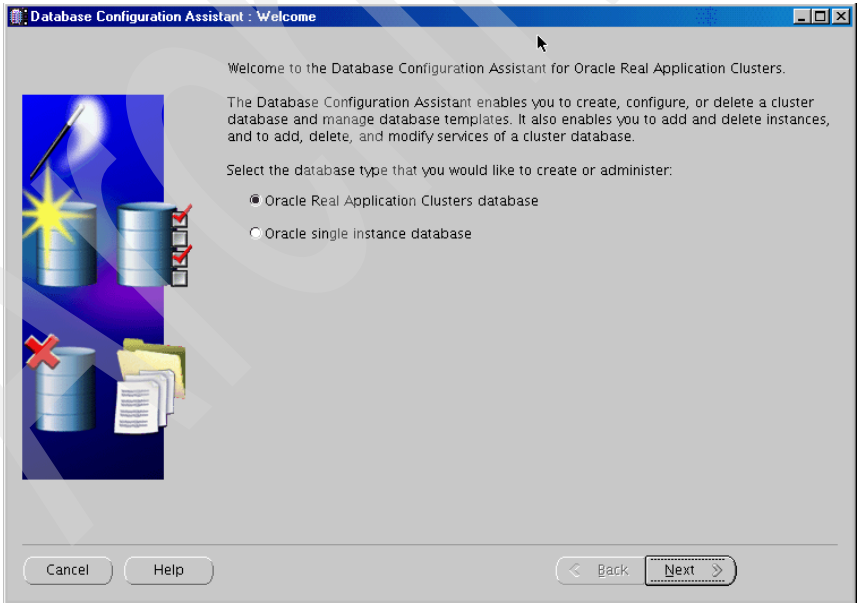


Figure 5-22 Welcome screen for DBCA process

Choose create a database, then click **Next**.

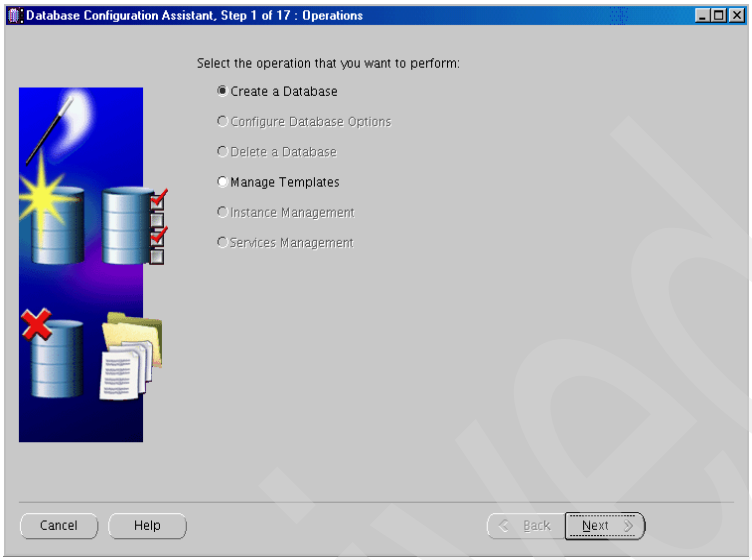


Figure 5-23 Select the DBCA operation

We were using DBCA to create our RAC database. Select the nodes.

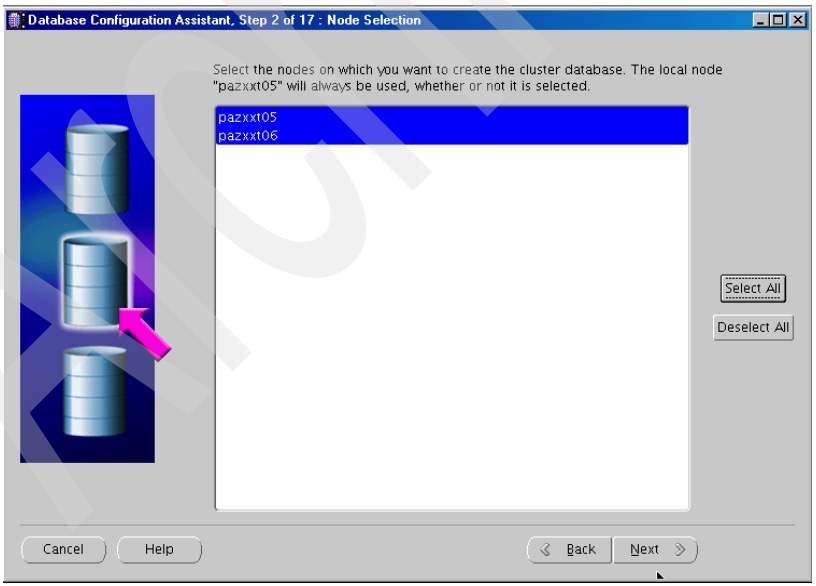


Figure 5-24 Select the RAC nodes

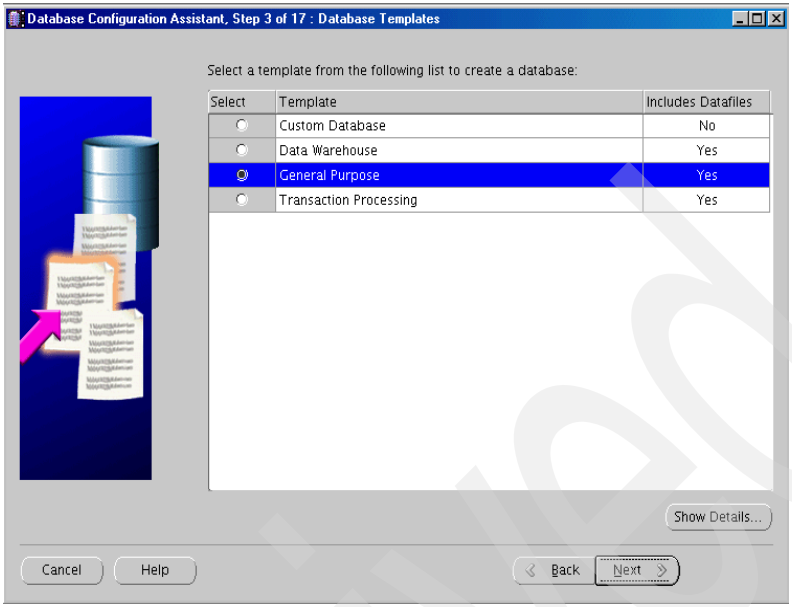


Figure 5-25 Select the database template for DBCA to use

We had created our map in the file /home/oracle/dbca_map.

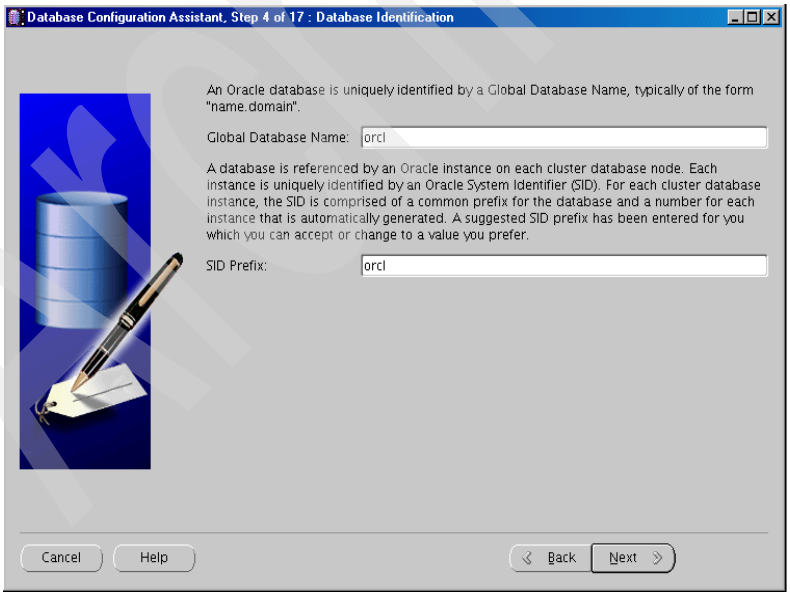


Figure 5-26 SID same

We used the SID name of orcl.

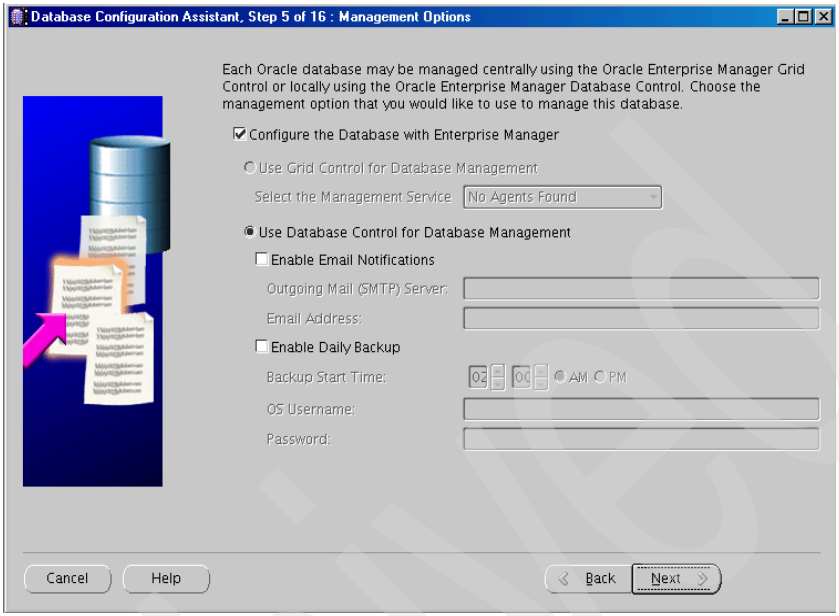


Figure 5-27 Choosing OEM or GRID manager

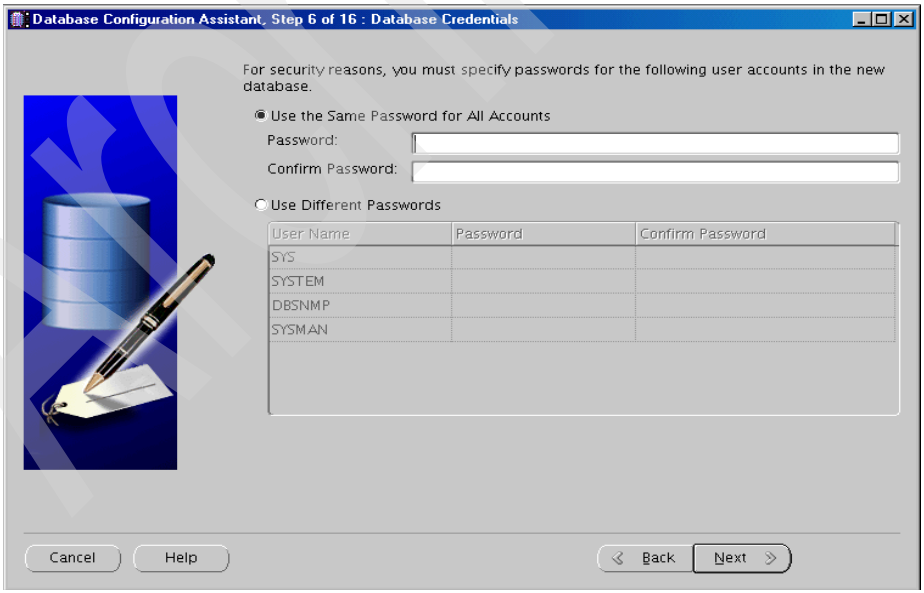


Figure 5-28 Enter password information

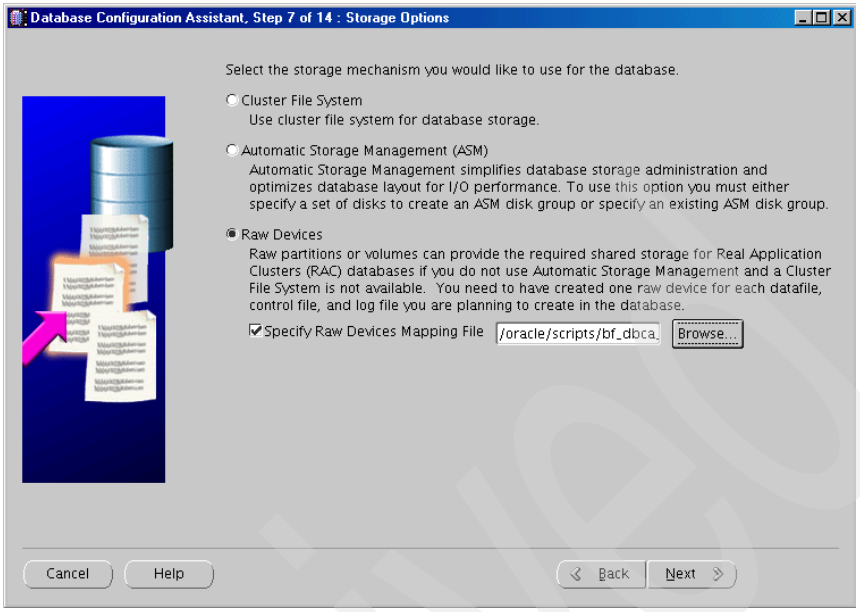


Figure 5-29 Choose storage option

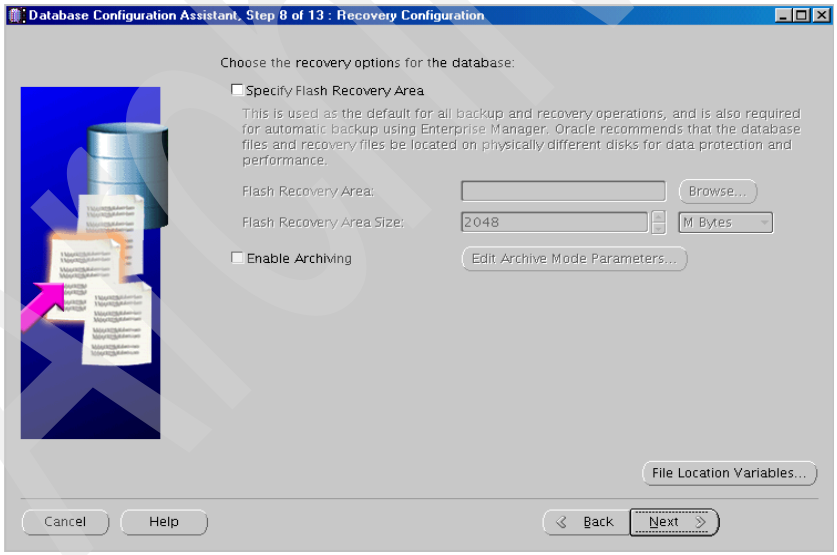


Figure 5-30 Chose recovery option

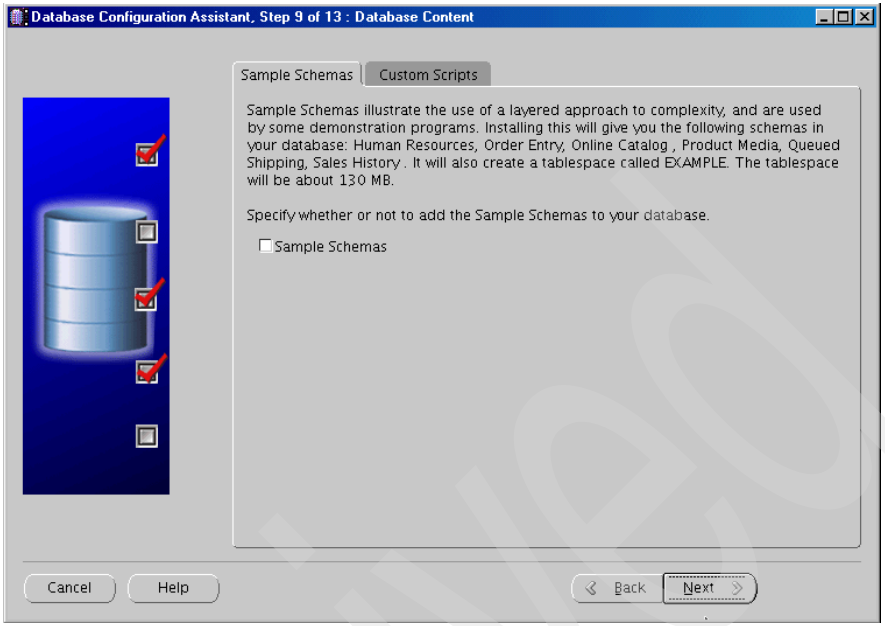


Figure 5-31 Using custom scripts

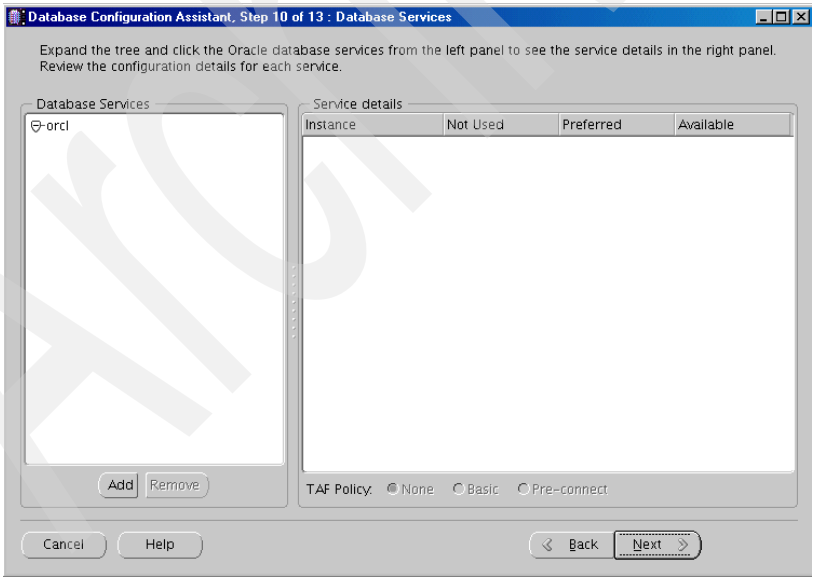


Figure 5-32 Expand the service information

Archived

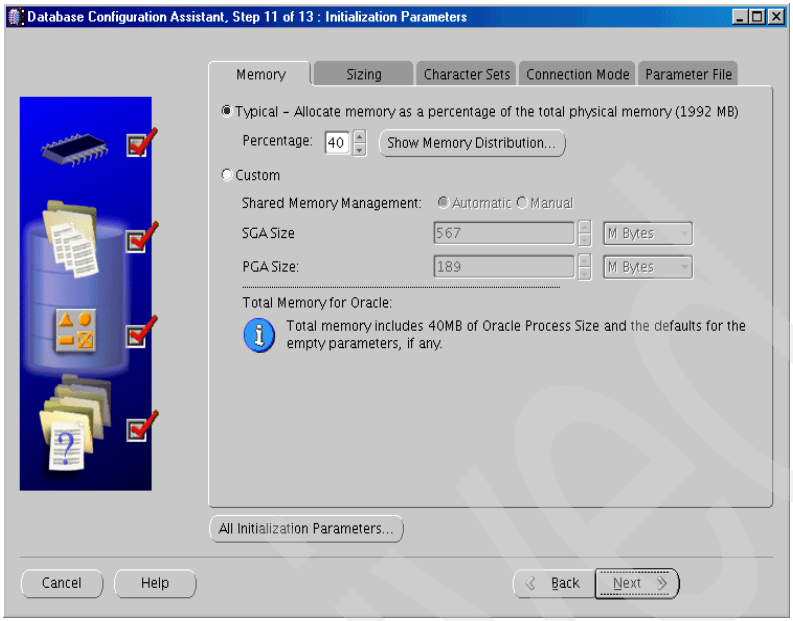


Figure 5-33 Validate the memory for SGA and PGA

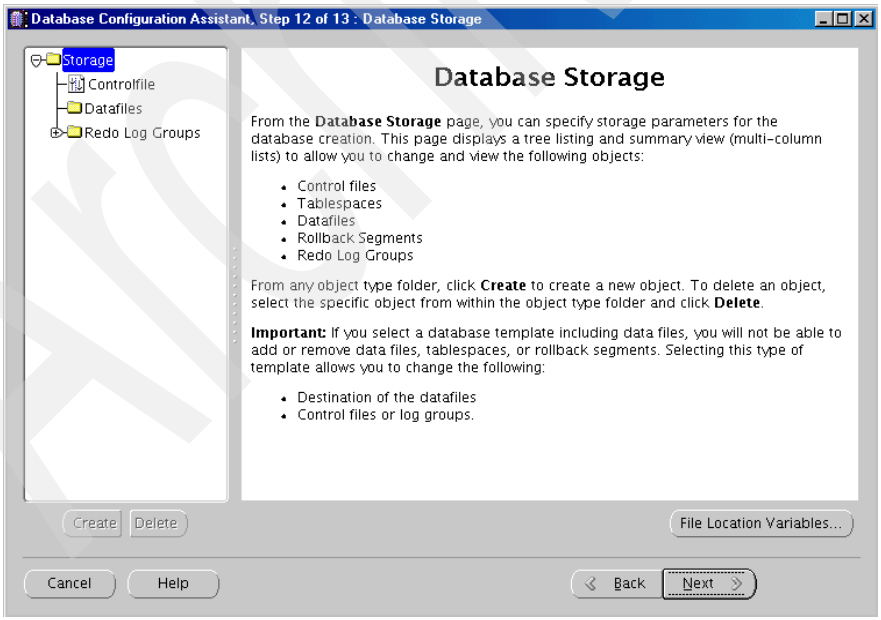


Figure 5-34 Check file locations

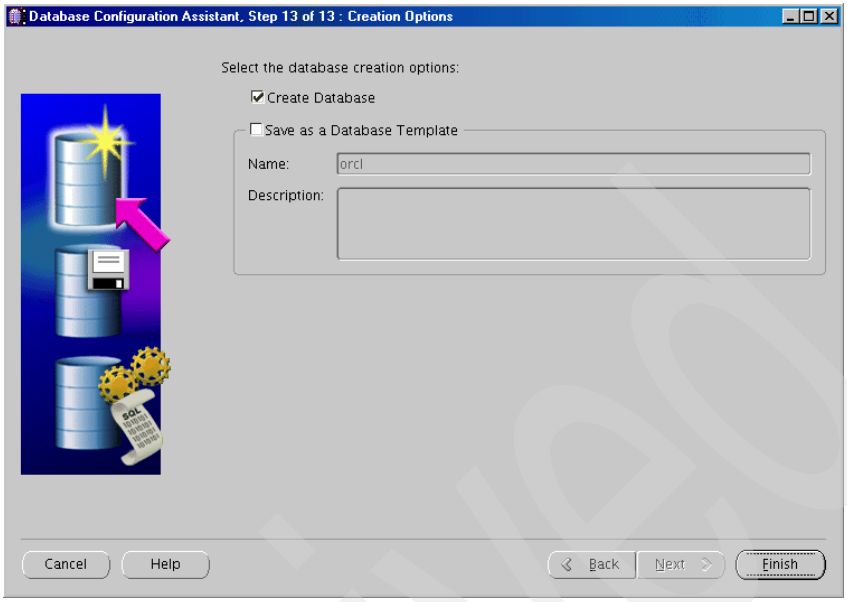
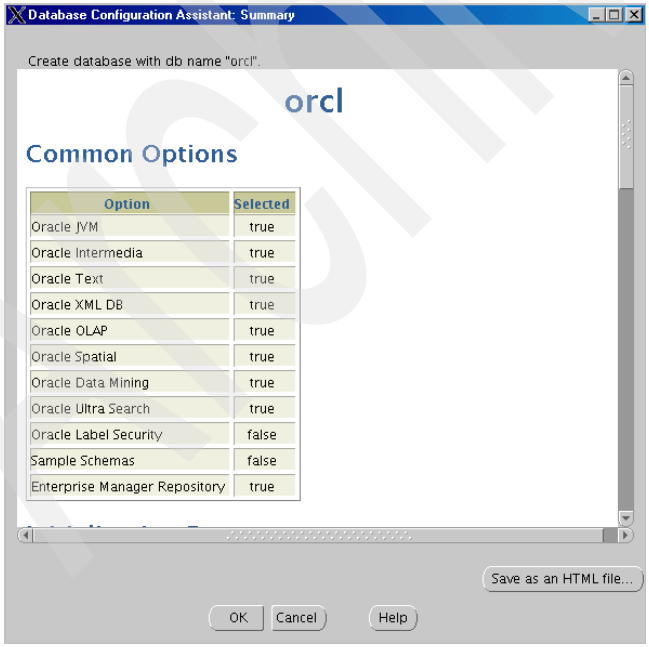


Figure 5-35 Select database creation options



Option	Selected
Oracle JVM	true
Oracle Intermedia	true
Oracle Text	true
Oracle XML DB	true
Oracle OLAP	true
Oracle Spatial	true
Oracle Data Mining	true
Oracle Ultra Search	true
Oracle Label Security	false
Sample Schemas	false
Enterprise Manager Repository	true

Figure 5-36 Review the database options

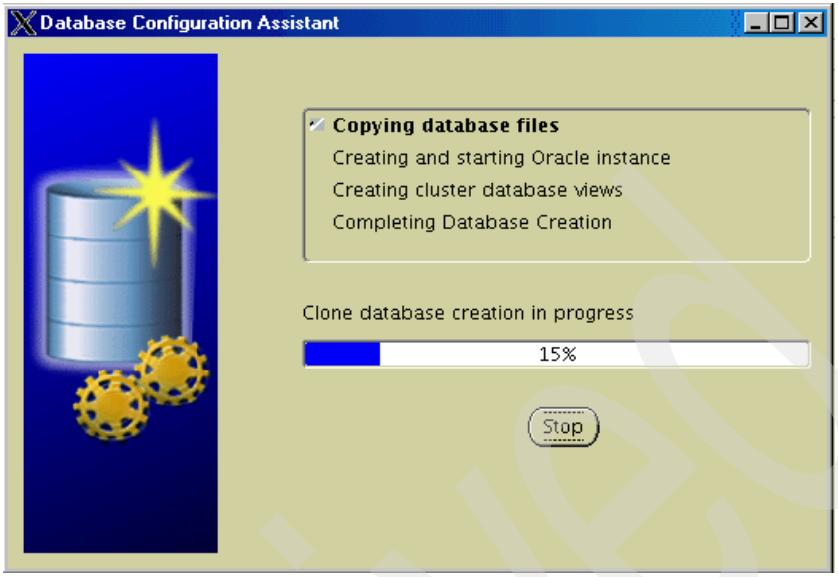


Figure 5-37 Install panel

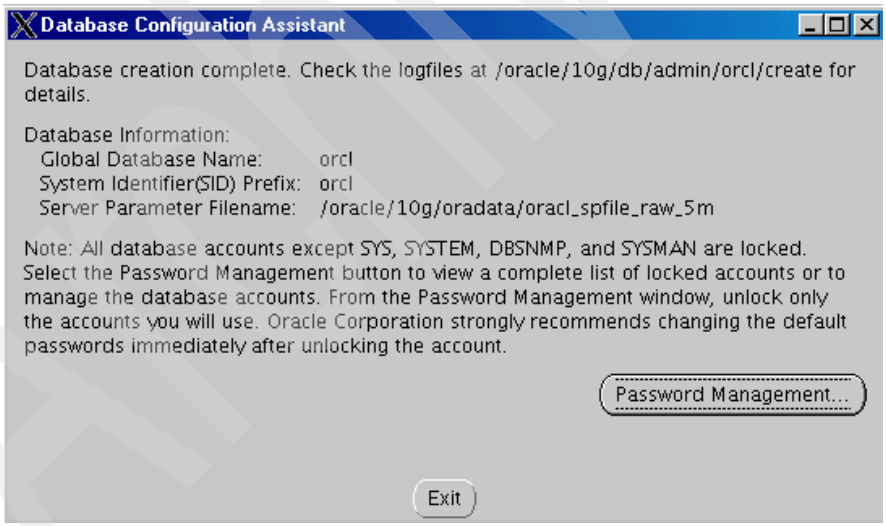


Figure 5-38 End of database creation panel

The database is now created with the SID of orcl. On the first node the instance orcl1 is running. On the second node the instance orcl2 is running.

5.6.1 Setting up the user profile

To access this database, users must set up their profile with the following variables:

- ORACLE_HOME=/oracle/10g
- ORACLE_SID=orcl
- PATH=$PATH:$ORACLE_HOME/bin

You can verify the database is up and running by the same steps we described in “Verifying that the database is running” on page 49.

You can use OEM to manage the database.

Archived

Archived

6

Using Tivoli Storage Manager and Tivoli Data Protect for Oracle Database 10g

This chapter is an introduction to the Tivoli Storage Manager (TSM) and Tivoli Data Protect for Oracle (TDPO) products. We demonstrate the installation process for both products on the Linux on zSeries environment. The TSM server will also run in z/VM and z/OS.

Archived

6.1 IBM Tivoli Storage Manager overview

IBM Tivoli Storage Manager provides automated, policy-based, distributed data and storage management for file servers and workstations in an enterprise network environment. The base functions provided by Tivoli Storage Manager include:

- Backup and Restore

  The backup process creates a copy of the file or application data that can be recovered if the original data is lost or destroyed. Backups can be scheduled, performed manually from the Tivoli Storage Manager client interface, or performed remotely using a Web-based interface. The restore process transfers a backup data copy from Tivoli Storage Manager server-managed storage onto a designated machine.

- Archive and Retrieval

  The archive process creates a copy of a file or a set of files and stores it as a unique object for a specified period of time. This function is useful for maintaining copies of vital records for historical purposes. Like the backup process, the archive process can be scheduled, performed manually from the Tivoli Storage Manager client interface, or performed remotely using a Web-based interface.

- Instant Archive and Rapid Recovery

  IBM Tivoli Storage Manager allows for the creation of a complete set of client files, called a backup set, on the Tivoli Storage Manager server system using the most recent backup versions stored by the server. In a process called Instant Archive, a backup set is used to retain a snapshot of a client file system for a designated period of time.

- Space Manager Client

  This feature provides for the automatic and transparent movement of operational data from a client system to server-managed storage. This process, called Hierarchical Space Management (HSM), is implemented as a client installation and controlled by policy defined to the Tivoli Storage Manager server. HSM frees up space on a client machine by using distributed storage media as a virtual hard drive for that machine.

Archived

6.2 Tivoli Storage Manager architecture

The Tivoli Storage Manager server component is installed on the computer that manages storage devices. The Tivoli Storage Manager server provides the following functions:

- Data management
- Storage device and media management
- Reporting and monitoring functions
- System security

The Tivoli Storage Manager server application is supported by a relational database that is specifically designed to manage a data storage environment. The server database operates transparently, requiring minimal administrative oversight. The server relies on the database to maintain an inventory of metadata associated with stored data objects.

The database is not used to store actual client data, which is maintained in server-managed storage. All database transactions are written to an external log file called the recovery log. The recovery log can be used to restore the database if necessary.

Tivoli Storage Manager server operations are configured, controlled, and monitored using graphical or command-line interfaces. Some tasks can be performed several different ways, so the interface you use depends on the type of task and your preferences. Support for SQL SELECT statements and ODBC data transfer is also available for advanced database management and reporting.

The Tivoli Storage Manager server uses the database to intelligently map business goals with storage management policies and procedures. The Tivoli Storage Manager server tracks the origin and location of each client data copy. Policies defined to the Tivoli Storage Manager server determine how data copies will be stored, migrated, and eventually replaced with newer data. Tivoli Storage Manager typically maintains several incrementally modified versions of client data files, up to a maximum number defined by the administrator.

6.3 Tivoli Data Protection for Oracle

Data Protection for Oracle currently supports Oracle8i (8.1.7) and Oracle 9*i* (9.0.1 or 9.2) and Oracle Database 10g Databases with the Oracle Recovery Manager (RMAN).

The TDP for the Oracle application client provides an integrated solution for performing full backup and restore operations on Oracle databases. It is a client

Archived

application that provides full backup of online databases and restore of full databases to the original or a different location.

TDP for Oracle is not intended as a substitute for the standard Tivoli Storage Manager backup/archive client. TDP for Oracle cannot be used to back up or restore any non-database data, such as history files or any other system configuration files. Those files need to be backed up by the Tivoli Storage Manager backup/archive client. Therefore, the two client types work together to provide full data protection for your Oracle environment.

The TDP for Oracle application client and the Tivoli Storage Manager backup/archive client can run simultaneously on the same Oracle server; however, they are totally separate clients as far as the Tivoli Storage.

6.4 RMAN and Tivoli Data Protection for Oracle

RMAN provides consistent and secure backup, restore, and recovery performance for Oracle databases. While the Oracle RMAN initiates a backup or restore, Data Protection for Oracle acts as the interface to the Tivoli Storage Manager Server Version 5.1.0 (or later). The Tivoli Storage Manager Server then applies administrator-defined storage management policies to the data. Data Protection for Oracle implements the Oracle-defined Media Management application program interface (API) 2.0. This API interfaces with RMAN and translates Oracle commands into Tivoli Storage Manager API calls to the Tivoli Storage Manager Server. With the use of RMAN, Data Protection for Oracle allows you to perform the following functions:

- Full backup function for the following while online or offline:
  - Tablespaces
  - Datafiles
  - Archive log files
  - Control files
- Full database restores while offline
- Tablespace and datafile restore while online or offline

6.5 Overview of installation process of TSM and TDPO

Oracle Database 10g is a 64bit database. At the time that Oracle 10g was made available, TDPO for Linux on zSeries was still 31bit, thus supporting only Oracle9*i*. We were able to receive a beta copy of the 64-bit TDPO product to test. We at least wanted to make sure it would install and perform basic functions for possible early users.

The following documents were used for this chapter:

- *IBM Tivoli Storage Manager for Linux, Quick Start Version 5.2*
- *Data Protection for Oracle for UNIX Installation and User's Guide Version 5.2*
- *Database Backup and Recovery Advanced User's Guide for 10g Release 1* (10.1) Document, B10734-01
- *Database Recovery Manager Reference Guide Document*, B10770-01

The process to get all this installed and working is a bit intricate and complicated. As a beginning it would be easiest if I described what the completed process yields and then provide a high-level installation process and then the details.

At the completion of the installation and configuration processes, we had two Linux virtual machines that were configured as follows:

- Linux22 had the following installed:
  - SuSE SLES8 at 2.4.21-251
  - Tivoli Storage Manager Version 5.2.3

    All defaults were used for storage management classes. Please note that setting up management classes is a very complicated topic and beyond the scope of this chapter. Please refer to the TSM documents for more detailed information on this topic.
  - Oracle Database 10g (10.1.0.3)
  - The Oracle database was set up as follows:
    - Archiving enabled
    - rmgr/rman user ID created for RMAN
    - Database configured as a RMAN catalog to track data about backups created
- Linux23 had the following installed:
  - SuSE SLES8 at 2.4.21-251
  - Tivoli Data Protect for Oracle (TDPO) Version 5.2.3
  - Oracle Database 10g (10.1.0.3)

    Archiving mode enabled (required to perform backups with RMAN and TDPO)

The sequence used was:

1. We installed Oracle Database 10g on both virtual machines. The installation process is the same as covered earlier in this book.
2. We created the RMAN user (rmgr) and RMAN catalog in the database on Linux22.

Archived

3. We enabled archiving on both databases.
4. We installed TSM on Linux22 and configured the options files. Then using the Web Admin screens, we completed the configuration process.
5. We installed TDPO on Linux23 and configured the options files. We registered the client node to the TSM server, then ran the tdpconf utility to change the password and insure the installation was correct.
6. We created two RMAN scripts, one to back up the user's tablespace and one to restore the same tablespace.
7. We executed the scripts from RMAN to insure everything worked as it was supposed to.

The next sections start with the second step from the above process, which is after the databases have been installed.

6.5.1 Configuring RMAN

To set up RMAN and the RMAN catalog to track Oracle backups we needed to created an ID for RMAN and give it the necessary privileges and then create a catalog in the database on this guest.

Working on Linux22, the following commands were entered in SQLPlus to create the user rmgr and grant it the proper privileges:

```
SQL> CREATE USER rmgr IDENTIFIED BY rman TEMPORARY TABLESPACE temp
2 DEFAULT TABLESPACE users QUOTA UNLIMITED ON users;
User created.
SQL> GRANT RECOVERY_CATALOG_OWNER, CONNECT, RESOURCE TO rmgr;
Grant succeeded.
```

Then we exited SQLPlus and went to the $ORACLE_HOME/bin directory to start the RMAN client.

```
oracle@/oracle/product/db10g/bin
```

And entered the following command:

```
./rman
Copyright (c) 1995, 2002, Oracle Corporation. All rights reserved.
RMAN> connect catalog rmgr/rman@rmancat
connected to recovery catalog database
recovery catalog is not installed
```

Now create the recovery catalog:

```
RMAN> create catalog;
recovery catalog created
```

Once the user ID was created for RMAN and the catalog was installed, we enabled archiving on both databases. Since the process is the same for both, we only cover the process for enabling it on the catalog database. The same commands are used on the test database on Linux 23.

We entered the following Oracle commands:

```
SQL> startup mount
ORACLE instance started.

Total System Global Area  595591168 bytes
Fixed Size                  1323152 bytes
Variable Size             170381168 bytes
Database Buffers          423624704 bytes
Redo Buffers                 262144 bytes
Database mounted.
SQL> alter database archivelog;

Database altered.

SQL> alter database open;

Database altered.

SQL> exit
```

Then in the init.ora we made the following additions;

```
log_archive_format                  string      %t_%s_%r.dbf

log_archive_dest_1                  string      LOCATION=/oracle/arc
OPTIONAL
                                                REOPEN=300
QUOTA_SIZE=1024MB

log_archive_max_processes           integer     2
```

The first parameter sets the format used to name the archive logs. We used local archiving and one log. The second parameter configures the location and file descriptors for the archive logs. The last parameter is a default parameter, and we left it set to 2.

After making these changes, we shut down the database, then restarted it. We made the same changes to the database on Linux23.

Archived

6.5.2 Installing TSM server

The version of TSM we installed was a *Try-and-Buy* version we obtained from an internal site. The version you would install would be the same with the exception that our licenses were only good for 90 days. We installed TSM Version 5.3.1.

Installing the TSM server was an easy task. It is a matter of installing the correct rpm's in Linux. We used *IBM Tivoli Storage Manager for Linux: Quick Start Version 5.3*, GC23-4692, to assist in the process.

The TSM server was installed by root on Linux22, a guest with one virtual processor and 1 GB of memory. Also installed was Oracle 10g, which is the RMAN catalog database.

We downloaded the code from the IBM internal site. We ftp'd the following as root and placed it in the /opt/TSMtarball directory. We then ran bunzip2 against the file.

```
TIVsm-server-5.3.1-0.s390x.tar.bz2
```

After bun-zipping the file, we needed to run **tar** against the file. After the **tar** command was executed, we had the following in /opt/TSM tarball:

```
linux22:/opt/TSMtarball # ls
.  .. README.SRV noarch TIVsm-server-5.2.3-0.s390x.tar  install_server
s390x
README.Install license
```

Included with the files is an install_server script. Executing this as root will prepare the licenses and install the appropriate rpm's.

```
linux22:/opt/TSMtarball # ./install_server
Preparing License Agreement
Software Licensing Agreement
1. Czech
2. English
3. French
4. German
5. Italian
6. Polish
7. Portuguese
8. Spanish
9. Turkish

Please enter the number that corresponds to the language
you prefer.
2

************************************************************
IMPORTANT: Read the contents of file /README
```

```
            for extensions and corrections to printed
            product documentation.
*************************************************************

Please select a package to install or "Q" to quit.
        1   TIVsm-webadmin-5.2.3-0
        2   TIVsm-webhelp.en_US-5.2.3-0
        3   TIVsm-server-5.2.3-0
        4   TIVsm-tsmscsi-5.2.3-0
        B   BASIC INSTALL
        Q   QUIT
```

The options selected were 1 (Web admin), 2 (Web help), and 3 (TSM server). This installed the package in the default directory. Selecting these options installed three rpm's:

- TIVsm-server-5.2.3-0
- TIVsm-webadmin-5.2.3-0
- TIVsm-webhelp.en_US-5.2.3-0

This can be verified by executing a query using rpm, as follows.

```
linux22:/opt/tivoli/tsm/server/bin # rpm -qa | grep TIV
TIVsm-server-5.2.3-0
TIVsm-webadmin-5.2.3-0
TIVsm-webhelp.en_US-5.2.3-0
linux22:/opt/tivoli/tsm/server/bin #
```

This is important, as the rpm's were installed by the installation script **install_server** but must be removed individually using the **rpm -e** command.

At this point the installation of TSM is complete, and now it needs to be configured. The dsm.opt file was first configured and then the Web Admin functions to compete the configuration. TSM provides a dsmserv.opt.smp file with sample configuration options. We used all the default options. The ones that are important to our project and are most commonly used are shown in Example 6-1.

Example 6-1 TSM server configuration file selections

```
*Communication methods used*
   COMMmethod TCPIP
   COMMmethod HTTP
*
*TCP Port address
   TCPPort 1500

*TCP Port for admin
   TCPADMINPort 1500
```

```
*HTTP port
   HTTPPort 1580
```

We were able to start the TSM server by executing **./dsmserv** in the /opt/tivoli/tsm/server/bin directory. We then pointed our Internet Explorer browser at http://9.12.4.173:1500 to start the Web-enabled functions. The first task was to set up some basic information such as a name for our server, oraTSM.

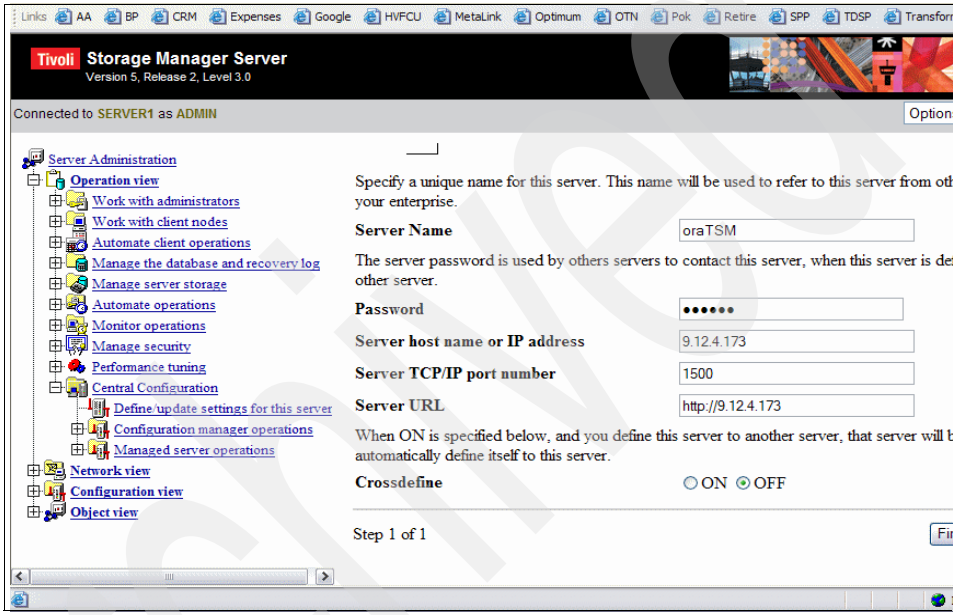


Figure 6-1 Configuring TSM server with Web Admin

After we entered the data we clicked **Finish**. The data was accepted and we then received the below message.

Archived

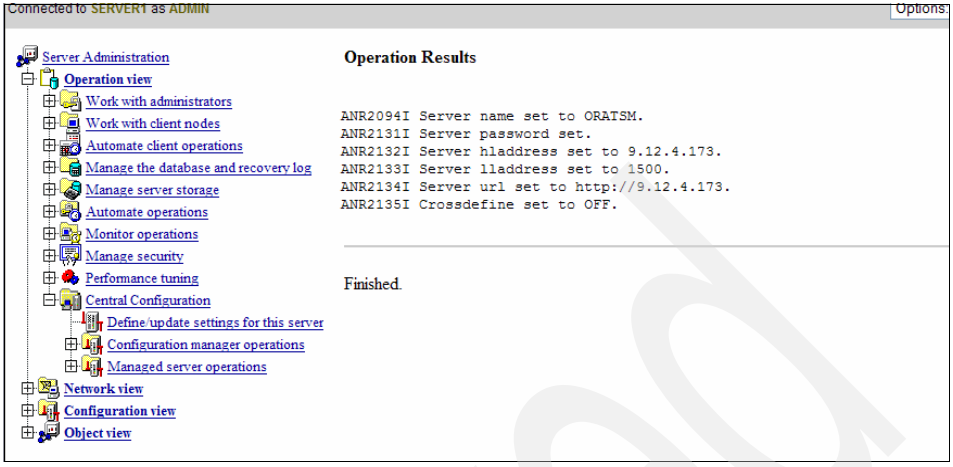


Figure 6-2 Successful results after configuring with Web Admin

By default, the Web administrator is logged off after about 5 minutes of idle time. To avoid having to continually reconnect, we set the time-out length to 480 minutes from the Set Web time out authentication screen.

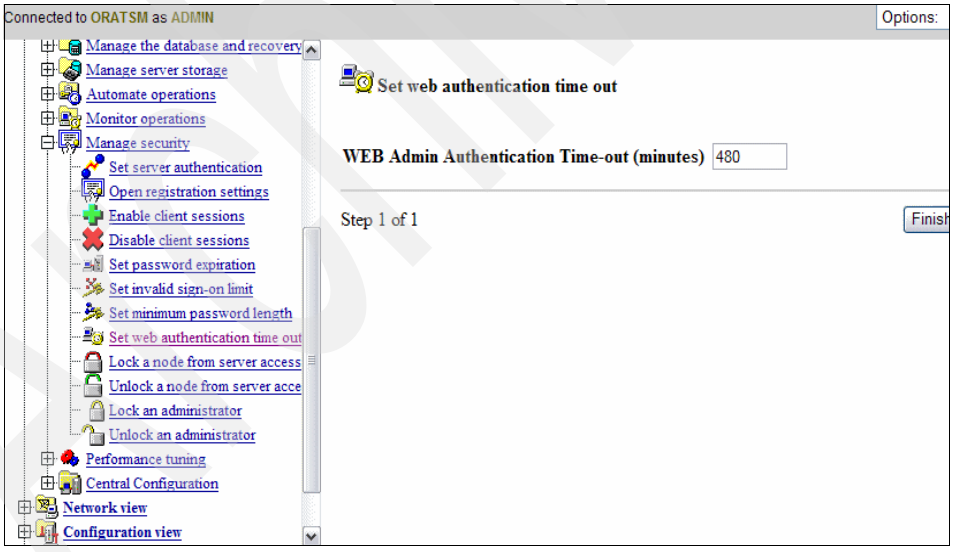


Figure 6-3 Set time out parameter

Selecting the **Finish** button accepted the change and provided a successful indication.

6.5.3 Install Tivoli Data Protect for Oracle

We installed the TDPO software in a Linux23 that also has an Oracle Database 10g installed. The database in this guest is the one that we use to create backup and restore scenarios.

We obtained the Tivoli software from an IBM internal Web site. We ftp'd it as root to the guest.

Next we executed a **bunzip** to the file and then **cpio** to the archive file. Just like TSM server, TDPO is installed with rpm's, but in this case the **rpm** command must be used to install the rpm's. We installed the following rpm's:

```
rpm -i TIVsm-API64.s390x.rpm
rpm -i TDP-Oracle.s390x.rpm
```

This installed Tivoli in the following two directory structures:

- Communications to Oracle in directory /opt/tivoli/tsm/client/oracle/bin64
- Communications to TSM server /opt/tivoli/tsm/client/api/bin64

There is one file that has to be customized for Oracle communications, tdpo.opt. We configured the file as follows.

Example 6-2 tdpo.opt configuration file

```
*****************************************************************************
* IBM Tivoli Storage Manager for Databases
*    Data Protection for Oracle
*
* Sample tdpo.opt for the LinuxZ64 Data Protection for Oracle
*******************************************************************

DSMI_ORC_CONFIG    /opt/tivoli/tsm/client/api/bin64/dsm.opt
DSMI_LOG           /opt/tivoli/tsm/client/oracle/bin64

TDPO_FS            /adsmorc
TDPO_NODE          tdpotest
TDPO_OWNER         oracle
TDPO_PSWDPATH      /opt/tivoli/tsm/client/oracle/bin64


*TDPO_DATE_FMT      1
*TDPO_NUM_FMT       1
*TDPO_TIME_FMT      1

*TDPO_MGMT_CLASS_2   mgmtclass2
*TDPO_MGMT_CLASS_3   mgmtclass3
*TDPO_MGMT_CLASS_4   mgmtclass4
```

We set the TDPO_FS parameter (the file space name on the TSM server) to /adsmorc, which is a default value. This identifies the backup sets on the TSM server.

The TDPO_NODE is a name we selected and will be how the TSM server identifies this client. We need to use the value tdpotest in the server when we add this node to TSM.

There are two files for the TDPO API (communication back to the TSM server) that need to be configured:

- dsm.opt
- dsm.sys

The following is the dsm.opt file and our choice of options.

Example 6-3 dsm.opt file options

```
************************************************************************
* IBM Tivoli Storage Manager                                           *
*                                                                      *
* Sample Client User Options file for UNIX (dsm.opt.smp)               *
************************************************************************

*  This file contains an option you can use to specify the ITSM
*  server to contact if more than one is defined in your client
*  system options file (dsm.sys).  Copy dsm.opt.smp to dsm.opt.
*  If you enter a server name for the option below, remove the
*  leading asterisk (*).

************************************************************************

SErvername       lin22RMAN
```

The SErvername lin22RMAN is the logical name of the TSM server.

The following are the selections we made for the dsm.sys option file.

Example 6-4 dsm.sys configuration file options

```
************************************************************************
* IBM Tivoli Storage Manager                                           *
*                                                                      *
* Sample Client System Options file for UNIX (dsm.sys.smp)             *
************************************************************************

*  This file contains the minimum options required to get started
*  using ITSM.  Copy dsm.sys.smp to dsm.sys.  In the dsm.sys file,
*  enter the appropriate values for each option listed below and
```

Archived

```
*  remove the leading asterisk (*) for each one.

*  If your client node communicates with multiple ITSM servers, be
*  sure to add a stanza, beginning with the SERVERNAME option, for
*  each additional server.

**********************************************************************

SErvername  lin22RMAN
   COMMmethod         TCPip
   TCPPort            1500
   TCPServeraddress   9.12.4.173
```

This file identifies the location of the server we are using. The SErvername in this file must agree with the one in the dsm.opt file. Multiple entries would be needed if we were dealing with multiple servers.

Now that both the TSM server and TDPO clients have the basic configuration complete, the next step is to register this client to the TSM server. This is done from the Web Admin console.

From the Web Admin navigator, we selected **Network View**, then expanded **Client Nodes**, and from the Operations drop-box selected **Register a new node**. We were then taken to the Register New Node page.

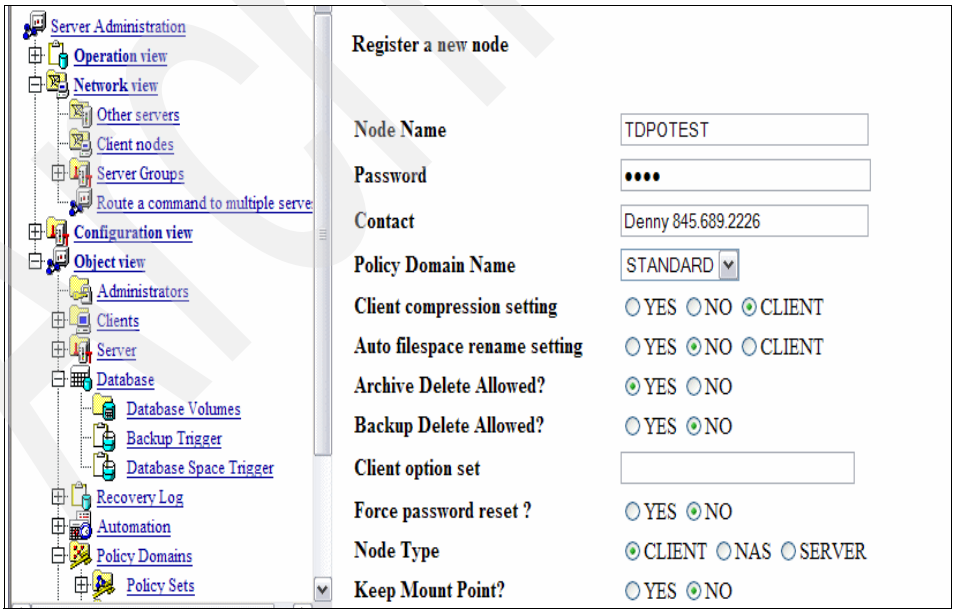


Figure 6-4 Registering the TDPO node

After completing the top three fields, we selected the defaults for the rest of the selections. When this was completed, viewing the Client Nodes pages shows the following (Figure 6-5).

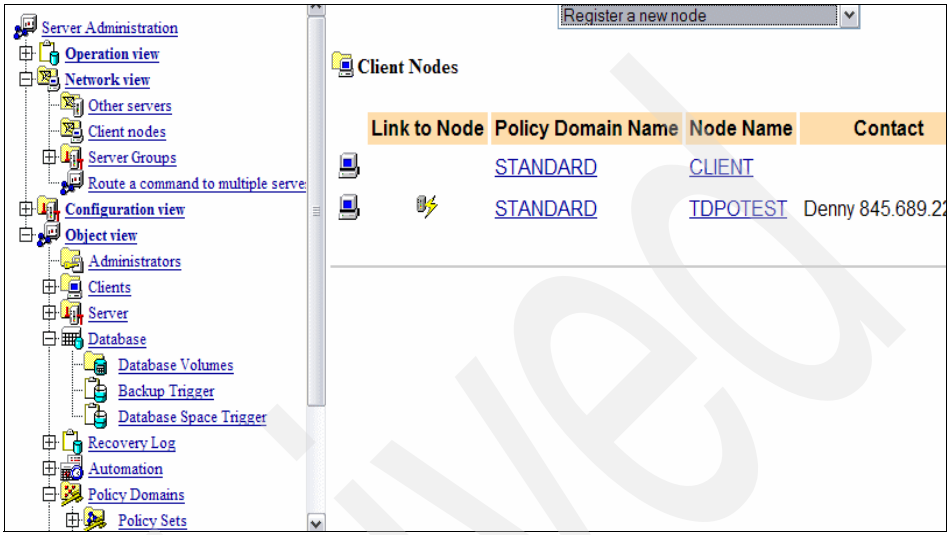


Figure 6-5 Registration for TDPO client completed

Now that the client was registered, were returned to the TDPO client and verified that the client was in fact configured correctly. This could be done either before or after registration. However, one of the functions of the **tdpoconf** utility is to change the password. This will not work from the TDPO client until the node is registered and the first password selected from the Web Admin page, Register new node page. The registration process places a password fileTDPO.tdpotest on the client, You will then have an “old” password to use. We left the password alone, but ran the **tdpoconf showenv** utility to see of the configuration was correct.

Example 6-5 tdpoconf showenv output

```
IBM Tivoli Storage Manager for Databases:
Data Protection for Oracle
Version 5, Release 2, Level 3.0
(C) Copyright IBM Corporation 1997, 2004. All rights reserved.


Data Protection for Oracle Information
 Version:              5
 Release:              2
 Level:                3
 Sublevel:             0
```

```
 Platform:             64bit TDPO LinuxZ64

Tivoli Storage Manager Server Information
 Server Name:          LIN22RMAN
 Server Address:       9.12.4.173
 Server Type:          Linux/s390x
 Server Port:          1500
 Communication Method: TCP/IP

Session Information
 Owner Name:           oracle
 Node Name:            tdpotest
 Node Type:            TDPO LinuxZ64
 DSMI_DIR:             /opt/tivoli/tsm/client/api/bin64
 DSMI_ORC_CONFIG:      /opt/tivoli/tsm/client/api/bin64/dsm.opt
 TDPO_OPTFILE:         /opt/tivoli/tsm/client/oracle/bin64/tdpo.opt
 Password Directory:   /opt/tivoli/tsm/client/oracle/bin64
 Compression:          FALSE
 License Information:  License file exists and contains valid license data.
```

Everything here looks good. One of the important options is the last line of the output. If your license information is not current or valid, you will not be able to connect to the server. If for any reason you are having problems connecting to the server, this is a good place to start to insure all the configuration data is OK.

When TSM is installed there is a default database created to hold backup data from clients. The strategy would be to back up data to this database and then archive to another data set or tape by setting up management classes. Since we have neither tape nor additional disk space, the backups we do will just be put in the databases in the TSM server. And at this point not knowing exactly how much space will be needed and having limited space in the mount point for the TSM server, we added another 2 GBytes of space, adding another database in another mountpoint with additional space available.

The process to accomplish this is from the Admin Navigator. Select **Object View**, then **Server Storage**, then **Disk Storage Pools**, then **Backup Pool**. You should then be at the below page.

Archived

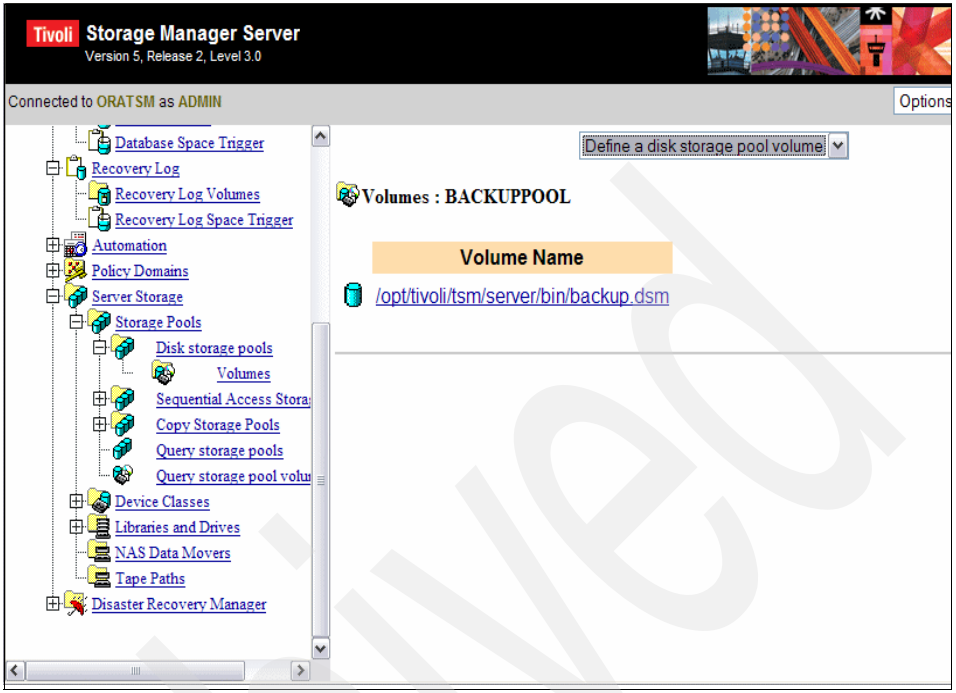


Figure 6-6 Creating additional backup storage pool space - Backup pool

From here, select **Define Disk Storage Pool Volume** from the operations drop-down menu.

This will take you to the following page.

Archived

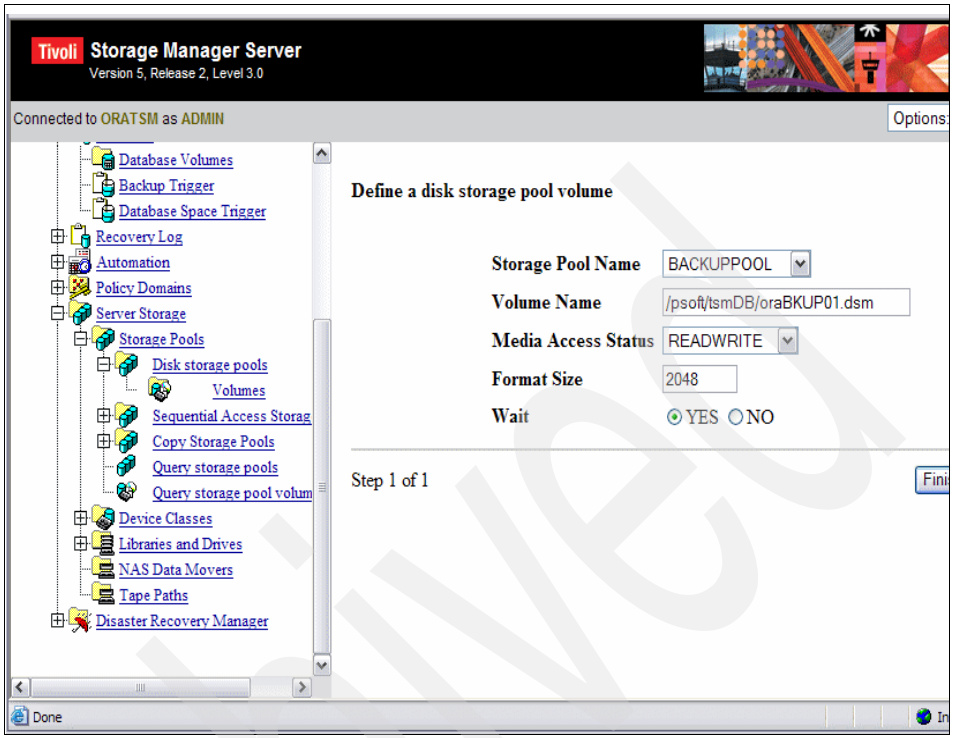


Figure 6-7 Configuring the new backup pool volume

The volume name for the new volume is /psoft/tsmDB/oraBKUP01.dsm. We sized this at 2048 MBytes. Selecting Wait=Yes means we will stay here until the formatting completes and presents either a successful or another message. We completed with no errors. Upon returning to the original page, we see that this volume is now present.

Archived

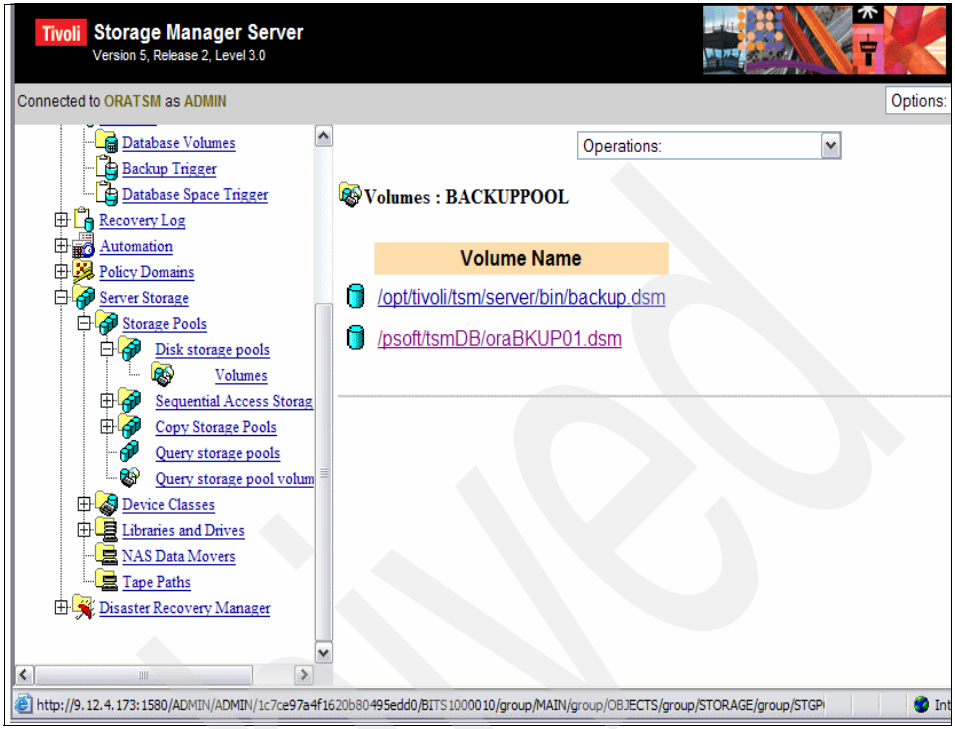


Figure 6-8 Additional backup pool complete

To summarize where we are at this point, we have:

- Configured RMAN and created the Recovery Catalog in Oracle
- Installed and configured the TSM server
- Installed and configured the TDPO client and registered it with the TSM server
- Added additional storage to the backup pool

6.6 Back up the user tablespace

The next step in our process is to create a script for RMAN to use to back up the *users* tablespace. This is a very basic script, as follows.

Example 6-6 Script to use TDPO to back up users tablespace

```
#script to backup tablespace USERS with tdpo
run
        {
```

```
        allocate channel t1 type 'sbt_tape' parms
        'ENV=(TDPO_OPTFILE=/opt/tivoli/tsm/client/oracle/bin64/tdpo.opt)';
backup
        MAXSETSIZE= 10m
        TABLESPACE users;
}
```

The first line of the script, below, is used by RMAN to identify that the TDPO API is to be used, and the line that follows is the location of the tdpo.opt file and sets and environmental to point to this location.

```
allocate channel t1 type 'sbt_tape' parms
```

The remaining lines tell RMAN what to backup and the max size of the backup sets.

We started RMAN.

Example 6-7 Starting RMAN and connecting to the target and catalog databases

```
oracle@linux23:/oracle/product/db10g/bin> ./rman

Recovery Manager: Release 10.1.0.3.0 - 64bit Production

Copyright (c) 1995, 2004, Oracle.  All rights reserved.

RMAN> connect target

connected to target database: TDPOTEST (DBID=3014898925)

RMAN> connect catalog rmgr/rman@rmancat

connected to recovery catalog database

RMAN>
```

To run the script we entered `@/home/oracle/rmScripts/tsBUusers.txt`.

Example 6-8 Execution of script to back up users tablespace

```
RMAN> @/home/oracle/rmScripts/tsBUusers.txt

RMAN> #script to backup tablespace USERS with tdpo
2> run
3>      {
4>      allocate channel t1 type 'sbt_tape' parms
5>      'ENV=(TDPO_OPTFILE=/opt/tivoli/tsm/client/oracle/bin64/tdpo.opt)';
6> backup
7>      MAXSETSIZE= 10m
```

```
8>         TABLESPACE users;
9> }
allocated channel: t1
channel t1: sid=147 devtype=SBT_TAPE
channel t1: Data Protection for Oracle: version 5.2.3.0

Starting backup at 19-MAY-05
channel t1: starting full datafile backupset
channel t1: specifying datafile(s) in backupset
input datafile fno=00004 name=/oracle/oradata/tdpotest/users01.dbf
channel t1: starting piece 1 at 19-MAY-05
channel t1: finished piece 1 at 19-MAY-05
piece handle=09gkquh5_1_1 comment=API Version 2.0,MMS Version 5.2.3.0
channel t1: backup set complete, elapsed time: 00:00:08
Finished backup at 19-MAY-05
released channel: t1

RMAN> **end-of-file**
```

Once the backup completed successfully, as seen above, we did a list of the Oracle Recovery Catalog to insure the backup was in fact successful and to determine its information. We did this by executing a **`list backup`** command in RMAN.

Example 6-9 List of backup sets in recovery catalog

```
RMAN> list backup
2> ;


List of Backup Sets
===================

BS Key  Type LV Size       Device Type Elapsed Time Completion Time
------- ---- -- ---------- ----------- ------------ ---------------
1713    Full    1M         SBT_TAPE    00:00:07     19-MAY-05
        BP Key: 1714   Status: AVAILABLE  Compressed: NO  Tag:
TAG20050519T173429
        Handle: 09gkquh5_1_1   Media:
  List of Datafiles in backup set 1713
  File LV Type Ckp SCN    Ckp Time  Name
  ---- -- ---- ---------- --------- ----
  4       Full 3109561    19-MAY-05 /oracle/oradata/tdpotest/users01.dbf
```

Archived

Now we can also look at the backup pool in TSM and see if we can find the same information. There are two ways (at least) to look for this information. The first way is to use a simple **select** command to query the storage pools. We used:

```
select * from contents where node_name='TDPOTEST'
```

On the command line of the Web Administrator, we used the command shown in Figure 6-9.

Figure 6-9 Query the storage pool for results

The results of the query are then displayed.

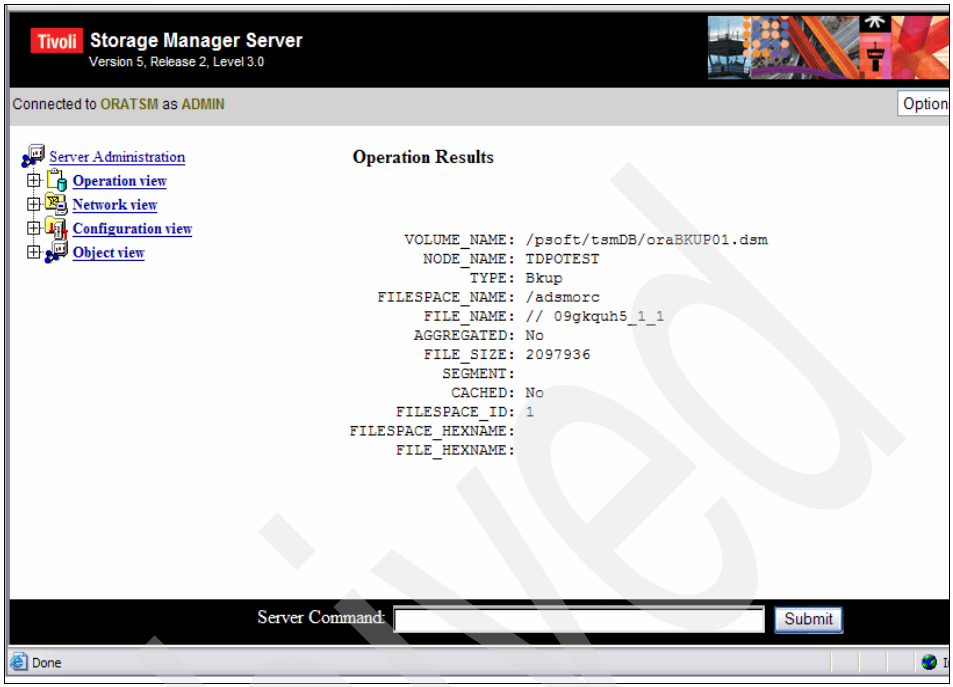


Figure 6-10 Results of query of storage pool

Upon examination of the figure, we can see the following;

```
FILE_NAME: // 09gkquh5_1_1
```

If we go back to the list backup command we did to the recovery catalog in Oracle we will see that the handle is the same:

```
Handle: 09gkquh5_1_1
```

This indicates that the object in the TSM backup storage pool is the same object that RMAN recorded in backing up the users tablespace.

One other way to see the number of files that are in the backup pool is to query the backup pool from the File Name Space. To get to this option select **Object view**, then **Clients**, then **Client Node**, then **File Space Name**, and then select **Query File Space Occupancy** from the Operations drop-down menu.

Archived

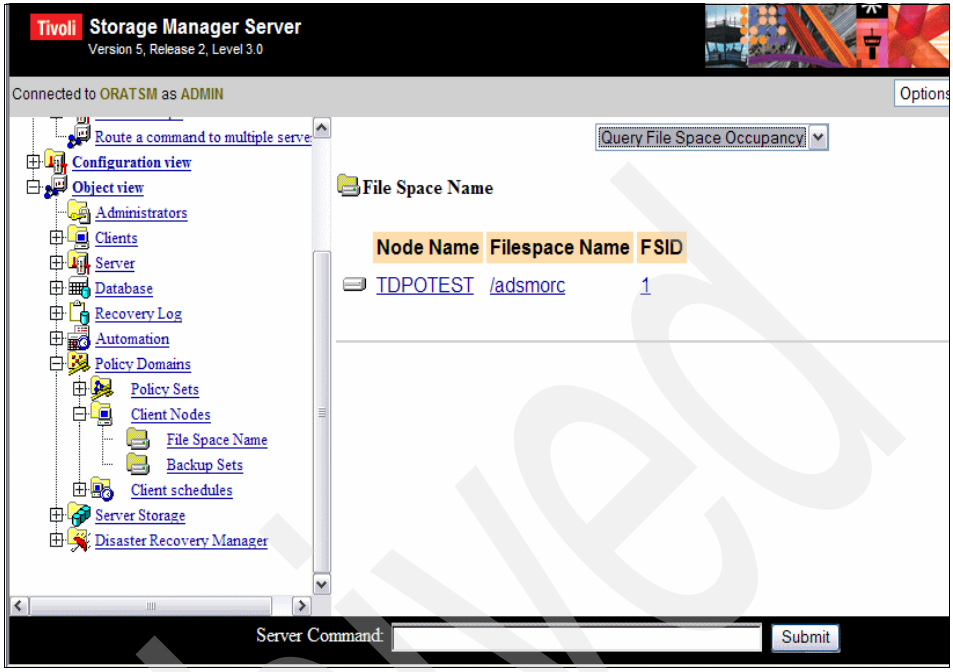


Figure 6-11 File space name to query the file space occupancy

From here you select the **Query File Space Occupancy** from the Operations drop-down menu and you will be presented with the following page.

Archived

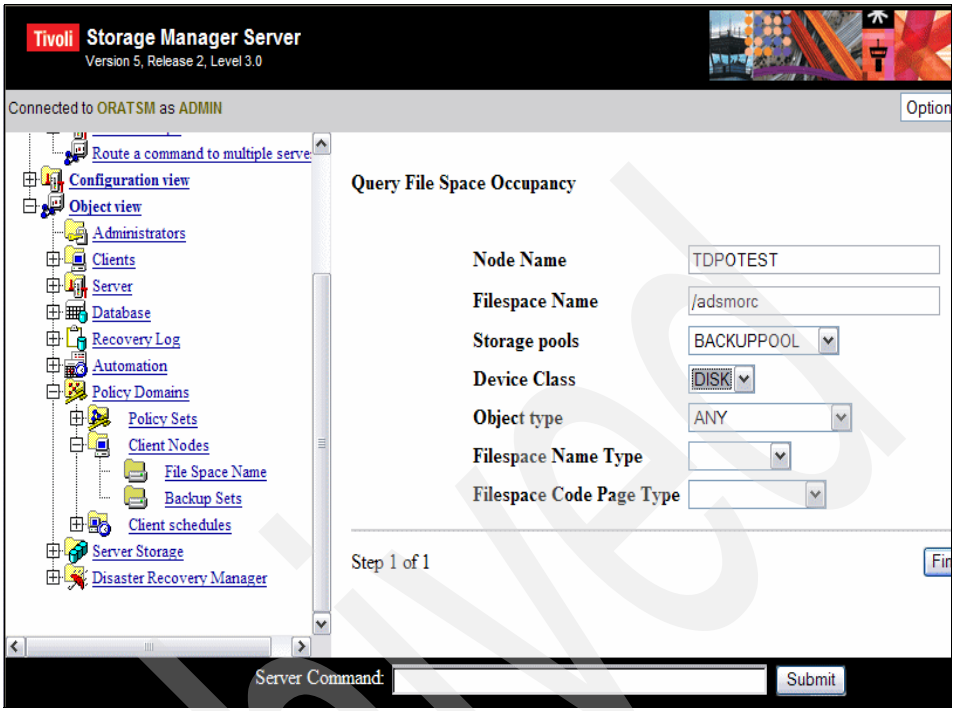


Figure 6-12 Query file space occupancy

Fill in the fields as appropriate. Our node is TDPOTEST. The filespace name is /adsmorc (remember we put this in the TDPO dsm.opt config file). The Backup Storage Pool and the Device Type are DISK. Select **Finish** and you will receive the results of the query.

Archived

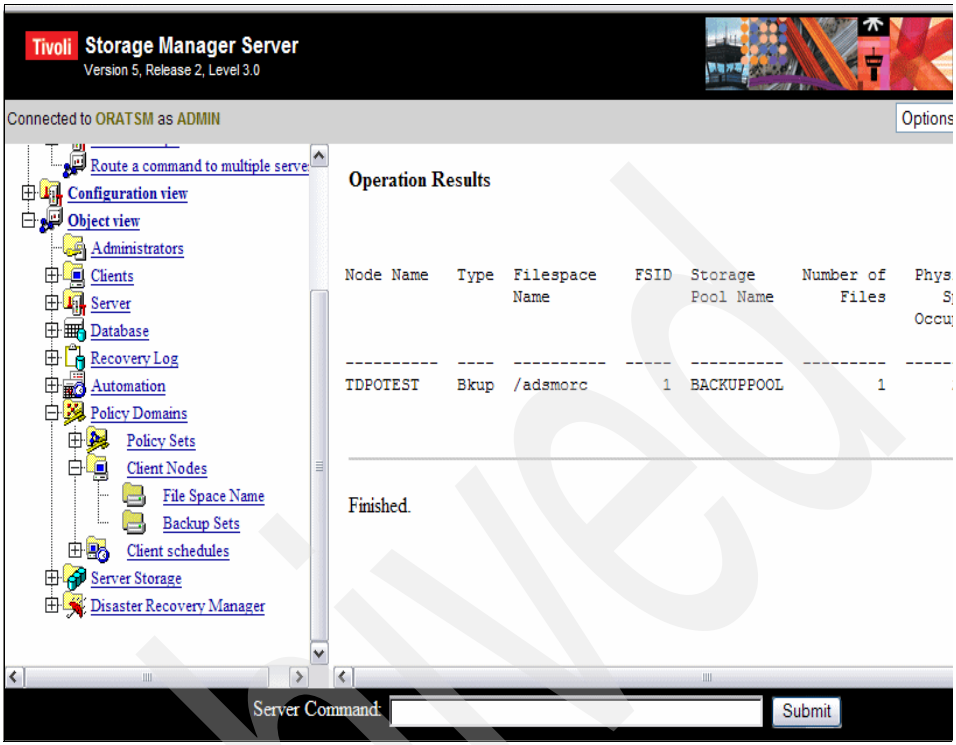


Figure 6-13 Query results

We did one backup, and one file is resident in the backup pool.

6.7 Restore and recover the users Tablespace

Once a backup of the user tablespace was successfully completed and verified, we chose to restore (essentially replace the datafile for the tablespace) and then do a recovery. We did a recovery because, as you will see from the process, we made changes to the tablespace so that it was not consistent with the original backup. This required Oracle to roll forward through the archive logs.

The following steps were executed:

1. Create a new user dutch/dutch that will own the table we will create.
2. Create the table TDPO_TEST.
3. Add three rows to the table we created.
4. Shut down the database.
5. Rename the users01.dbf data file so that the tablespace was *corrupted*.
6. Start the database and verified there was no problem.

7. Shut down the database and restart in mount mode.
8. Run a RMAN script to restore the tablespace.
9. Execute the **recover** command from rman.
10. Shut down and then start up the database.
11. Verify the recovery was successful by querying the table.

6.7.1 Restore and recover process

After we created the user ID dutch, we created two simple scripts, as follows, to create the table TDPO_TEST and add three rows to the table.

Example 6-10 SQL script to create TDPO_TEST

```
-- script to build a table
-- use storage defaults
-- script will be executed by user dutch in Oracle

drop table tdpo_test;

create table tdpo_test
(
        DBNAME          varchar2(12),
        FUNCTION        varchar2(24),
        SYSTEM_ID       varchar2(12)
);
```

Example 6-11 SQL script to add rows to TDPO_TEST

```
-- add rows into table we just created for the tdpo test

INSERT INTO tdpo_test
        VALUES('RMANCAT', 'CATALOG', 'LINUX22');

INSERT INTO tdpo_test
        VALUES('TDPO_TEST','TEST DATA BASE', 'LINUX23');

INSERT INTO tdpo_test
        VALUES('ORCL','9I TEST','LINUX1');
```

We verified that the table was created and rows were inserted there by executing a query against the table. We logged into the database as dutch and executed a **select *** against the table. From the results below, you can see the table and three rows are there.

Example 6-12 Result of select statement

```
SQL> select * from tdpo_test;
```


Archived

```
DBNAME       FUNCTION                 SYSTEM_ID
------------ ------------------------ ------------
RMANCAT      CATALOG                  LINUX22
TDPO_TEST    TEST DATA BASE           LINUX23
ORCL         9I TEST                  LINUX1
```

Next we logged off as dutch and logged in as sysdba to shut down the database. Once the database was down, the database file users01.dbf was renamed (using a **mv** command) to users01.dbf.old.

Example 6-13 Renaming database file for users tablespace

```
oracle@linux23:/oracle/product/db10g> cd ../../oradata/tdpotest
oracle@linux23:/oracle/oradata/tdpotest> ls
control01.ctl  example01.dbf  redo03.log    temp01.dbf     users01.dbf.old
control02.ctl  redo01.log     sysaux01.dbf  undotbs01.dbf
control03.ctl  redo02.log     system01.dbf
oracle@linux23:/oracle/oradata/tdpotest>
```

With the users01.dbf file missing, the following error was detected on startup:

```
SQL> startup
ORACLE instance started.

Total System Global Area  595591168 bytes
Fixed Size                  1323152 bytes
Variable Size             170381168 bytes
Database Buffers          423624704 bytes
Redo Buffers                 262144 bytes
Database mounted.
ORA-01157: cannot identify/lock data file 4 - see DBWR trace file
ORA-01110: data file 4: '/oracle/oradata/tdpotest/users01.dbf'
```

The database was shutdown and then brought up in mount by usng the following command while logged in as sysdba:

```
startup mount
```

When the database came up as mount, sys was logged off and rman was started and connected to the target, and catalog databases were completed.

```
RMAN> connect target

connected to target database: TDPOTEST (DBID=3014898925)

RMAN> connect catalog rmgr/rman@rmancat

connected to recovery catalog database
```

The script we created to restore the database was executed.

Example 6-14 RMAN script to RESTORE the users tablespace

```
#script to recover tablespace USERS with tdpo
run
        {
        allocate channel t1 type 'sbt_tape' parms
        'ENV=(TDPO_OPTFILE=/opt/tivoli/tsm/client/oracle/bin64/tdpo.opt)';
restore
        TABLESPACE users;
}
```

Executing the script gave the following results:

```
RMAN> @/home/oracle/rmScripts/tsRESTusers.txt

RMAN> #script to recover tablespace USERS with tdpo
2> run
3>      {
4>      allocate channel t1 type 'sbt_tape' parms
5>      'ENV=(TDPO_OPTFILE=/opt/tivoli/tsm/client/oracle/bin64/tdpo.opt)';
6> restore
7>         TABLESPACE users;
8> }
allocated channel: t1
channel t1: sid=160 devtype=SBT_TAPE
channel t1: Data Protection for Oracle: version 5.2.3.0

Starting restore at 31-MAY-05

channel t1: starting datafile backupset restore
channel t1: specifying datafile(s) to restore from backup set
restoring datafile 00004 to /oracle/oradata/tdpotest/users01.dbf
channel t1: restored backup piece 1
piece handle=09gkquh5_1_1 tag=TAG20050519T173429
channel t1: restore complete
Finished restore at 31-MAY-05
released channel: t1

RMAN> **end-of-file**
```

Of interest, other than the successful completion, note that the handle used to identify the backup piece is `piece handle=09gkquh5_1_1`, which is the handle we identified as the backed up tablespace when we did the backup. That is how TSM identifies through RMAN and TDPO what file to use to restore the requested object.

Archived

Looking at the directory containing the database files we now see that the users01.dbf file has been restored:

```
oracle@linux23:/oracle/product/db10g> cd ../../oradata/tdpotest
oracle@linux23:/oracle/oradata/tdpotest> ls
control01.ctl  example01.dbf  redo03.log    temp01.dbf     users01.dbf.old
control02.ctl  redo01.log     sysaux01.dbf  undotbs01.dbf
control03.ctl  redo02.log     system01.dbf  users01.dbf
oracle@linux23:/oracle/oradata/tdpotest>
```

At this point, the database was shutdown and then started again to an open status (open is the default parameter to startup). However, errors were still present:

```
SQL> startup
ORACLE instance started.

Total System Global Area  595591168 bytes
Fixed Size                  1323152 bytes
Variable Size             170381168 bytes
Database Buffers          423624704 bytes
Redo Buffers                 262144 bytes
Database mounted.
ORA-01113: file 4 needs media recovery
ORA-01110: data file 4: '/oracle/oradata/tdpotest/users01.dbf'
```

This message is a bit different from the previous time we tried to start up the database. This time the message indicates that media recovery is needed. The media recovery is accomplished using the **recover** command in **rman**. This process will search through the archive files and roll the database forward so all the transactions committed in the archive logs since the backup of the tablespace was taken will be in the database and the database is in a consistent state.

The command **recover tablespace users;** was entered. Oracle started reading the existing archive logs to get the scn's (system change numbers), and therefore the database changes needed to recover the database to the point in time the database file users01.dbf was renamed (effectively corrupted). The output at the console was a follows:

```
RMAN>
Starting recover at 01-JUN-05
allocated channel: ORA_DISK_1
channel ORA_DISK_1: sid=159 devtype=DISK

starting media recovery

archive log thread 1 sequence 959 is already on disk as file
/oracle/arc/1_959_5
```

```
51812206.dbf
archive log thread 1 sequence 987 is already on disk as file
/oracle/arc/1_987_551812206.dbf
archive log thread 1 sequence 988 is already on disk as file
/oracle/arc/1_988_551812206.dbf
archive log thread 1 sequence 989 is already on disk as file
/oracle/arc/1_989_551812206.dbf
-- Lines deleted
-- Lines deleted
archive log filename=/oracle/arc/1_1044_551812206.dbf thread=1
sequence=1044
archive log filename=/oracle/arc/1_1045_551812206.dbf thread=1
sequence=1045
archive log filename=/oracle/arc/1_1046_551812206.dbf thread=1
sequence=1046
archive log filename=/oracle/arc/1_1047_551812206.dbf thread=1
sequence=1047
media recovery complete
Finished recover at 01-JUN-05
starting full resync of recovery catalog
full resync complete

> RMAN
```

The channel opened to read the archive logs is DISK. We did not back up the archive logs to TSM. All the archive files were present in /oracle/arc. If we had backed up the archive files to TSM we would have had to connect to the target database, the catalog database, and use the **sbt_tape** api to retrieve the archive files.

After the recovery complete message from rman, the database was shutdown and a **startup** was issued with the following results:

```
SQL> startup
ORACLE instance started.

Total System Global Area  595591168 bytes
Fixed Size                  1323152 bytes
Variable Size             170381168 bytes
Database Buffers          423624704 bytes
Redo Buffers                 262144 bytes
Database mounted.
Database opened.
SQL> quit
```

After completing the media recovery, we logged into the database with user dutch and queried the table we created to insure it was there as well as the data we inserted, indicating a good recovery.

```
oracle@linux23:/oracle/oradata/tdpotest> sqlplus dutch/dutch
SQL*Plus: Release 10.1.0.3.0 - Production on Wed Jun 1 11:01:09 2005
Copyright (c) 1982, 2004, Oracle.  All rights reserved.

Connected to:
Oracle Database 10g Enterprise Edition Release 10.1.0.3.0 - 64bit
Production
With the Partitioning, OLAP and Data Mining options

SQL> select * from tdpo_test;

DBNAME       FUNCTION                 SYSTEM_ID
------------ ------------------------ ------------
RMANCAT      CATALOG                  LINUX22
TDPO_TEST    TEST DATA BASE           LINUX23
ORCL         9I TEST                  LINUX1

SQL>
```

6.8 Summary

This was a basic experience in using the TDPO client to back up a tablespace. Both RMAN and TSM/TDPO contain much more function than shown here. The Recovery Manager function along with the Tivoli applications provide the tools necessary to develop and maintain a backup strategy for Oracle databases. All databases with critical data should have a backup strategy in place. While Tivoli is one product to help manage the backup sets created by Recovery Manager, there are other products available. We chose to use Tivoli, as it was an IBM product and easy for us to obtain, and gave us the chance to work with a beta copy of TDPO.

Archived

7

Using Cobol and C/C++ with Oracle Database 10g

This chapter contains examples of using COBOL and C/C++ with an Oracle 10g Database on Linux on zSeries. In this chapter we used the COBOL compiler from ACUCORP and the COBOL compiler from MicroFocus.

Archived

© Copyright IBM Corp. 2005. All rights reserved.

7.1 Working with Pro*Cobol and sample programs

Pro*COBOL is a programming tool that enables you to embed SQL statements in a COBOL program. The Pro*COBOL precompiler converts the SQL statements in the COBOL program into standard Oracle run-time library calls. The generated output file can then be compiled, linked, and run in the usual manner.

The objective of this section is to perform the setup necessary to run the Pro*COBOL precompiler programs provided with Oracle 10g Database.

7.1.1 Install the Pro*COBOL precompiler

You need to install the Pro*COBOL precompiler, as it is not installed with the database installation. You can install it from the client CD.

1. Start a VNC client session for the oracle user.
2. Extract the files from the client CD.

   ```
   # cpio -idmc  <  ship_lnx390_client.cpio
   ```

   This will extract contents of the client CD and place it in a directory called Disk1.
3. Start the Oracle Universal Install (OUI).

   ```
   # cd  Disk1
   # ./runInstaller &
   ```
4. At the Welcome window, click **Next** to continue.
5. At the Specify File Locations window, click **Next** to accept the defaults.
6. At the Select Installation Type window, select **Administrator** and click **Next**.

Archived

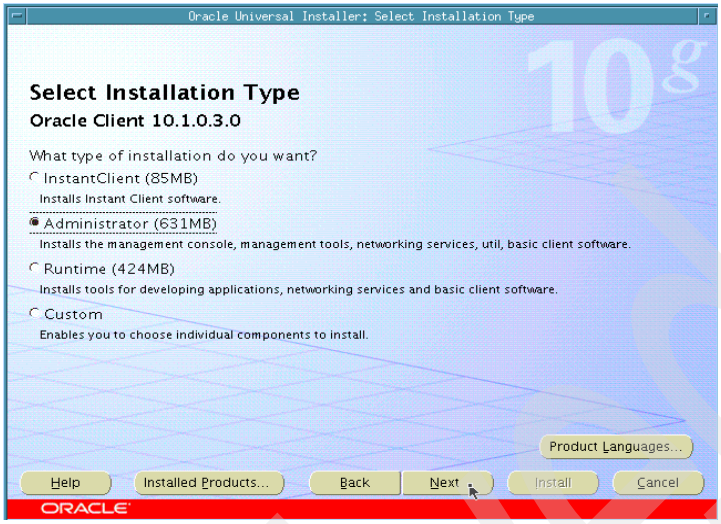


Figure 7-1 Select Installation Type panel

7. At the Summary window, click **Install** to start the installation. Note that Pro*COBOL is listed as one of the new installations.

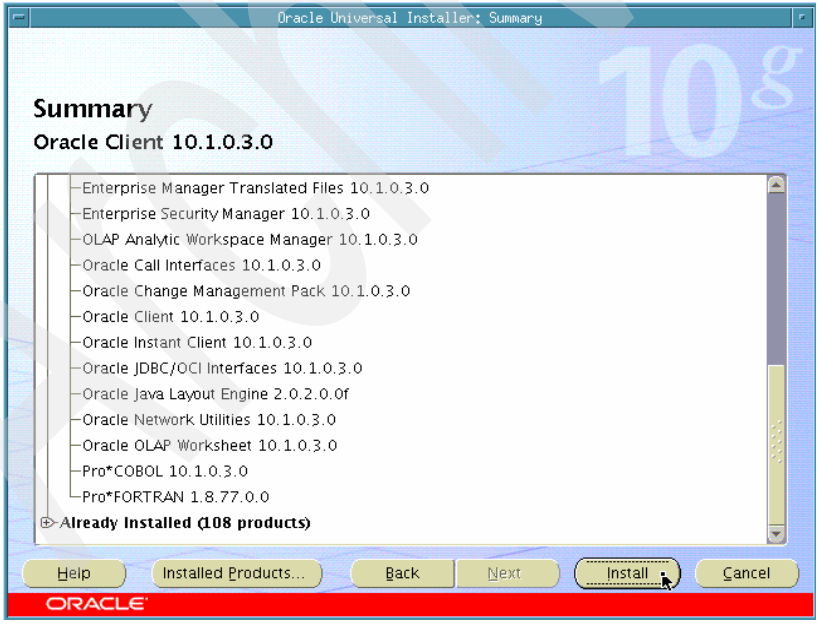


Figure 7-2 Summary panel

8. The installation will take a few minutes, depending on the system load. When the installation is complete, click **Exit**.
9. Verify that the Pro*COBOL precompiler has been installed successfully.

```
# procob  | more
```

You should see output similar to the following:

```
Pro*COBOL: Release 10.1.0.3.0 - Production on Tue Aug 10 19:15:36 2004
Copyright (c) 1982, 2004, Oracle.  All rights reserved.
System default option values taken from:
/oracle/o10g/precomp/admin/pcbcfg.cfg
Option Name    Current Value  Description
….
```

7.1.2 Sample Pro*COBOL programs

There are fifteen sample Pro*COBOL precompiler programs provided that demonstrate the use of embedded SQL to access an Oracle 10g Database. The samples are installed in the $ORACLE_HOME/precomp/demo/procob2 directory from the Oracle Companion Products CD. The samples are sample1-14.pco and lobdemo1.pco.

```
Sample Program 1:  Simple Query
Sample Program 2:  Cursor Operations
Sample Program 3:  Host Tables
Sample Program 4:  Datatype Equivalencing
Sample Program 5:  SQL*Forms User Exit
Sample Program 6:  Dynamic SQL Method 1
Sample Program 7:  Dynamic SQL Method 2
Sample Program 8:  Dynamic SQL Method 3
Sample Program 9: Calling a Stored Procedure
Sample Program 10: Dynamic SQL Method 4
Sample Program 11:  Cursor Variable Operations
Sample Program 12: Dynamic SQL Method 4 using ANSI Dynamic SQL
Sample Program 13:  Nested Program
Sample Program 14:  Tables of group items
LOB Demo 1: DMV Database
```

Sample program 6, sample6.pco, can be precompiled by changing to the directory containing the sample and running the precompiler. For example:

```
cd $ORACLE_HOME/precompl/demo/procob2
procob sample6.pco
```

The precompiler will create two files: sample6.lis and sample6.cob. The .lis file is the precompiler listing file and the .cob is a COBOL source file ready for input to the COBOL compiler.

The embedded SQL details can be studied by looking directly at the Pro*COBOL precompiler source code. Additional details can be found in the *Oracle Pro*COBOL Programmer's Guide Release 9.2,* A96109-03.

7.2 Using ACUCOBOL-GT Version 6.1

We used ACUCOBOL-GT Version 6.1 from Acucorp, Inc. Follow these steps to install the product:

1. Log in to the root user.
2. Get the contents from the CD.
3. Extract the contents of the installation file.

   ```
   # tar   xfz   acucobol.tgz
   ```

   You should now have a /opt/acucobol directory with the following contents:

   ```
   # ls   /opt/acucobol
   .  ..  READ_ME  RELEASE  bin  etc  lib  sample  ship.sums  tools
   ```

4. Activate the license file for the product you have just installed by executing the activator script.

   ```
   # /opt/acucobol/bin/activator
      Enter the product code []: <Located on last page of this document >
      Enter the product key []:   <Located on the last page of this document >
      Creating  /opt/acucobol/bin/runcbl.alc... Done...
      Creating  /opt/acucobol/bin/ccbl.alc... Done...
   ```

5. Set up environment variables.

   ```
   # echo  '# Settings for ACUCOBOL'  >>/etc/profile.local
   # echo  'export PATH=/opt/acucobol/bin:$PATH'   >>/etc/profile.local
   # echo  'export A_TERMCAP=/etc/termcap'   >>/etc/profile.local
   #  .   /etc/profile.local
   ```

6. Test the installation by compiling and running a sample program.

   ```
   # cd  /opt/acucobol/sample
   # ccbl  tour.cbl
   # runcbl  tour.acu
   ```

7.2.1 Relinking ACUCOBOL-GT with Oracle

As you can see in the above sample program, the **ccbl** command is used to compile a COBOL program. The compiler generates an executable file with the '.acu' suffix. The **runcbl** command is then used to run the '.acu' executable file.

Before you can run any sample programs provided by Oracle, you must generate a version of the **runcbl** command that can execute oracle database operations.

There is a white paper, using ACUCOBOL-GT with the Oracle Pro*COBOL Precompiler, and a shell script, AcuPro.sh, available from your ACUCORP sales representative that describe and ease the implementation of the following tasks.

Continue with the root user and perform the following:

1. Back up the files we need to modify.

```
# cd   /opt/acucobol/bin
# cp   runcbl  runcbl.orig
# cd    ../lib
# cp   Makefile  Makefile.orig
# cp   direct.c  direct.c.orig
```

2. Edit the makefile and direct.c files to include the necessary Oracle libraries and function definitions.

3. Find the SQLLIB routines that will be called by the precompiler source code by issuing the commands:

```
cd $ORACLE_HOME/precomp/lib
nm cobsqlintf.o |grep SQL
```

   The **nm** command list symbols from object files.

4. Declare the routines obtained from the **nm** command as external to the ACUCOBOL runtime. An example is:

```
extern void sqladr()
```

5. Add the names of the routines to the DIRECTTABLE structure in direct.c. An example is:

```
{"SQLADR", FUNC sqladr, C_void}
```

   Putting steps 4 and 5 together, we would have the following in direct.c to be able to call the functions "sqladr":

```
extern void sqladr():
struct DIRECTTABLE LIBDIRECT[]
{"sqladr", FUNC sqladr, C_void},
{NULL, NULL, 0} };
```

6. Verify by comparing the original file and the new file with the **diff** command. Lines starting with < denote deleted lines. Lines starting with > denote added lines.

```
# diff  direct.c.orig   direct.c
81,83c81
< struct        DIRECTTABLE LIBDIRECT[] = {
<       { NULL,                 NULL,           0 }
<       };
---
> /*  Start of Pro*COBOL declarations */
84a83,148
```

```
> #define VOID void
> #include "$ORACLE_HOME/precomp/public/sqlda.h"
> extern VOID sqlbex();
> extern VOID sqlnul();
> extern VOID sqlab1();
> extern VOID sqladr();
> extern VOID sqladrvc();
> extern VOID sqlad1();
> extern SQLDA *sqlald();
> extern VOID sqlbs1();
> extern VOID sqlcls();
> extern VOID sqlcom();
> extern VOID sqlfcc();
> extern VOID sqlfch();
> extern VOID sqlgb1();
> extern VOID sqlgd1();
> extern VOID sqllo1();
> extern VOID sqllda();
> extern unsigned int sqllen();
> extern VOID sqloca();
> extern VOID sqlopn();
> extern VOID sqlora();
> extern VOID sqlos1();
> extern VOID sqlosq();
> extern VOID sqlpcs();
> extern VOID sqlrol();
> extern VOID sqlsca();
> extern VOID sqlsqs();
> extern VOID sqltfl();
> extern VOID sqltoc();
> extern VOID sqlgri();
>
> struct  DIRECTTABLE LIBDIRECT[] = {
>           { "SQLBEX",FUNC sqlbex,C_void },
>           { "SQLNUL",FUNC sqlnul,C_void },
>           { "SQLAB1",FUNC sqlab1,C_void },
>           { "SQLADR",FUNC sqladr,C_void },
>           { "SQLADRVC",FUNC sqladrvc,C_void },
>           { "SQLAD1",FUNC sqlad1,C_void },
>           { "SQLALD",FUNC sqlald,C_pointer },
>           { "SQLBS1",FUNC sqlbs1,C_void },
>           { "SQLCLS",FUNC sqlcls,C_void },
>           { "SQLCOM",FUNC sqlcom,C_void },
>           { "SQLFCC",FUNC sqlfcc,C_void },
>           { "SQLFCH",FUNC sqlfch,C_void },
>           { "SQLGB1",FUNC sqlgb1,C_void },
>           { "SQLGD1",FUNC sqlgd1,C_void },
>           { "SQLLO1",FUNC sqllo1,C_void },
>           { "SQLLDA",FUNC sqllda,C_void },
```

Archived

```
>           { "SQLLEN",FUNC sqllen,C_unsigned },
>           { "SQLOCA",FUNC sqloca,C_void },
>           { "SQLOPN",FUNC sqlopn,C_void },
>           { "SQLORA",FUNC sqlora,C_void },
>           { "SQLOS1",FUNC sqlos1,C_void },
>           { "SQLOSQ",FUNC sqlosq,C_void },
>           { "SQLPCS",FUNC sqlpcs,C_void },
>           { "SQLROL",FUNC sqlrol,C_void },
>           { "SQLSCA",FUNC sqlsca,C_void },
>           { "SQLSQS",FUNC sqlsqs,C_void },
>           { "SQLTFL",FUNC sqltfl,C_void },
>           { "SQLTOC",FUNC sqltoc,C_void },
>           { "SQLGRI",FUNC sqlgri,C_void },
>           { NULL,      NULL,          0 }
>   };
> /*  End of Pro*COBOL declarations */
```

7. Add the Oracle libraries to the FSI_LIBS line in the makefile.

```
FSI_LIBS=$(ACU_LIBDIR)/libexpat.a -L$($ORACLE_HOME)/lib -lclntsh
‘cat $(ORACLE_HOME/lib/sysliblist’
```

In our install $(ACU_LIBDIR)=/opt/acucobol/lib:

```
# diff  Makefile.orig  Makefile
124,125c124,125
< FSI_LIBS = $(ACU_LIBDIR)/libexpat.a
<
---
> FSI_LIBS = $(ACU_LIBDIR)/libexpat.a \
>           -L$(ORACLE_HOME)/lib -lclntsh `cat
$(ORACLE_HOME)/lib/sysliblist`
```

8. Generate a new **runcbl** command.

First make sure you have an Oracle environment. Execute the profile you have set up for the oracle user during the Oracle database installation.

```
# .   /home/oracle/.profile
(don't forget the dot (.) and  space before the file name)
```

Execute the Makefile to relink the **runcbl** command.

```
#  make  runcbl
cc -D_LARGEFILE64_SOURCE -s -o runcbl amain.o sub.o filetbl.o axml.o
./libruncbl.a \
./libclnt.a ./libacvt.a ./libfsi.a ./libacuterm.a ./libvision.a
./libexpat.a -L/oracle/o10g/lib -lclntsh `cat /oracle/o10g/lib/sysliblist`
./libsocks.a ./libmessage.a ./libcfg.a ./liblib.a ./libstdlib.a
./libmemory.a  -ldl  ./libz.a -lm
```

Install the newly generated **runcbl** command.

```
#  cp runcbl  ../bin/runcbl
```

Make sure the **runcbl** command still works with the ACUCOBOL sample program.

```
# runcbl /opt/acucobol/sample/tour.acu
```

Attention: The licenses runcbl.alc and ccbl.alc must be in the same directory as runcbl and ccbl, respectively. A simple test is to execute the command with the -v option to see the license information.

7.2.2 Work with the Oracle Pro*COBOL samples

In this section we work with the samples provided by Oracle.

Sample tables

Most of the sample programs use two database tables: DEPT and EMP, which have been created for you during the Oracle database installation. Their definitions follow:

```
CREATE TABLE DEPT
    (DEPTNO    NUMBER(2),
     DNAME     VARCHAR2(14),
     LOC       VARCHAR2(13))

CREATE TABLE EMP
    (EMPNO     NUMBER(4) primary key,
     ENAME     VARCHAR2(10),
     JOB       VARCHAR2(9),
     MGR       NUMBER(4),
     HIREDATE  DATE,
     SAL       NUMBER(7,2),
     COMM      NUMBER(7,2),
     DEPTNO    NUMBER(2))
```

Sample data

Respectively, the DEPT and EMP tables contain the following rows of data:

```
DEPTNO  DNAME      LOC
------- ---------- ---------
10      ACCOUNTING NEW YORK
20      RESEARCH   DALLAS
30      SALES      CHICAGO
40      OPERATIONS BOSTON

EMPNO ENAME   JOB          MGR  HIREDATE    SAL   COMM  DEPTNO
----- ------- --------- ------ --------- ------ ------ -------
 7369 SMITH   CLERK       7902 17-DEC-80    800             20
 7499 ALLEN   SALESMAN    7698 20-FEB-81   1600    300      30
```

Archived

```
7521 WARD    SALESMAN    7698 22-FEB-81   1250    500      30
7566 JONES   MANAGER     7839 02-APR-81   2975             20
7654 MARTIN  SALESMAN    7698 28-SEP-81   1250   1400      30
7698 BLAKE   MANAGER     7839 01-MAY-81   2850             30
7782 CLARK   MANAGER     7839 09-JUN-81   2450             10
7788 SCOTT   ANALYST     7566 19-APR-87   3000             20
7839 KING    PRESIDENT        17-NOV-81   5000             10
7844 TURNER  SALESMAN    7698 08-SEP-81   1500             30
7876 ADAMS   CLERK       7788 23-MAY-87   1100             20
7900 JAMES   CLERK       7698 03-DEC-81    950             30
7902 FORD    ANALYST     7566 03-DEC-81   3000             20
7934 MILLER  CLERK       7782 23-JAN-82   1300             10
```

7.2.3 Prepare and run the sample programs

To prepare and run the sample programs:

1. Log in to the oracle user.
2. Unlock the SCOTT user account if you have not done so already in a previous exercise.

   Note that the user SCOTT with password TIGER is hard coded into the sample programs.

   ```
   # sqlplus  /  'as sysdba'
   SQL> alter user scott identified by tiger account unlock;
   SQL> quit
   ```

3. Go to the directory where the oracle COBOL sample programs reside.

   ```
   # cd   /oracle/o10g/precomp/demo/procob2
   ```

4. Precompile the '.pco' file to generate a '.cob' file.

   ```
   # procob sample1.pco
   ```

5. Compile the generated '.cob' file to generate an '.acu' executable file.

   ```
   # ccbl sample1.cob
   ```

6. Run the generated executable file.

   ```
   # runcbl sample1.acu
   CONNECTED TO ORACLE AS USER: SCOTT

   ENTER EMP NUMBER (0 TO QUIT): 7900

   EMPLOYEE    SALARY    COMMISSION
   --------    ------    ----------
   JAMES       950.00          NULL

   ENTER EMP NUMBER (0 TO QUIT): 0
   ```

```
TOTAL NUMBER QUERIED WAS 0001.

HAVE A GOOD DAY.
```

7. Repeat steps 4–6 for the rest of the sample programs. Read the prolog in each sample program for any additional setup procedure.

7.3 Running MicroFocus Cobol

The environment variables COBDIR and LD_LIBRARY_PATH must be set to use the MicroFocus Cobol Compiler. The MicroFocus Server Express V4.0 Compiler was installed in the directory /opt/MFcobol. The MicroFocus binaries must also be added to the PATH variable. The following settings were used:

```
export COBDIR=/opt/MFcobol
export LD_LIBRARY_PATH=/opt/MFcobol/lib:$LD-LIBARY_PATH
export PATH=$PATH:/opt/MFcobol/bin
```

7.3.1 Makefile for sample Pro*COBOL programs

There is a makefile, demo_procob.mk, that can be used to build the sample program using the precompiler and the Micro Focus COBOL compiler.

Some of the sample programs have a prerequisite that certain .sql scripts need to be run. **make** will run the scripts if the parameter RUNSQL=run is added to **make**. The sample program sample5 is a SQL*Forms user_exit and is bypassed in the **make** for the other samples. See the makefile and the Oracle Forms and Oracle Forms Services 10g documentation about integrating a user_exit within a trigger or subprogram.

Running **make -f demo_procob.mk samples RUNSQL=run**, builds all of the samples except sample5. Running **make -f demo_procob.mk clean**, fails to clean up the lobdemo1 sample. Be careful not to remove lobdemo1.pco if doing a manual cleanup for lobdemo1, as this is the Pro*COBOL precompiler input loaded from the companion CD.

7.3.2 Makefile output for sample1 program

The samples can be built individually using the makefile. Alternatively, the Pro*Cobol precompiler and MicroFocus compiler could be used separately. The following example shows details for sample1 using the makefile.

```
oracle@pazxxt03:/oracle/10g/precomp/demo/procob2> make -f demo_procob.mk
sample1
make -f /oracle/10g/precomp/demo/procob2/demo_procob.mk build
COBS=sample1.cob EXE=sample1
```

```
make[1]: Entering directory `/oracle/10g/precomp/demo/procob2'
procob  iname=sample1.pco

Pro*COBOL: Release 10.1.0.3.0 - Production on Tue Jul 19 11:25:38 2005

Copyright (c) 1982, 2004, Oracle.  All rights reserved.

System default option values taken from:
/oracle/10g/precomp/admin/pcbcfg.cfg

cob -C IBMCOMP -C NESTCALL  -x -o sample1 sample1.cob -L/oracle/10g/lib/
/oracle/10g/precomp/lib/cobsqlintf.o -lclntsh `cat /oracle/10g/lib/ldflags`
`cat /oracle/10g/lib/sysliblist` -ldl -lm
* Ignored  - NESTCALL
make[1]: Leaving directory `/oracle/10g/precomp/demo/procob2'
oracle@pazxxt03:/oracle/10g/precomp/demo/procob2>
```

The compiler directive NESTCALL is ignored by MicroFocus Server Express 4.0.

7.3.3 Execution of sample1 program

The sample program sample1 depends on the demobld.sql script. This script will already have been run if the RUNSQL=run option was specified on the **make** described above. The script can be run directly from $ORACLE_HOME/sqlplus/demo/demobld.sql. Sample1 depends on the demo tables being in the SCOTT schema and the SCOTT password being TIGER. With sample1 built and the necessary tables created, the following shows the execution of the sample1 embedded SQL application.

```
oracle@pazxxt03:/oracle/10g/precomp/demo/procob2> ./sample1
CONNECTED TO ORACLE AS USER: SCOTT
ENTER EMP NUMBER (0 TO QUIT): 7788
EMPLOYEE    SALARY    COMMISSION
--------    ------    ----------
SCOTT      3000.00         NULL

ENTER EMP NUMBER (0 TO QUIT): o
TOTAL NUMBER QUERIED WAS 0001.

HAVE A GOOD DAY.

oracle@pazxxt03:/oracle/10g/precomp/demo/procob2>
```

Archived

7.3.4 User programs

The demonstration makefile may be used to create user programs when the MicroFocus COBOL compiler is used. The general syntax for linking a user program with the demonstration makefile is:

```
$ make -f demo_procob.mk target COBS="cobfile1 cobfile2 ..." \
 EXE=exename
```

For example, to create the program myprog from the Pro*COBOL source myprog.pco, use one of the following commands, depending on the type of executable and use of shared library resources desired.

For a dynamically linked executable with client shared library:

```
$ make -f demo_procob.mk build COBS=myprog.cob EXE=myprog
```

For a statically linked executable without client shared library:

```
$ make -f demo_procob.mk build_static COBS=myprog.cob EXE=myprog
```

For a dynamically loadable module that is usable with the Oracle Run Time System, rtsora:

```
$ make -f demo_procob.mk myprog.gnt
```

An attempt to use the MicroFocus provided runtime system, cobrun, will result in the following error:

```
$ cobrun myprog.gnt
Load error : file 'SQLADR'
error code: 173, pc=0, call=1, seg=0
173     Called program file not found in drive/directory
```

7.4 Oracle 10g Pro*C/C++ Precompiler

To use Oracle Pro*C/C++ you will need to install it from the Companion CD.

Run the following to unpack the files. This will create a directory structure with a Disk1 directory. In the Disk1 directory you will see an executable program called runInstaller. You will run this later, but there is further setup to be done first.

```
$ cpio -idcmv < ship_lnx390_ccd.cpio
```

7.4.1 Run the Installer

You will need to open a VNC window. Open the window as root and then **su** oracle to get to the oracle ID.

Now go the directory where you unpacked the cpio file and go to the Disk1 directory. In this directory there is a script called runInstaller. To start the OUI enter:

```
$ ./runInstaller
```

It will take a few minutes for the OUI initial screen to display in the VNC session.

Once the OUI starts, you will be presented with the Welcome screen. Click the **Next** button to continue.

The next screen will be Specify File Location. In this screen you need to verify that your ORACLE_HOME is the correct one, the one you used when installing the DB section. If the fields are correct click the **Next** button to get to the Select a Product to Install screen.

In the Select a Product to Install screen select the first entry **Oracle Database 10g Products 10.1.0.3.0**. This will install the Pro*C/C++.

7.4.2 Pro*C/C++ demonstration programs

Demonstration programs are provided to show the features of the Pro*C/C++ precompiler. There are three types of demonstration programs: C, C++, and Object programs. All of the demonstration programs are located in the $ORACLE_HOME/precomp/demo/proc directory. By default, all programs are dynamically linked with the client shared library.

To run, the programs require the demonstration tables created by the $ORACLE_HOME/sqlplus/demo/demobld.sql script to exist in the SCOTT schema with the password TIGER.

Use the demo_proc.mk make file, located in the $ORACLE_HOME/precomp/demo/proc/ directory, to create the demonstration programs. For example, to precompile, compile, and link the sample1 demonstration program, enter the following command:

```
$ make -f demo_proc.mk sample1
```

To create all of the C demonstration programs for Pro*C/C++, enter:

```
$ make -f demo_proc.mk samples
```

Note that you must unlock the SCOTT account and set the password before creating the demonstrations.

7.4.3 Creating demo tables

After the installation of the Companion CD completes, you need to create the demo tables and unlock user SCOTT using the following commands:

```
$cd $ORACLE_HOME/sqlplus/demo
$sqlplus '/as sysdba'
SQL> @demobld.sql

$ sqlplus '/as sysdba'
SQL> alter user scott account unlock;

User altered.
SQL> exit

$ sqlplus
SQL*Plus: Release 10.1.0.3.0 - Production on Mon Aug 2 21:21:14 2004
Copyright (c) 1982, 2004, Oracle.  All rights reserved.
Enter user-name: scott
Enter password:
ERROR:
ORA-28001: the password has expired
Changing password for scott
New password: tiger
Retype new password: tiger
Password changed

Connected to:
Oracle Database 10g Enterprise Edition Release 10.1.0.3.0 - 64bit
Production
With the Partitioning, OLAP and Data Mining options

SQL> exit
Disconnected from Oracle Database 10g Enterprise Edition Release 10.1.0.3.0
- 64
bit Production With the Partitioning, OLAP and Data Mining options
```

The demo tables could have also been created by putting the option RUNSQL=reun on the make command.

7.4.4 Precompile and compile C source

These are the commands we used to run the precompiler and to compile the source:

```
proc [pgm_name].pc
make -f demo_proc.mk OBJS=[pgm_name].o EXE=[pgm_name] build
```

Archived

To build a simple Pro*C program (let us call it myprog.pc) it should be sufficient to issue the command:

```
make -f demo_proc.mk build EXE=myprog OBJS=myprog.o
```

The following explains each of these options used in the **make** command:

- **make -f demo_proc.mk**

  This runs the make utility and identifies the parameter file to use.

- **build**

  This identifies the rule within the makefile to use.

- **EXE=myprog**

  This identifies the name of the target executable you want to build.

- **OBJS=myprog.o**

  This identifies the components other than Oracle Libraries that will be used to build the program.

If you look at the makefile using an editor, the build rule is as follows:

```
build: $(OBJS)
$(CC) -o $(EXE) $(OBJS) -L$(LIBHOME) $(PROLDLIBS)
```

This tells the make utility to ensure that $(OBJS) is up to date. In this simple case, myprog.ot uses the .pc.o rule to precompile and compile your .pc file. Then it calls the C compiler to link the program into $(EXE), which you passed on the command line.

You will notice that there are only two more entries, -L$(LIBHOME) $(PROLDLIBS). These are both defined in env_precomp.mk, and translate to the Oracle Library Directory ($ORACLE_HOME/lib), and the list of library files that a precompiled executable will require, to link successfully.

There are four default rules provided in the makefile:

```
build, which we have just covered,
build_static, which builds a statically linked executable,
cppbuild, for building C++ programs,
cppbuild_static, for building statically linked C++ programs.
```

7.4.5 Creating and executing sample2

To test the sample2 program we used the following commands:

```
$ cd $ORACLE_HOME/precomp/demo/proc
$ proc sample2.pc
$ make -f demo_proc.mk OBJS=sample2.o EXE=sample2 build
```

```
$ ./sample2
Connected to ORACLE as user: SCOTT
The company's salespeople are--
Salesperson   Salary   Commission
-----------   ------   ----------
ALLEN        1600.00       300.00
WARD         1250.00       500.00
MARTIN       1250.00      1400.00
TURNER       1500.00         0.00

Arrivederci.

$
```

Archived

Archived

8

Monitoring VM and Linux

This chapter is based on work done monitoring Oracle on Linux on VM in IBM Montpellier and at Oracle HQ in Redwood Shores. We used ESALPS software from Velocity Software for our tool. See “ESALPS overview” on page 224 for more details.

Archived

© Copyright IBM Corp. 2005. All rights reserved.

8.1 Oracle measurements

The environment used for these measurements was a 32-processor z990, of which 16 processors were dedicated to the z/VM LPAR. The VM LPAR configuration included 40 GB of main storage and 14 GB of expanded storage. z/VM 4.4 was utilized, but no significant changes were expected if we had used z/VM 5.1.

8.2 Configuration guidelines

In the z/VM shared resource environment, the objective is to share the resources as effectively as possible. In a shared resource environment, any server requiring excessive levels of a resource takes away from other servers' abilities to use that resource. Minimizing and controlling each server's resource consumption is critical. The following guidelines focus on minimizing resource requirements, sharing resources, and providing optimal service levels in an Oracle for Linux on zSeries with z/VM environment.

The following sections discuss various guidelines for configuring your z/VM and Linux environment. Other recommendations can be found in the Web pages provided by z/VM development. See:

http://www.vm.ibm.com/perf/tips/2gstorag.html

8.2.1 Minimize Total Storage Footprint®

The storage requirement of a Linux guest of z/VM should be reduced as much as possible. This is a shared resource environment where memory used by one server takes memory resource away from other servers. Linux tends to use all storage defined to it, so the *working set size* of the Linux guest usually approaches the size of the virtual machine. On other platforms, without z/VM, performance is usually improved by adding memory. In the z/VM environment, however, *price/performance* is improved by sharing resources effectively. Giving servers more storage than they need impacts all other servers.

There are two parts to this requirement. The first requirement is to reduce the overall storage requirement by minimizing the virtual machine size and eliminating unneeded processes.

The second requirement is to reduce the requirement for z/VM storage below the 2 GB bar. In measurements in z/VM 4.4, contention for storage below 2 GB proved to be a serious bottleneck. Many of the following guidelines help to address this issue. Technology is rapidly changing, with improvements being

made both to Linux and to z/VM that will reduce or eliminate this bottleneck. Until this newer technology is available, we recommend that you implement the recommendations in the following sections.

One of the obvious ways to reduce the impact of the 2 GB issue in z/VM is to use the Fixed Buffer option available in SLES9 SP1 and in RHEL4 Update 1. See http://www-128.ibm.com/developerworks/linux/linux390/perf/index.html and click **Fixed I/O buffers** for more information, including how to enable this option.

The following ESAUCD2 report (Linux Memory Analysis) from ESALPS shows several Linux servers, most of which were defined as 2 GB virtual machines. The data in the report is provided from inside Linux using the NETSNMP agent. Storage is broken down into storage in use and storage available.

The servers that have almost 2 GB of available storage are likely very lightly loaded. Linux will consume real storage as cache whenever possible. If servers continuously have large amounts of available storage, or a large amount of storage in use as cache, then the size of the server should be reduced by at least half of the available storage.

To make the most efficient use of real memory, Linux automatically uses all free RAM for buffer cache, but also automatically makes the cache smaller when programs need more memory. The amount of storage used for buffer cache is reported in the right-hand column of the ESAUCD2 report, as shown in Figure 8-1 on page 174. This is the amount of storage currently allocated to cache. When Oracle starts up, it will allocate the SGA. The amount of storage in the cache will be reduced to allocate the storage for the Oracle SGA.

If there is no swap storage used, then there was always sufficient storage to meet the requirements in the Linux guest so far. If no swap storage was used and the cache size is very large, then you could reduce the virtual machine size so that the cache size is 50 MB to 100 MB larger than the SGA. As long as swap storage remains little or unused, then the storage size is sufficient.

Archived

```
Report: ESAUCD2      LINUX UCD Memory Analysis Report

Linux Test
Monitor initialized: 05/11/04 at 13:19:32 on 2084 serial 83D2A   First record
-------------------------------------------------------------------------------
Node/    <-------------------------Storage Sizes (in MegaBytes)------------->
Time/    <--Real Storage--> <-----SWAP Storage----> Total <--Storage in Use->
Date     Total  Avail Used  Total Avail Used  MIN   Avail Shared Buffer Cache
-------- ------ ----- ----- ----- ----- ----- ----- ----- ------ ------ -----
13:21:00
PAZXOB03 1995.0 153.6  1841  1464  1034 429.9  15.6  1187      0   17.3  1424
PAZXOB04 1995.0  67.5  1927  1424  1424     0  15.6  1491      0   37.6  1645
PAZXOQ01 1882.8   2.0  1881  1424 539.5 885.0  15.6 541.5      0    1.3  1654
PAZXOQ03 1995.0  1805 189.9  1024  1024     0  15.6  2829      0   12.3 120.1
PAZXXQ01 1882.8   2.0  1881  1424 858.0 566.5  15.6 860.0      0    2.7  1590
PAZXXQ02 1882.8  28.7  1854  1424  1329  95.2  15.6  1357      0    6.5  1335
PAZXXQ04 1995.0  1790 205.1  1024  1024     0  15.6  2814      0   24.4 123.2
PAZXXQ05 1995.0  1793 202.1  1024  1024     0  15.6  2817      0   22.6 122.5
PAZXXQ03 1995.0  1612 383.5  3072  3072     0  15.6  4683      0   41.1  69.2
PAZXXQ06 3994.8  3789 205.4  3072  3072     0  15.6  6861      0   21.5 121.3
t03       502.6 189.5 313.0     0     0     0  15.6 189.5      0    4.6 267.1
```

Figure 8-1 ESAUCD2 report (Linux memory analysis)

8.2.2 SGA must fit in memory

The Oracle SGA is virtual storage, and you could define an SGA larger than the Linux server, but this is definitely not recommended. When the SGA is larger than available memory, you will see high rates of Linux swapping. Free storage in Linux is assigned to the disk cache. If this storage is required, say to back a page of the SGA, the page will be removed from the cache and assigned to the SGA. In this way you can think of extra cache as available storage.

The SGA size is normally set by specifying each component of the SGA and the size will be the sum of the sizes of all the components.

There are several ways of determining if an Oracle server is swapping. The following data based on the z/VM monitor data shows that the ORACLEC server did 2,771 I/Os in the last 60-second interval to the Virtual Disk swap area. Each I/O could be a transfer of a number of pages. If the swap files were to a dedicated disk, the swap activity would show up as DASD I/O instead of Virtual Disk I/O. Note that we recommend the use of VDISK for swapping.

Swap I/O is having minimal impact in this environment. Degradation starts to become obvious when swapping gets into the hundreds or thousands per

second. If the SGA is larger than the cache available, Linux will start swapping at a high rate when Oracle becomes very active. If this occurs, you will want to either reduce the size of the SGA further, or increase the size of the server to accommodate the SGA requirements—with the understanding that storage given to one server takes away from the other servers.

```
Report: ESAUSR3      User Resource Utilization - Part 2
Monitor initialized:
---------------------------------------------
                DASD MDisk Virt Cache I/O
UserID    DASD Block Cache Disk   Hit Prty
/Class     I/O   I/O  Hits  I/O   Pct Queued
-------- ----- ----- ----- ---- ------ -----
 ***User Class Analysis***
*Servers   678     0   649    0  95.7      0
*Keys        0     0     0    0     0      0
*TheUsrs    19     0    15    0  78.9      0
ORACLE    215K     0 35502 2801  17.6      0
 ***Top User Analysis***

ORACLEC  76780     0 11973 2771  18.5      0
ORACLEA  73748     0 12339   16  16.7      0
ORACLEB  63582     0 10997   14  17.3      0
ESAWRITE    34     0    12    0  35.3      0
```

Figure 8-2 ESAUSR3 (user resource utilization)

8.2.3 Use Oracle direct I/O

For I/O in our measurements using the EXT2 file system, enabling the direct I/O option in Oracle 10g had very positive performance effects. With direct I/O, Oracle does all the I/O from the Oracle database cache in the SGA. Without this option, the data for a database write will be moved to the Linux file system cache. When the actual I/O operation takes place from the file system cache, there are a large number of different addresses that could used for the I/O operations than if all the I/O comes from the Oracle db_buffers. This increases the working set of the Linux virtual machine, but much more importantly increases the requirement for low storage in z/VM. When z/VM 4.4 or 5.1 does an I/O operation, all the pages associated with the operation are moved to pages with real addresses less than 2 GB. At high I/O rates, contention for storage below 2 GB real can be a severe bottleneck. One of the most positive tuning impacts in today's Linux under z/VM environment is to reduce z/VM's storage requirement below the 2 GB bar. The options are to reduce LInux storage, force the I/O to use only the Oracle buffers in the SGA, and to minimize the size of those buffers.

Archived

Experiments where direct I/O was used greatly reduced the impact on the below the 2 GB bar storage. This option is enabled by specifying filesystemio_options=directIO in the init.ora parameter file. Obviously, the use of direct I/O will not help if the total size of the db_buffers in Oracle is more than 2 GB. If the total size is more than 2 GB, then you should ensure the you are using fixed I/O buffers as discussed in “Use Oracle direct I/O” on page 175.

8.2.4 Use virtual disk for swap

With virtual machines being defined as small as possible to effectively share memory, there is a potential problem of running out of memory. When Linux runs out of real pages of memory, the **killmem** process runs, trying to kill processes in the system that are using large amounts of memory. Often the processes killed are essential to Oracle operation, and the system fails. This problem is resolved by providing swap space, so that Linux can move assigned pages to the swap file to free them. The best device for swap in this shared resource environment is a virtual disk. Define a large amount of virtual disk for each Linux server for swap. Possibly you will need to increase the system and user limits for virtual disk with the commands shown in Example 8-1.

Example 8-1 VDISK limits

```
SET VDISK SYSLIM INFINITE
SET VDISK USERLIM 2G
```

It is fairly safe to define virtual disk in large amounts, as the cost to defining the storage is almost nothing. The pages in the virtual disk are only allocated when referenced, and then it is allocated in normal z/VM storage as any other address space page. These pages are pageable when un-referenced, and may be paged out to expanded storage and Disk paging storage. However, you should note that the DAT structures to create the VDISK (segment tables and PGMBKs) are defined as non-pageable and must reside below 2 GB. Over defining them could actually add to contention below 2 GB.

Define multiple virtual disks for swap

One other very useful guideline for virtual disk for swap is to define two or three virtual disks and prioritize them, rather than just one large virtual disk. When the swap disks are assigned a priority, Linux will use the first disk until it is completely full, and then will start using the second disk. The reason this is useful to implement is because of the way Linux allocates swap areas. It allocates using a moving cursor from the front of the space to the end of the space. This forces all of the space to be allocated over time by z/VM. If the size of the first virtual disk is smaller than the maximum requirement, then Linux will never start allocating into the second virtual disk. This reduces the overall requirements placed on the z/VM storage subsystem.

The following virtual disk report shows the virtual disk swap configuration for servers DDRXXX01, DDRZXX02, PAZXOB01, and PAZXOQ01. This shows that there is a virtual configuration of about 1,000,000 pages, of which about 350,000 pages are actually referenced and in use. The system limit is set to 75 GB (76800 MB), with a user limit set at 4 GB each. Each virtual disk has an internal name (shown here) that shows the user ID, the virtual disk address, and an internal counter. Note that the 207 disk for PAZXOB03 is used, but the 208 and 209 disks are not. This means they were set up correctly and prioritized. When all of the virtual disks have the same allocation, they are not correctly prioritized and much more resource is required.

```
Report: ESAVDSK      VDISK Analysis Report    x Test          Page 515
Monitor initialized: 05/11/04 at 13:19:32 on 2t record analysis
-----------------------------------------------------------------
Maximum VDISK:     Blocks  (MB)
 System storage:     157M 76800
 Storage per user:  8389K  4096
                                  <--Size--->  <--pages--> <DASD    X-
                                  AddSpc VDSK  Resi- Lock-  Page Store
Owner    Space Name               Pages  Blks   dent    ed  lots  Blks
-------- ------------------------ ----- -----  ----- ----- ----- --
13:21:00
DDRZXX01 VDISK$DDRZXX01$0201$0015 25840  205K    32     0     0     4
DDRZXX01 VDISK$DDRZXX01$0202$0016 25840  205K    32     0     0     4
DDRZXX02 VDISK$DDRZXX02$0201$0017 25840  205K    32     0     0     4
DDRZXX02 VDISK$DDRZXX02$0202$0018 25840  205K    32     0     0     4
PAZXOB01 VDISK$PAZXOB01$0201$0001 51440  410K    60     0     0     7
PAZXOB01 VDISK$PAZXOB01$0203$0003 51440  410K    60     0     0     7
PAZXOB01 VDISK$PAZXOB01$0501$0002 51440  410K    60     0     0     7
PAZXOB03 VDISK$PAZXOB03$0207$0028  262K 2097K  232K     0     0     0
PAZXOB03 VDISK$PAZXOB03$0208$0029 51440  410K     4     0     0     0
PAZXOB03 VDISK$PAZXOB03$0209$002A 61424  491K     4     0     0     0
PAZXOQ01 VDISK$PAZXOQ01$0207$0006  262K 2098K  116K     0     0     0
PAZXOQ01 VDISK$PAZXOQ01$0208$0007 51440  410K     2     0     0     0
PAZXOQ01 VDISK$PAZXOQ01$0209$0008 51440  410K     2     0     0     0
```

Figure 8-3 ESAVDSK VDISK Analysis Report

8.2.5 Enable the timer patch

z/VM manages users with a fair share scheduler that manages the dispatchability of all virtual machines by keeping the machines in a number of queues. In a well-running environment where Linux servers are running with proper behavior, the Linux servers will drop from queue at idle times and free up the storage that has been assigned to them. Each server should spend some

amount of time in Q1, Q2, and Q3, depending on what they are doing. The worst case (such as that shown here) is that the servers remain in Q3 and retain their storage, even when idle.

Standard Linux has a scheduler that wakes up 100 times per second, which is every 10 ms. For the virtual environment this keeps the virtual machine *in queue*, as CP will not drop a user from queue unless it is idle for 300 ms. Waking up every 10 ms to do process accounting and other housekeeping, even when idle, breaks the virtualization model.

There is a patch, often referred to as the timer patch, that is now standard with SLES8 or current releases of RedHat. Disabling the patch will increase your storage and CPU requirements.

`sysctl -w kernel.hz_timer=1`, or **`echo 1 > /proc/sys/kernel/hz_timer`** will DISABLE the timer patch, and there will be a timer interrupt every 10 ms.

`sysctl -w kernel.hz_timer=0` or **`echo 0 > /proc/sys/kernel/hz_timer`** will ENABLE the timer patch, and timer interrupts will be much less likely.

In the following user queue analysis, in a properly working environment, the users would be placed in Q1, Q2, and Q3. Instead, the virtual machines remain in Q3 indefinitely. Q3 is the queue for long-running work. By taking a timer interrupt every 10 ms, the Linux guests appear to be long running. If the virtual machine never drops from queue, its pages will not have their reference bits reset, and their storage will never be trimmed. This must be corrected to provide an optimal running virtual environment. Note that some of the virtual machines seem to be counted more than one time. These virtual machines have multiple virtual processors, of which at least one is always active and in queue.

Archived

```
Report: ESAUSRQ      User Queue and Load Analysis                  Linux
Monitor initialized: 05/11/04 at 13:19:32 on 2084 serial 83D2A     First
-----------------------------------------------------------------------
         <-----------User Load------------>      <-----------Average Num
UserID   Logged  Non-          Disc- Total  Tran <-------Dispatch List--
/Class       on  Idle  Active  conn  InQue  /min    Q0    Q1    Q2    Q3
-------- ------ ----- ------- ------ -----  ---- ----- ----- ----- -----
13:21:00   67.0     .    35.0      .  36.0   216   2.0     0     0  34.0
Hi-Freq:   67.0    35    35.0     57  32.9   218   2.0   0.1   0.1  30.7
 ***Top User Analysis***
PAZXXQ01    1.0     1     1.0      1   1.0     0   0.0     0     0   1.0
PAZXOB03    1.0     1     1.0      1   1.0     0     0     0     0   1.0
PAZXXQ02    1.0     1     1.0      1   1.0     0   0.0     0     0   1.0
PAZXOQ01    1.0     1     1.0      1   1.0     0     0     0     0   1.0
PAZXOQ02    1.0     1     1.0      1   1.0     0   0.1     0     0   0.9
PAZXOQ06    1.0     1     1.0      1   2.1     0   0.0     0     0   2.1
PAZXXQ06    1.0     1     1.0      1   1.4     0     0     0     0   1.4
PAZXOQ05    1.0     1     1.0      1   2.1     0     0     0     0   2.1
```

Figure 8-4 ESAUSRQ - User queue and load analysis

8.2.6 Use virtual switch

Linux servers must be allowed to drop from the VM scheduler queue because that is the only time z/VM will trim their storage to be trimmed and page out unreferenced pages. In cases where the Linux servers do not drop from queue, their storage remains assigned, and will be in real storage even if the server is completely idle. This holds resources at the expense of the servers currently performing work. Besides the above-mentioned timer patch, another reason for Linux servers to remain in queue is active I/O. Servers waiting for an active I/O operation, including I/O for most communications adapters such as virtual channel-to-channel or dedicated OSA adapters, do not drop from queue.

Until z/VM 4.4 APAR VM63282, a server with outstanding I/O to a dedicated OSA adapter will not drop from queue. A server with a dedicated OSA adapter that runs a batch workload overnight will retain its storage, including any storage allocated below the 2 GB bar, even though completely idle during the day. This APAR is built into the z/VM 5.1 base, and so dedicated OSA adapters can be used safely.

As of z/VM 4.4, servers attached to a guest LAN or utilizing a VSWITCH (virtual switch) to access the Internet are eligible to drop from queue while waiting for network I/O. The virtual switch is also expected to use much less resource than having a guest Linux act as a virtual router.

8.2.7 Use expanded storage for paging

Because z/VM is constrained by its use of low storage below the 2 GB bar, there must be very fast paging space. As of z/VM 4.4, any pages moved from below the 2 GB line must be moved to paging space, and Estore is the fastest page device available. In systems that are constrained on their 2 GB storage but have limited expanded storage you will see lots of paging to DASD—even if there is central storage available. This problem was found during these measurements and found to cause extreme degradation.

To avoid these performance problems on installations constrained by the CP low storage, you should have enough expanded storage configured such that little or no paging is ever done to DASD. Some installations may find that even half of the main storage should be reconfigured as expanded storage for paging.

Mini-Disk Caching (MDC) can also compete with paging for the EStore frames. We feel that because the defaults for MDC are excessive, MDC can and should reside only in main storage. We recommend that the MDC size in expanded storage should be set to zero with the command SET MDC XSTORE 0M 0M.

8.2.8 Ensure sufficient page space

On systems that have non-zero paging rates, it is important to understand that the page space needs to be sufficient to hold two times the requirement of all your guests. Thus for each 2 GB server there should be the equivalent of two 3390-3 added to the paging subsystem. As some of the servers will fit in main storage, the amount of main storage can be deducted from the page requirement.

There are two issues with page space. When the system runs out of page space, it crashes and re-IPLs itself. This means all Linux servers must be rebooted.

The second issue is for performance. The paging subsystem performs much better when there is about 50 percent free space.

The following is the first half of the ESAPAGE report showing that this system has 18 GB of expanded storage available, of which there are page movements in the 1000s per second. In the second part of the report, it shows that 84 GB of paging storage is available on disk, of which 2 GB are in use. This system is over-configured for this particular workload but shows where to watch to make sure you do not run into a page space full condition.

```
Report: ESAPAGE      Paging Analysis
Monitor initialized: 05/11/04 at 13:19:32 on 2084
--------------------------------------------------
          <----------Expanded Storage----------->
           (MB) <------pages/second------>  Page
Time      Avail  Alloc Relse   PGIN  PGOUT  Age
--------  ----- ------ ----- ------ ------  -----
13:21:00  18432   3526  3066 3039.9 3520.8  630.0
13:22:00  18432   3938  2072 2026.6 3938.0  528.0
13:23:00  18432   3238  4995 3655.9 1874.3  612.0
13:24:00  18432   3559  3483 3071.4 3106.9  595.0
13:25:00  18432   2127  3618 3551.9 2124.9  650.0
13:26:00  18432   2806  2154 1763.7 2505.6  666.0
13:27:00  18432   2814  2973 2763.4 2608.2  669.0
13:28:00  18432   4925  6181 3950.7 2539.2  673.0
**************************************************
Average:  18432   3366  3567 2978.3 2777.9  627.9
```

Figure 8-5 ESAPAGE - Paging analysis 1

```
Report: ESAPAGE    Paging Analysis
-------------------------------------------
           <-----Paging----->  Page  Space
           <-pages/sec-> Resp  <megabytes>
Time         Read  Write Time  Avail InUse
--------  ------ ------ ----  ----- -----
13:21:00       0      0 18.2  84510     2
13:22:00       0      0 18.2  84510     2
13:23:00       0      0 18.2  84510     2
13:24:00       0      0 18.2  84510     2
13:25:00       0      0 18.2  84510     2
13:26:00       0      0 18.2  84510     2
13:27:00       0      0 18.2  84510     2
13:28:00       0      0 18.2  84510     2
**********Summary**************************
Average:       0      0 18.2  84510     2
```

Figure 8-6 ESAPAGE - Paging analysis 2

8.3 Storage analysis

In the measurements, after determining that CP storage below the 2 GB line was a limiting resource for I/O-intensive jobs, this became one of the primary focuses of each test.

Archived

8.3.1 Detecting storage problems - Paging

Lack of storage results in paging to either expanded storage or DASD. This is not by itself bad. As virtual machines go idle, they should be paged out—why keep virtual machines in main storage if they are idle? And then when the server becomes active again, the pages are paged back in. There are rules of thumb about how much an installation can overcommit storage. Such a rule of thumb leaves out one very important factor—what percent of your servers are idle? If at peak time 75 percent of the servers are idle, then over committing by at least a factor of four would make sense.

In capacity planning for storage, the key metrics are working sets and percent active. Even though a virtual machine has 2 GB defined does not mean it needs 2 GB of storage to operate. Pages that are not referenced do not need to be resident. Thus the required storage, the *working set*, may be much smaller than the virtual machine size. An active server is a server that performs some work (consumes some CPU time) in a minute. WIth Linux servers that wake up every second even with the timer patch, it is difficult to define an active server. But evaluating CPU requirements over time of the servers will show when the virtual machine is above the minimal idle operating threshold, and now you have the input to calculate the active percent. During the active intervals determine the working set.

8.3.2 Detecting 2 GB storage problems - Paging

The following subsystem summary from the ESALPS (ESASSUM) report in listing format is very similar to the ESAMAIN display used during the experiments. Showing the 15-minute experiment, minute by minute, you can see the processor requirement going up, with the page rate to expanded storage as well as the I/O rate going up. The minidisk cache rate is the number of I/O eligible for MDC, and the %hit is the percent of those that were MDC hits that did not result in a real I/O. The capture ratio is provided, showing how much of the processor that was used can be accounted for. One hundred percent is typical; anything less identifies a problem with the monitor or a bug. With lower capture ratios, there is CPU used by some virtual machine that is not identified.

```
Report: ESASSUM Subsystem Activity  MOP Oracle experiments  ESAMAP 3.4.1 07/25/04 9
Monitor initialized:   on 2084 serial 1F86C First record analyzed: 07/24/04 10:34:01

--------------------------------------------------------------------------------
         <Processor> Storage (MB) <-Paging--> <-----I/O-----> <MiniDisk> Spool Captur
         Utilization Fixed Active <pages/sec> <-DASD--> Other <-Cache-->  Page  Ratio
Time      Total Virt.  User Resid. XStore DASD Rate Resp  Rate  Rate %Hit  Rate (pct)
-------- ----- ----- ----- ------ ------ ---- ---- ---- ----- ----- ---- ----- ------
07/24/04
11:13:01     4     2 161.5 7268.9      0    0  180  0.5     0   1.2 58.7     0 100.00
11:14:01    22    19 161.3 7269.2      0    0  487  0.5     0  71.4 78.8     0 100.00
11:15:01    74    68 159.4 8082.8      0    0  819  1.2     0  1697 76.2     0 100.00
11:16:01   112    91 163.7 8497.6    268    0  638  0.8     0 724.9 85.4     0 100.01
11:17:01   130    65 160.4 8533.8   1540    0  666  0.7     0 305.8 72.1     0  99.99
11:18:01   134    63 158.7 8542.6   1572    0  783  0.7     0 228.8 62.7     0 100.00
11:19:01   136    59 160.3 8553.0   1515    0  825  0.7     0 175.3 58.2     0 100.01
11:20:01   137    60 157.2 8571.1   1533    0  858  0.7     0 159.2 55.5     0 100.00
11:21:01   130    55 160.3 8577.9   1636    0  810  0.7     0 127.0 53.3     0  99.99
11:22:01   135    55 157.5 8588.6   1575    0  772  0.7     0 119.6 55.3     0 100.00
11:23:01   128    54 158.4 8593.4   1647    0  678  0.7     0 101.1 54.8     0 100.00
11:24:01   121    49 162.6 8597.9   1482    0  612  0.7     0  89.7 54.9     1 100.00
11:25:01   123    50 158.6 8598.7   1361    0  623  0.7     0  84.1 54.0     0 100.00
11:26:01   126    48 161.8 8598.4   1369    0  617  0.6     0  72.2 54.7     0  99.99
11:27:01   135    47 162.6 8604.2   1359    0  682  0.7     0  69.4 55.1     0 100.01
11:28:01   135    47 159.0 8608.4   1403    0  499  0.6     0  65.1 55.6     0  99.99
11:29:01   133    50 157.3 8613.9   1368    0  715  0.6     0  62.8 56.7     0 100.03
11:30:01   134    46 163.6 8618.0   1371    0  690  0.6     0  54.9 55.8     0  99.97
11:31:01   138    48 162.7 8626.0   1274    0  673  0.6     0  57.9 56.9     0 100.00
11:32:01   138    52 158.0 8586.2   1401    0  809  0.6     0  47.5 58.8     0 100.00
11:33:01    39    13 158.9 8530.7    827    0  559  0.5     0   5.1 72.5     0  99.98
11:34:01     7     4 159.1 8524.8    118    0  190  0.5     0   0.4 72.7     0 100.00
*************************************Summary*************************************
Average:    63    40 159.1 7346.1    402    0  603  0.7   0.0 241.7 63.8     0  99.98
```

Figure 8-7 ESASSUM subsystem activity

During the experiments, the 2 GB issue was the primary concern. There are several ways to identify this problem. For most of these experiments, paging to expanded storage above 1500 per second indicated that the limit was reached. The measurement above shows an example of one of these experiments.

Other installations with other configurations have had similar thresholds, sometimes higher, sometimes lower.

8.3.3 Detecting 2 GB problems - Demand scan

ESALPS also maintains a performance database with many data items that are not usually required. In analyzing this problem, the following performance database extract was done for people literate in VM storage algorithms will actually mean something. VM maintains two available lists of storage, one for

below the 2 GB bar and one for above the bar. When either becomes smaller than a threshold, VM will attempt to find more pages. It does this with a facility called demand scan. There are three forms of demand scan: Scan 1, scan 2, and emergency. As you could expect, emergency scan is a bad thing.

What the following PDB™ extract shows minute-by-minute is the key areas where CP was getting pages. The extract and field descriptions follow this extract. This PDB extract shows that not only is CP having to go to demand scan 2 for a thousand pages per second, but is having to go to emergency scan as well.

Optimally, CP gets pages from dormant users. In this case, none of the users go dormant so that is not possible. And since all users are in queue, there is no distinction between a good and a bad virtual machine (an idle or an active virtual machine); CP is just as likely to take a page from the virtual machine that needs storage as one that does not. And in fact, this measurement was an 8 GB Linux server running by itself. So all page stealing from the dispatched virtual machine was indeed from the virtual machine that had an immediate requirement for storage below the line.

One of the other indicators of the 2 GB issue is that the CP monitor acts strange and seems to miss intervals. This extract explains why. The emergency scan is stealing pages from the monitor DCSS (PLSSRPE). The monitor at every 60 seconds collects a large amount of performance metrics, puts this data into the monitor DCSS, and before the monitor application can process, the data pages it out. It does not matter how efficient the data collector is. It cannot get to the data.

At the point of having emergency scans of this magnitude, steps *must* be taken to correct the problem, as little work will get done.

Archived

```
*HDR
 Date    Time    PLSDSPPLLLSDRMP2  PLSDSPP2  PLSDRMPE  PLSSHRPE  PLSDSPPE
 20040724 111401     0.0       0.0       0.0       0.0       0.0       0.0
 20040724 111501     0.0       7.5     207.5       0.0       0.2       0.0
 20040724 111601     0.0      18.6    1124.2       3.5      17.3     166.7
 20040724 111701     0.0      14.1    1050.8       3.2      14.4     225.9
 20040724 111801     0.0      11.5     939.6       4.0      13.9     267.0
 20040724 111901     0.0       6.6     951.4       3.2      12.4     265.6
 20040724 112001     0.0      10.1    1003.8       2.1      14.3     238.4
 20040724 112101     0.0       6.3     980.9       3.0      17.1     210.9
 20040724 112201    93.4      17.6     942.3       3.2      13.6     181.0
 20040724 112301   346.3      15.8     610.8       1.9      12.7     143.8
 20040724 112401   354.6      20.2     583.7       1.8      20.3      96.9
 20040724 112501   245.6      15.5     681.5       2.3       8.3     112.1
 20040724 112601     0.0       9.7     822.8       4.4      12.9     185.9
 20040724 112701     0.0       8.2     857.8       1.1       9.6     178.1
 20040724 112801     0.0      10.1     849.2       2.3      11.6     181.0
 20040724 112901     0.0      15.0     805.2       2.6      10.1     189.6
 20040724 113001     0.0       8.5     779.2       2.8      10.2     183.2
 20040724 113101     0.0       9.4     784.4       3.3      10.6     168.4
 20040724 113201   136.0      14.0     338.7       0.2       0.8      38.4
 20040724 113301    23.0       7.5      39.5       0.0       0.0       0.0
 20040724 113401    40.2       6.1      53.4       0.0       0.0       0.0
 20040724 113501    39.7       6.5      52.2       0.0       0.0       0.0
*MIN
 20040724 113501     0.0       0.0       0.0       0.0       0.0       0.0
*MAX
20040724 113501   354.6      20.2    1124.2       4.4      20.3     267.0

interval = 'INTERVAL'
y = 'STORSP.PLSDSPP1' ; 'Frames taken from dispatch users'
y = 'STORSP.PLSDRMP2' ; 'Frames taken from long-term dormant users'
y = 'STORSP.PLSDSPP2' ; 'Frames taken from dispatch users'
y = 'STORSP.PLSDRMPE' ; 'Frames taken from dormant users'
y = 'STORSP.PLSSHRPE' ; 'Frames taken from NSS and DCSS'
y = 'STORSP.PLSDSPPE' ; 'Frames taken from dispatch users'
```

Figure 8-8 Custom extract for demand scan measurements

8.3.4 Detecting 2 GB problems - State analysis

In solving many performance problems, the state analysis is normally very easy to understand. Each virtual machine has an assigned state, such as Running (using a processor), CPU Wait (ready to run), Sim Wait (CP Simulation wait), page wait, I/O wait, and others. In the following analysis, only the ORACLE2 virtual machine was in use for the experiment. Note that it is considered running

Archived

for 63 percent of the interval, and indeed, it was using 63 percent of a processor. But on this system with 16 processors, 15 of them idle, the CPU wait and Simulation wait are difficult to explain. It is also difficult to explain why the other Linux servers are also waiting for CPU, as this is not normal.

The normal resting state of most Linux guests is asynchronous I/O, as they sit waiting for I/O from the communications subsystem. But this is also the state if the servers are doing I/O. With the ORACLE2 server, there should be about 270 ms of I/O per second to explain what the server was doing. At no point in time should it be waiting for work such as in the case of waiting for communications. This experiment was for a 2 GB server that, of course, most of which fit below the line without contention, and it was performing 800 I/O per second.

The second case was for an 8 GB server performing the same work. Note that the I/O wait time is now only 3.4 percent or 34 ms per second—and it is only performing 10 I/O per second. Simulation wait is the majority wait state. The simulation wait is the time for CP to steal pages from below the line from the working server, page the pages to expanded storage, move the required page below the line, and re-dispatch the user. Thus, high simulation wait is another way to show the impact of the 2 GB problem.

The solution for this experiment is to use a smaller virtual machine.

```
Report: ESAXACT      Transaction Delay Analysis       MOP Oracle Benchmark
Monitor initialized:  on 2084 serial 1F86C     First record analyzed: 07/23/04
--------------------------------------------------------------------------------
                     <---------------Percent non-dormant---------------------->
UserID   <-Samples->                      E-  D-  T-     Tst <Asynch>        Lim
/Class   Total  In Q Run Sim CPU SIO Pag SVM SVM SVM  CF Idl I/O Pag Ldg Oth Lst
-------- ----- ----- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
11:19:01    42     9  11   0   0   0   0   0   0   0   0  11  78       .   0   0
Hi-Freq:  2520   822 8.4 2.2 2.7   0   0   0  28 0.1   0 3.0  55   0   0   0   0
 ***User Class Analysis***     0   0   0   0   0   0   0   0   0   0   0   0   0
*TheUsrs   540   125 4.8 2.4 1.6   0   0   0  48   0   0 1.6  42   0   0   0   0
ORACLE     480   479  13 2.9 4.2   0   0   0   0   0   0   0  80   0   0   0   0
 ***Top User Analysis***       0   0   0   0  87 0.5   0  12   0   0   0   0   0
ORACLE2     60    60  63 1.7 8.3   0   0   0   0   0   0   0  27   0   0   0   0
ORACLE3     60    59 5.1 5.1 6.8   0   0   0   0   0   0   0  83   0   0   0   0
ORACLE1     60    60 6.7 1.7 5.0   0   0   0   0   0   0   0  87   0   0   0   0
```

Figure 8-9 ESAXACT - Transaction delay analysis 1

```
Report: ESAXACT      Transaction Delay Analysis MOP Oracle experiments
Monitor initialized:   on 2084 serial 1F86C     First record analyzed: 07/23/04
-------------------------------------------------------------------------------
                     <---------------Percent non-dormant-------------------->
UserID   <-Samples->                      E-  D-  T-     Tst <Asynch>        Lim
/Class   Total  In Q Run Sim CPU SIO Pag SVM SVM SVM  CF Idl I/O Pag Ldg Oth Lst
-------- ----- ----- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
07/23/04
13:02:01    42    11  18   0   0   0   0   0   0   0   0   0  82       .   0   0
Hi-Freq:  5117  1661 5.8 8.5 3.7   0   0   0  19 0.8 0.2 2.3  59   0   0   0   0
 ***User Class Analysis***
*TheUsrs  1071   252 1.6 4.0 0.4   0   0   0  47   0   0 7.1  40   0   0   0   0
ORACLE    1071  1071 8.6  10 5.5   0   0   0   0   0 0.4   0  76   0   0   0   0
 ***Top User Analysis***
ORACLE3    119   119  45  50   0   0   0   0   0   0 0.8   0 3.4   0   0   0   0
ORACLE5    119   119  10 5.9 3.4   0   0   0   0   0   0   0  81   0   0   0   0
ORACLE2    119   119  12   0 8.4   0   0   0   0   0 0.8   0  79   0   0   0   0
ORACLE1    238   238 2.5 0.4  11   0   0   0   0   0 0.8   0  85   0   0   0   0
```

Figure 8-10 ESAXACT - Transaction delay analysis 2

8.4 I/O subsystem

In this section we discuss the I/O subsystem.

8.4.1 LVM

One of the options for defining disk storage to Linux is the Logical Volume Manager (LVM). There are two forms of LVM, one form that allows you to add storage to the logical volume, and the other being striped—where the logical volume is fixed in size but the I/O is striped across all volumes.

In the following analysis of each server's I/O, the question was why was ORACLE5 slower than ORACLE2 and ORACLE4, which were all supposedly identical. From the user seek analysis on the ESAUSEK report, the I/O activity on ORACLE2 and ORACLE4 seem localized. When one volume fills, the next volume is utilized. It would appear that these servers have the non-striped version of LVM defined. ORACLE5, on the other hand, has its I/O balanced across the striped volumes.

Archived

```
Report: ESAUSEK      User DASD Seeks Report ESAMAP 3.4.1 07/28/04  Page 707
Monitor initialized:                      First record analyzed: 07/28/04 10:00:00
Monitor period:        18000 seconds      Last record:           07/28/04 15:00:00
-------------------------------------------------------------------------------
 Userid    Dev Volume <--Minidisk-> <Cylinder> Total <---Non-zero---> Read <Pct. of>
 /Time     No. Serial Ownerid  Addr Start Stop Seeks Seeks Pct. Dist. Pct.  Sys  Vol
--------  ---- ------ -------- ---- ----- ---- ----- ----- ---- ----- ---- ---- ----
12:41:51
ORACLE2   1111 ORA200 ORACLE2  0500     010016  3283  3073 93.6  2724  0.7  0.0  100
          1112 ORA201 ORACLE2  0501     010016 4022K 4015K  100   812  0.0 10.1  100
          1113 ORA202 ORACLE2  0502     0 9093 4635K 4307K 92.9  1147  2.8 11.6  100
          1214 ORA203 ORACLE2  0503     0    0     8     0    0     0  100  0.0  100
          1314 ORA204 ORACLE2  0504     0    0     8     0    0     0  100  0.0  100
          1414 ORA205 ORACLE2  0505     0    0     8     0    0     0  100  0.0  100
          1514 ORA206 ORACLE2  0506     0    0     8     0    0     0  100  0.0  100
          1614 ORA207 ORACLE2  0507     0    0     8     0    0     0  100  0.0  100
          1714 ORA208 ORACLE2  0508     0    0     8     0    0     0  100  0.0  100
ORACLE4   1914 ORA401 ORACLE4  0501     0 9830 2722K 2717K  100   841  0.0  6.8  100
          1A14 ORA402 ORACLE4  0502     0 9093 2325K 2291K 98.6   860  1.5  5.8  100
          1814 ORA400 ORACLE4  0500     0 9830  3363  3241 96.4  2721  0.1  0.0  100
ORACLE5   1115 ORA501 ORACLE5  0501     0 9284  2903  2066 71.2  2441    0  0.0  100
          1315 ORA503 ORACLE5  0503     6 4291 1116K 1080K 96.8  1048  5.1  2.8  100
          1415 ORA504 ORACLE5  0504     6 4291 1073K 1040K 96.9  1067  5.1  2.7  100
          1515 ORA505 ORACLE5  0505     6 4291 1097K 1063K 96.9  1049  5.1  2.7  100
          1615 ORA506 ORACLE5  0506     6 4287 1085K 1049K 96.7  1077  4.9  2.7  100
          1715 ORA507 ORACLE5  0507     6 4291 1099K 1063K 96.7  1059  5.0  2.8  100
          1815 ORA508 ORACLE5  0508     6 4287 1093K 1055K 96.5  1072  5.0  2.7  100
          1015 ORA500 ORACLE5  0500     0 9830  2497  2396 96.0  2435    0  0.0  100
```

Figure 8-11 ESAUSEK - User DASD seeks report

In following up on the LVM analysis, one of the impacts of the different configuration in this case showed up as I/O Wait. The three servers during the test were thought to be identical, except on analysis, the server with the balanced I/O was consistently waiting for I/O more than half the time and using much less CPU. The ESAXACT report shows a sampled wait state. ESAMON samples each server at a given rate and counts the number of times the server is running, waiting for a resource, or in several other categories requiring more of an explanation than needed here. The class of users is the total of all of the ORACLE Linux servers, of which the three active servers are reported as top users on the system.

```
Report: ESAXACT Transaction Delay AnalysisMOP Oracle Tests   ESAMAP 3.4.1 07/28/04  Page 228
Monitor initialized:      on 2084 serial 1F86C     First record analyzed: 07/28/04 10:00:00
-----------------------------------------------------------------------------------------------
                   <---------------Percent non-dormant---------------------->         Times
UserID   <-Samples->                    E-  D-  T-     Tst <Asynch>       Lim  Pct     I/O
/Class  TotalIn Q Run Sim CPU SIO Pag SVM SVM SVM  CF Idl I/O Pag Ldg Oth Lst Elig
Throttled
------- ---- ---- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ----
---------

10:48:00   42   11  18 9.1   0   0   0   0   0  18   0  36  18      .   0   0   0       .
Hi-Freq: 2520  728  15 0.3 0.1   0   0   0  32   0   0  30  23   0   0   0   0   0       0
 ***User Class Analysis***
ORACLE    540  329  33 0.6 0.3   0   0   0   0   0   0  45  21   0   0   0   0   0       0
 ***Top User Analysis***
ORACLE2    60   60  67   0 1.7   0   0   0   0   0   0   0  32   0   0   0   0   0       0
ORACLE4    60   60  77 1.7   0   0   0   0   0   0   0   0  22   0   0   0   0   0       0
ORACLE5    60   60  38 1.7   0   0   0   0   0   0   0   0  60   0   0   0   0   0       0
-----------------------------------------------------------------------------------------------
10:49:00   42    8  25   0   0   0   0   0   0   0   0  38  38      .   0   0   0       .
Hi-Freq: 2520  703  16   0 0.4   0   0   0  29 1.1   0  29  24   0   0   0   0   0       0
 ***User Class Analysis***
ORACLE    540  316  35   0 0.9   0   0   0   0   0   0  43  21   0   0   0   0   0       0
 ***Top User Analysis***
ORACLE2    60   60  73   0   0   0   0   0   0   0   0   0  27   0   0   0   0   0       0
ORACLE4    60   60  67   0 3.3   0   0   0   0   0   0   0  30   0   0   0   0   0       0
ORACLE5    60   60  45   0 1.7   0   0   0   0   0   0   0  53   0   0   0   0   0       0
-----------------------------------------------------------------------------------------------
```

Figure 8-12 ESAXACT transaction delay analysis

Further analysis on the ESAUSR3 report is from the server perspective. This report shows the actual number of I/O performed in one interval and seems to make LVM look better. The ORACLE5 server is actually doing twice the I/O. The question is why this server from the workload perspective was doing less work. There is still some work to be done on understanding the impact of LVM.

Archived

```
Report: ESAUSR3 User Resource Utilization - Part 2  MOP Oracle Tests  ESAMAP 3.4.1 07/28/04
Page 396
Monitor initialized:                      on 2084 serial 1F86C     First record analyze
-------------------------------------------------------------------------------------------
                 DASD MDisk Virt Cache I/O    <---Virtual Device---->
UserID    DASD Block Cache Disk   Hit Prty   <----I/O Requests----->  Diag <Transfers>
/Class     I/O   I/O  Hits  I/O   Pct Queued  Cons   U/R  CTCA Other    98  IUCV  VMCF
-------- ----- ----- ----- ---- ------ ----- ----- ----- ----- ----- ----- ----- ---- -
10:48:00  396K     0  154K    0  38.9      0    16     0     0     0     0  7359
 ***User Class Analysis***
ORACLE    395K     0  154K    0  38.9      0     0     0     0     0     0     0
 ***Top User Analysis***      - ------ ----- ----- ----- ----- ----- ----- -----
ORACLE2   103K     0 17306    0  16.8      0     0     0     0     0     0     0
ORACLE4  80156     0 12913    0  16.1      0     0     0     0     0     0     0
ORACLE5   212K     0  123K    0  58.2      0     0     0     0     0     0     0
-------------------------------------------------------------------------------------------
10:49:00  422K     0  171K    0  40.5      0     2     0     0     0     0  7266
 ***User Class Analysis***
ORACLE    422K     0  171K    0  40.5      0     0     0     0     0     0     0
 ***Top User Analysis***      0  16.3      0     0     0     0     0     0     0
ORACLE2   102K     0 17455    0  17.1      0     0     0     0     0     0     0
ORACLE4  80476     0 12808    0  15.9      0     0     0     0     0     0     0
ORACLE5   239K     0  141K    0  58.9      0     0     0     0     0     0     0
```

Figure 8-13 ESAUSR3 User Resource Utilization

8.5 Processor analysis

During our runs we suspected that there was a problem with excess CPU being consumed by spin locks in the Linux kernel. This section shows how we measured the DIAG 44 instructions. DIAG 44 is a Voluntary Time Slice End, and is used by the Linux kernel when the alternative to giving up control of the logical CP is to spin waiting for a lock to be released.

In order to trace DIAG 44 instructions we used the CP commands in Figure 8-14.

```
#cp spool e cmsuser
#cp trace diag 44 printer run
#cp trace end
#cp close e
```

Figure 8-14 CP commands to get a trace of diagnose 44 instructions

Archived

```
+00:  -> 0000000000113A32'  EX    44000214
 00:         0000000000000214'  DIAG  83000044    0000000000000044     C
 00:  -> 0000000000113A32'  EX    44000214
 00:         0000000000000220'  DIAG  83000044    0000000000000044     C
 00:  -> 0000000000113A32'  EX    44000214
 00:         0000000000000224'  DIAG  83000044    0000000000000044     C
 00:  -> 0000000000113A32'  EX    44000214
 00:         0000000000000244'  DIAG  83000044    0000000000000044     C
 00:  -> 0000000000113A32'  EX    44000214
 00:         0000000000000264'  DIAG  83000044    0000000000000044     C
 00:  -> 0000000000113A32'  EX    44000214
 00:         0000000000000284'  DIAG  83000044    0000000000000044     C
 00:  -> 0000000000113A32'  EX    44000214
 00:         0000000000000204'  DIAG  83000044    0000000000000044     C
 00:  -> 0000000000113A32'  EX    44000214
 00:         0000000000000214'  DIAG  83000044    0000000000000044     C
 00:  -> 0000000000113A32'  EX    44000214
 00:         0000000000000214'  DIAG  83000044    0000000000000044     C
 00:  -> 0000000000113A32'  EX    44000214
 00:         0000000000000214'  DIAG  83000044    0000000000000044     C ...
```

Figure 8-15 Sample CP trace of diagnose code 44

This trace listing file will be in the virtual reader of whatever user ID was used on the **spool** command in Figure 8-16. The file must be read and placed in the CMS file system. The listing file can then be reduced by the REXX code in Figure 8-16. This creates a summary file with the file type SUMM.

```
 /* REXX */
trace 'O'
parse arg fn ft fm
address command
'PIPE < ' fn ft fm              ,
'| specs 10-25 1 read         ' ,
'| sort count                 ' ,
'| specs 1-10 1 11-* nextword ' ,
'| sort  1-10 desc            ' ,
'| > ' fn ' SUMM ' fm
```

Figure 8-16 Rexx exec to reduce trace

Archived

```
1725 00000000000CEC76
1444 0000000000113A32
1156 0000000000031E04
 516 0000000100A3CC6A
 168 0000000000101444
  77 00000000000682B4
  54 0000000000058682
  52 0000000100A3D8E4
  44 0000000000097C22
  38 00000000000730CE
  38 0000000100A3DD9A
  36 0000000000084740
  31 0000000000058848
  29 000000000001DCF6
  29 0000000000067E08
  28 000000000014012C
  27 0000000000059112
  27 0000000100A3D9B6
  26 0000000000072F1A
  26 00000000000843E0
  26 00000000001CA88E
  23 000000000014E202
  23 0000000100A40FDC
  21 00000000000301D4 ...
```

Figure 8-17 Reduced trace of diagnose code 44

This reduced trace shows the count and the corresponding address of the DIAG 44. Note the high number of DIAG 44 at the top three locations. Looking at the fragment of the system map in Figure 8-18 on page 193 you can see that the highest frequency DIAG 44 was executed in routine **`nfs_commit_write`**. Similarly, we discovered that the next two routines are **`do_IRQ`** and **`do_schedule`**.

By examining the source code for these routines we found that each was a request for the *Big Kernel Lock*. This is a spin lock used by several Linux kernel functions. Removing these locks would require source changes in the open source code. In fact, on reviewing these results, the IBM team in Boeblingen did follow up on the lock contention issue that showed up in **`do_IRQ`**. The routine was called by **`do_adapter_IO`**. They were able to redesign this routine to remove its dependency on the Big Kernel Lock. This change is now part of the 2.6 kernel.

Archived

```
0000000000000000 A _text
0000000000000298 t iplstart
0000000000000800 T start
0000000000010000 t startup
0000000000010400 T _pstart
0000000000011000 T _pend
0000000000011000 T _stext
0000000000011090 t rest_init
0000000000011104 t init
00000000000112c8 T get_root_device_name
00000000000112f4 t do_linuxrc
....
00000000000ce6c8 t nfs_file_flush
00000000000ce790 t nfs_kvec_write
00000000000ce860 t nfs_kvec_read
00000000000ce930 t nfs_file_read
00000000000cea38 t nfs_file_mmap
00000000000ceb20 t nfs_fsync
00000000000cec14 t nfs_prepare_write
00000000000cec38 t nfs_commit_write
00000000000ceccc t nfs_sync_page
00000000000ced70 t nfs_file_write
00000000000cee80 T nfs_lock
00000000000cf19c t nfs_reqlist_init
00000000000cf300 T nfs_reqlist_exit
....
00000000004d8468 B errno
00000000004d8470 B ___strtok
00000000004d8478 A _end
```

Figure 8-18 Fragment of Linux system map

8.6 LPAR weights and options

There are two current trends:

- One is to consolidate multiple slower processors to much faster z/900s.
- The other is to separate workloads via the use of LPAR mode to avoid issues with the 2 GB line.

This section should clarify some of the configuration options.

Figure 8-19 on page 194 and Figure 8-20 on page 195 show an LPAR report. As z/VM often operates in a multiple LPAR environments, understanding LPAR

Archived

options and their impacts on performance will help configure your systems to meet your business requirements.

```
Report: ESALPAR  Logical Partition Analysis      ITSO Residency ESAMAP 3.3.0
Monitor initialized: on 2064 serial C0ECB First record analyzed: 01/30/03
-------------------------------------------------------------------------
         <--Complex--> <--Logical-> <--------Logical Processor---------->
         Phys Dispatch <-Partition> VCPU <%Assigned>         Cap-  Wait
Time     CPUs    Slice Name     No. Addr Total  Ovhd  Weight  ped   Comp
-------- ---- -------- -------- --- ---- -----  ----  ------  ----  ----
18:00:00   13  Dynamic  A12      12    0  22.3   0.4     10    No    No
                                      1  35.7   0.4     10    No    No
                                          ----  ----
                                   LPAR  58.0   0.8
                         A1         1    0   5.3   0.5    180    No    No
                                      1   5.3   0.5    180    No    No
                                         -----  ----
                                   LPAR  10.6   0.9
                        A2          2    0   5.2   0.5     10    No    No
                                      1   5.2   0.5     10    No    No
                                         -----  ----
                                   LPAR  10.5   0.9
                        A3          3    0   5.3   0.5    180    No    No
                                      1   5.3   0.5    180    No    No
                                         -----  ----
                                   LPAR  10.6   0.9
                        A4          4    0   1.5   0.2     10    No    No
                                      1   0.3   0.0     10    No    No
                                         -----  ----
                                   LPAR   1.8   0.2
                        A5          5    0   3.2   0.3     10    No    No
                                      1   3.0   0.3     10    No    No
                                      2   2.9   0.3     10    No    No
                                      3   2.8   0.3     10    No    No
                                         -----  ----
                                   LPAR  11.8   1.2
```

Figure 8-19 ESALPAR Logical Partition Analysis Part 1

Archived

```
A6          6    0   5.1   0.5     10    No    No
                 1   5.2   0.5     10    No    No
                    -----  ----
            LPAR  10.3   0.9
            7    0   9.2   0.5     10    No    No
                 1   9.0   0.5     10    No    No
                    -----  ----
            LPAR  18.3   1.0
A8          8    0   4.6   0.5     10    No    No
                 1   4.8   0.5     10    No    No
                    -----  ----
            LPAR   9.4   0.9
A9          9    0   4.5   0.5     10    No    No
                 1   4.6   0.5     10    No    No
                    -----  ----
            LPAR   9.1   0.9
A10        10    0   4.5   0.5     10    No    No
                 1   4.5   0.5     10    No    No
                    -----  ----
            LPAR   9.0   1.0
A11        11    0   5.7   0.5    180    No    No
                 1   5.7   0.5    180    No    No
                    -----  ----
            LPAR  11.4   0.9
C1         13    0 100.0   0.0    Ded    No   Yes
                 1 100.0   0.0    Ded    No   Yes
                    -----  ----
            LPAR 200.0   0.0
C2         14    0 100.0   0.0    Ded    No   Yes
                 1 100.0   0.0    Ded    No   Yes
                    -----  ----
            LPAR 200.0   0.0
C3         15    0 100.0   0.0    Ded    No   Yes
                 1 100.0   0.0    Ded    No   Yes
                    -----  ----
            LPAR 200.0   0.1

System total logical partition busy:     770.7  10.8
```

Figure 8-20 ESALPAR Logical Partition Analysis Part 2

In Figure 8-19 on page 194, the z/VM Logical Partition (A12) is listed first in the report. The report also shows what z/VM considers its processor utilization to be. Assigned time is the time that a physical processor is assigned to a logical one. Logical Partitions are prioritized by weights, which are explained in 8.6.2, “Converting weights to logical processor speed” on page 197.

Archived

The processor that is shown first is the z/VM LPAR in which the experiments were run. ESAMAP always shows the VM LPAR first. All of the assigned times and utilizations shown on this report (and the rest of both ESAMAP and ESAMON) are in absolute numbers, meaning these are percents of one processor. Thus, there is never a “percent of a percent” being reported where one of the percents is not provided. Thus, the VM LPAR is shown as 58 percent out of 200 percent.

Other values reported on this report discussed further in this section are *wait completion*, set at *NO*, and *capped*, also set at *NO*.

There are two forms of overhead reported: Logical and physical. Logical overhead can be charged to the Logical Partition, whereas physical overhead is not. There is a correlation between the number of logical processors defined and the amount of physical overhead involved in time slicing the physical processor between the logical processors: The more logical processors, the higher the overhead.

This example should be almost a worst case example. In this example, the logical overhead was 10.8 percent of one processor, but the physical overhead, as shown in the next section, was 39.9 percent.

8.6.1 Physical LPAR overhead

The overhead of managing the physical processors from Figure 8-19 on page 194 and Figure 8-20 on page 195 is shown in Figure 8-21 on page 197.

Archived

```
Physical CPU Management time:
       CPU   Percent
       ---   -------
         0     6.815
         1     4.968
         2     5.000
         3     6.735
         4     4.874
         5     6.513
         6     4.920
         9     0.007
        10     0.007
        11     0.007
        12     0.008
        13     0.007
        14     0.007
                ------
    Total:     39.870
```

Figure 8-21 LPAR physical CPU management time

From the report, we see there are seven physical processors (0–6). These are shared and have an overhead in the 5–6 percent range. Dedicated processors have less overhead. This should not mean that each installation dedicates processors to Logical Partitions to reduce overhead. In this case, the high overhead is caused by a large number of logical processors competing for time on the physical processors in order to be dispatched.

Note: To reduce physical overhead, use fewer Logical Partitions, and fewer logical processors.

8.6.2 Converting weights to logical processor speed

An LPAR is granted control of processors based on time slices. Each LPAR gets time slices based on the weighting factor for the LPAR. To determine the weight of each logical processor, use the following calculations:

1. Add up all the weights of the logical processors.

   In Figure 8-19 on page 194 and Figure 8-20 on page 195 there are 12 LPARs sharing seven logical processors based on weighting. Of the 12, three have a weight of 180 (LPARs A1, A3, and A11). The remainder have a weight of 10. Therefore, the total weight is 630.

2. Divide the weight of the *interesting LPAR* into the total.

Archived

This is the *logical share* of the physical complex allocated to the *interesting LPAR*. The VM Logical Partition (LPAR A12) has a weight of 10. Dividing this by the total weight (630) yields a 1.6 percent share of seven shared processors.

3. Divide the number of logical processors of the LPAR into the logical share.

   This is the percentage of each logical processor that is directly relative to the maximum speed that a logical processor will operate if capped. Thus, 1.6 percent of seven processors is equivalent to about 10 percent of one processor (1.6 * 7 = 11.2). Divide this into the two logical processors used by our VM system in LPAR A12, and each processor would be targeted to get 5 percent of one processor.

If all Logical Partitions are busy, this percentage would be the amount of processing that would be performed. In our case very few of the Logical Partitions were very busy, and so the A12 Logical Partition could get as much as 90 percent of each logical engine in its assigned time.

8.6.3 LPAR analysis example

As an example, if the weight of a Logical Partition is 10, and the total weight of all the Logical Partitions is 1000, then the Logical Partition is targeted at 1 percent of the system. If the system is comprised of 10 processors, the Logical Partition is expected to get an amount of processing equivalent to one physical processor. If the LPAR has two logical processors, then each logical processor is allocated 5 percent of a physical processor. Thus, increasing the number of logical processors in a complex will decrease the relative speed of each logical processor in a Logical Partition.

8.6.4 LPAR options

Logical Partitions may be defined as *capped*, meaning that their share of the physical system is capped to their given weight. Given the situation of 1 percent allocated, this would be a very small amount of resource. If not capped, unused CPU resources are available to any Logical Partition that can utilize the resource based on given weights. Capped Logical Partitions should rarely be used except in installations where a financial agreement exists to provide a specific speed of processor.

The time slice is either specific or *dynamic*. Specifying a specific time slice value for LPAR is rarely if ever done. We recommend that you specify that LPAR use a *dynamically determined* value.

Wait completion yes or no is an LPAR parameter that defines whether a logical CP gives up control of the real CP when it enters a wait state. Wait completion

specified as yes will keep the real processor assigned for the complete time slice, even if the processor is in the wait state. Wait completion specified as no means that the physical CP will be released when the logical CP enters a wait state, giving up the remainder of the time slice. We recommend that wait completion be specified as no, meaning that when work is completed, the processor is given up voluntarily so that other Logical Partitions may use the physical processor.

8.6.5 Shared versus dedicated processors

When there are multiple physical processors on a system to be utilized by many different logical partitions, there is the option of dedicating processors to a Logical Partition. In general, this is only used for two reasons:

- For benchmarks to eliminate questionable impacts from other workloads.
- When the workload is steady enough to utilize the processors and sufficient in requirements to justify dedicated processors.

To justify the cost of zSeries implementations, the objective should always be high utilization. High utilization leverages the value of the reliability, availability, and serviceability of the zSeries systems, and the cost of the zSeries. Other platforms rarely operate at high utilizations. Dedicating physical resources such as processors to one Logical Partition has the potential for reducing the overall system utilization. Reducing the system distillation reduces the effectiveness of the platform and increases the cost per unit of work for that system.

Archived

Archived

9

Using Radius Server and z/OS RACF LDAP for Oracle DB user authentication

In this chapter, we describe the steps we followed to use a Radius Server on Linux on zSeries to connect to LDAP and RACF® on z/OS. This allowed external Oracle users to be authenticated when connecting to an Oracle 10g Database on Linux on zSeries.

Archived

© Copyright IBM Corp. 2005. All rights reserved.

9.1 Overview

The Oracle DB user password authentication method discussed in this chapter is not restricted to a particular configuration; however, this particular method may provide utility in a z/VM environment running multiple Linux systems where each Linux is running an Oracle DB server. The usefulness of this method is enhanced further if the z/OS system running the centralized Lightweight Directory Access Protocol (LDAP) server is on this same zSeries as the z/VM.

An Oracle 10g Database server was configured to do external password authentication with the Oracle Advanced Security Option (ASO) using a RADIUS server on Linux. The RADIUS server in turn was configured to use LDAP as its database. The LDAP database was on z/OS RACF.

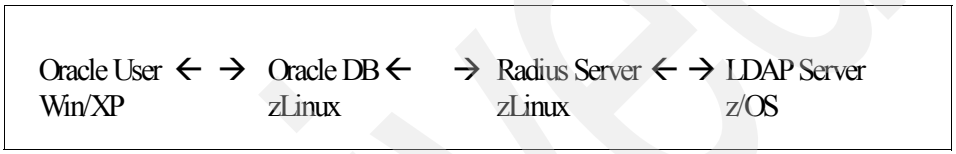


Figure 9-1 Order of processing

Oracle ASO is a RADIUS client to the RADIUS server and the RADIUS server is an LDAP client to the z/OS LDAP Server. Many of the details on the configuration of FreeRADIUS, z/OS LDAP, the Oracle DB, and the Oracle client to use z/OS LDAP as the repository for Oracle DB user passwords are discussed in the following sections.

9.2 FreeRADIUS on Linux on z/OS

RADIUS is a client/server security protocol that can be used by the Oracle Advanced Security Option (ASO) to do external password authentication of Oracle users. There are a number of RADIUS servers available and many of them can be found by searching the Internet. The FreeRADIUS server available from `http://www.freeradius.org` was installed and configured on an SLES9 Linux system. z/OS LDAP was selected as the FreeRADIUS database backend, but it also supports MySQL, PostgreSQL, and Oracle databases.

The Linux SLES9 CDs contain the freeradius-0.9.3-106.6 package. This package version of FreeRADIUS did not work with Oracle 10g. Specifically, the encrypted password from Oracle was never decrypted correctly by the freeradius-0.9.3-106.6 level server. A decision was made to upgrade to the latest version of FreeRADIUS. The latest FreeRadius source, freeradius-1.0.4.tar.gz,

was downloaded. Binaries are not available from `http://www.freeradius.org`, so the FreeRADIUS server was built from the source code.

1. FreeRadius installation

   Several problems were encountered during the install of FreeRadius-1.0.4 from the downloaded source files. Most of the problems were resolved by the installation of the following packages: gdbm-devel-1.8.3-228.1, openssl-devel-0.9.7d-15.10, unixODBC-devel-2.2.8-58.4, mysql-devel-4.0.18-32.1, openldap2-devel-2.2.6-37.19, pam-devel-0.77-221.1, cyrus-sasl-devel-2.1.18-33.1, heimdal-devel-0.6.1rc3-55.3, e2fsprogs-devel-1.34-115.1, and db-devel-4.2.52-86.3. These packages contained C header files that are included by various source files of FreeRADIUS. Another problem involving compilation failures of Kerberos and SQL functions was avoided by removing these functions for the install, as they were not required when LDAP is used as the backend database. These functions were removed (not included) by specifying the options --without-rlm_krb5 and --without-rlm_sql on the **`configure`** command. The defaults for the target libraries were used and the install put the FreeRADIUS executables in /usr/local/bin and /usr/local/sbin, the configuration files in the /usr/local/etc/raddb directory and the log files in the /usr/local/var/log/radius directory.

2. FreeRadius configuration

   FreeRADIUS was configured to operate as a RADIUS server for Oracle Advanced Security and as an LDAP client to z/OS LDAP. Three FreeRADIUS configuration files, radisud.conf, clients.conf, and users, had to be modified. The details below show the specific pieces of each file that were changed from the installed default values. The radiusd.conf file was changed to allow the use of LDAP as the database for FreeRADIUS. The z/OS LDAP server was identified by network name, mvs09.us.oracle.com; the port was defaulted to 389; the ID and password of the user that could bind to LDAP and search for users seeking authentication were specified; the filter for the LDAP search was specified; and finally metadata for the Oracle user seeking z/OS LDAP authentication was specified. The actual values specified for the *identity* and *basedn* parameters are further discussed in the z/OS LDAP section. These two values are required to make the connection between the FreeRADIUS server and z/OS LDAP. The clients.conf file was changed to identify the network name of the Oracle DB server that would be connecting to FreeRADIUS and the default port of 1812 for the connection was used. The Radius Server *secret* is also specified in this file. This is the key that the Oracle DB as a RADIUS client and the FreeRADIUS server uses to encrypt and decrypt the password. The third file, users, identifies the Oracle client user, TERRY1, who will authenticate using LDAP.

Attention: Most of these files are very large. We have just included the lines that we needed to change.

3. radiusd.conf file

   We created the radiusd.conf file in /etc/local/etc/raddb.

```
ldap {
server = "mvs09.us.oracle.com"
identity = "racfid=TERRY2,profiletype=user,cn=ORARACF,o=IBM,c=US"
password = test123
basedn =
   "racfid=%{Stripped-User-Name:-%{User-name}},
   profiletype=user,cn=ORARACF,o=IBM,c=US"
   filter = "(objectclass=*)"
   }

Auth-Type LDAP {
ldap
}
```

4. Clients.conf file

   We created the clients.conf file in /usr/local/etc/raddb.

Example 9-1 clients.conf file

```
client pazxxt03.us.oracle.com {
        secret          = TESTING12345678
        shortname       = pazxxt03
```

5. Users file

   We created the users file in /usr/local/etc/raddb.

Example 9-2 users file

```
TERRY1 Auth-Type = LDAP
```

6. FreeRadius debug mode

   FreeRADIUS can be started in debug mode by supplying the -X option, for example, /usr/local/sbin/radiusd -X. This provides a lot of information and is particularly helpful when the conf files above are changed, as much of the information from the files is printed when FreeRADIUS is started. Additionally, in this mode the information about the connections to both the Oracle DB and z/OS LDAP is printed.

7. FreeRadius test driver

Archived

FreeRADIUS provides a test driver, radtest, that is useful in testing that FreeRADIUS is installed and configured correctly, for example, executing /usr/local/bin/radtest TERRY1 test123 127.0.0.1:1812 0 testing123 would test that TERRY1 is in the users file; would test the encrypt and decrypt of the password test123 with the secret of testing123; and FreeRADIUS would send the user ID/password, TERRY1/test123, to z/OS LDAP. The local loop has a different *secret* in the clients.conf file than the Oracle database. It is a coincidence that the test123 password here matches the password in the radiusd.conf file above for TERRY2.

8. References

   There is much general Radius Server information available on the Internet, but the readme files (and referenced files) and comments in the configuration files (particularly radisud.conf) seemed to be the best source of detailed documentation.

9.3 z/OS LDAP

The z/OS LDAP server, part of the Integrated Security Services for z/OS V1R6.0, is based on a client/server model that provides client access to an LDAP server. An LDAP directory provides a way to maintain directory information in a central location. The LDAP server provides access to RACF user information stored in the SDBM database. RACF group information is also available, but is not used here. SDBM is also known as the RACF database backend of the LDAP server. The SDBM database allows for directory authentication using the RACF user ID and password. The RACF user ID must have an OMVS segment defined. The RACF Subsystem function of RACF must be defined and activated to allow the LDAP server to communicate with RACF through the SDBM backend and the LDAP server must be configured and started.

The LDAP data model is closely aligned with the X.500 data model. In this model, a directory service provides a hierarchically organized set of entries. Each of these entries is represented by an object class. The object class of the entry determines the set of attributes that are required to be present in the entry as well as the set of attributes that can optionally appear in the entry. An attribute is represented by an attribute type and one or more attribute values. In addition to the attribute type and values, each attribute has an associated syntax that describes the format of the attribute values. Every entry in the directory has a distinguished name (DN). The DN is the name that uniquely identifies an entry in the directory. A DN is made up of attribute=value pairs, separated by commas, for example, cn=Ben Gray,ou=editing,o=New York Times,c=US is an LDAP DN. The format of the DN is restricted when using SDBM, as the DN must match the schema of the underlying RACF data. A RACF SDBM style DN for a user contains two required attributes plus a valid DN suffix: racfid=user

ID,profiletype=user,suffix. The required attributes are racfid and profiletype. For the work here the attribute value of the profiletype attribute must be user. In other contexts the attribute value of group is also available for attribute profiletype when using the SDBM backend. The DN suffix used here is cn=ORARACF,o=IBM,c=US.

The SDBM database of the LDAP server implements portions of the **adduser**, **addgroup**, **altuser**, **altgroup**, **deluser**, **delgroup**, **listuser**, **listgrp**, **connect**, **remove**, and **search** RACF commands. A user has the same authority through SDBM as with normal RACF commands. In an actual implementation many of these commands would be needed and used either through LDAP or directly through RACF, for example, adding/deleting users, changing/resetting passwords. See Chapter 16, "Accessing RACF Information," in *z/OS V1R6.0 Integrated Security Services LDAP Server Administration and Use,* SC24-5923-06, on using the LDAP functions not covered here.

If the Linux system running FreeRadius is not on the same zSeries machine as the z/OS LDAP server, the communication can be encrypted using Secure Socket Layer (SSL). This function is not covered here but it would avoid the possible issues associated with sending a clear text password across a network.

1. Configuration

   LDAP server runtime configuration is accomplished through its configuration file, slapd.conf. For the first-time use of the LDAP server, this file needs to be copied: "cp /usr/lpp/ldap/etc/slapd.conf /etc/ldap/slapd.conf", and modified. Optionally, a second file that is used for tracing and debugging can be copied: "cp /usr/lpp/ldap/etc/slapd.envvars /etc/ldap/slapd.envvars", and modified as required.

2. RACF user IDs and authorizations

   There are at least three different levels of RACF authorizations that are used when using a z/OS LDAP server with the SDBM backend. The first is the authority of the user ID whose password is to be authenticated when the ID attempts to connect to the Oracle DB. The examples here use the first user ID of TERRY1. This ID has a password (test123) and an OMVS segment defined in RACF. The second is the authority of the user ID that is able to do the initial connection from FreeRADIUS to z/OS LDAP. In addition to an OMVS segment and password, this ID needs RACF authority to do a search of the SDBM database for the first user ID. In RACF terms this ID needs authority to do a **listuser** command with the other ID as an argument. The examples here use the second user ID of TERRY2. The third is the RACF authority needed by the user ID that starts the LDAP server on z/OS. The LDAP server is started from the USS shell and the user ID is TERRY3. This ID needs access to TSO and OMVS. The ID needs read access to the BPX.DAEMON facility class and update access to the BPX.SERVER facility

class, if these two facility classes are defined to RACF. The IRR.RUSERMAP RACF facility class must be defined, and this ID needs read access to it.

The connection and authentication protocol between the LDAP client (FreeRADIUS) and the LDAP server (RACF with SDBM) is to first bind with ID TERRY2 using its DN that is explicitly specified in the FreeRADIUS radiusd.conf file parameter identity. If the bind succeeds, a search is done to find the DN of the user ID, TERRY1, to be authenticated for access to the Oracle DB. If the search succeeds, a bind is done for the TERRY1 ID, and success here equals authentication. The point here is this seems like duplication of effort and it is in this case. Since the DNs for all users will be the same, it should be possible to modify FreeRADIUS to use the fixed DN and do the bind with the TERRY1 ID for authentication, and the TERRY2 ID could be eliminated from the scenario.

```
PERMIT BPX.DAEMON CLASS(FACILITY) ID(TERRY3) ACCESS(READ)

PERMIT BPX.SERVER CLASS(FACILITY) ID(TERRY3) ACCESS(UPDATE)

RDEFINE FACILITY IRR.RUSERMAP UACC(NONE)

PERMIT IRR.RUSERMAP CLASS(FACILITY) ID(TERRY3) ACCESS(READ)

SETROPTS RACLIST(FACILITY) REFRESH
```

3. slapd.conf file in /usr/lpp/ldap/etc

   The SDBM database must be enabled for use and the adminDN and suffix parameters must be aligned with the identity and basedn parameters in the FreeRADIUS radisud.conf file.

```
database sdbm GLDBSDBM

adminDN "racfid=TERRY2,profiletype=user,cn=ORARACF,o=IBM,c=US"

suffix "cn=ORARACF,o=IBM,c=US" .
```

4. Starting the LDAP Server

   A sample shell script is located in /usr/lpp/ldap/sbin/slapd. The LDAP server is started from the USS shell with:

```
/usr/lpp/ldap/sbin/slapd  & .
```

   To stop the LDAP server in the z/OS shell, it is necessary to know its process ID. Enter:

```
ps -ef | grep slapd
```

   This will provide the process ID for the LDAP server, which can be used in **`kill -15 process-ID`** to shut down the LDAP server.

5. Debug and trace

The debugging facility can be turned on and off dynamically using the MVS™ **modify** command. Here the ID TERRY3 was used to start the LDAP server from the USS shell, so the jobname for use with the **modify** command can be determined from either an SDSF PS command or an MVS **display system activity** command. The USS jobname assigned by USS will be of the form TERRY3*x*, where *x* is a digit between 1 and 9. The **modify** command entered is:

```
f TERRY3x,appl=debug=debug_level
```

The debug_level is a mask that specifies the desired debug level and can be found in the references. Similarly, tracing and the use of the parameters in the slapd.envvars file can be found in the references.

6. References
   - *z/OS V1R6.0 Integrated Security Services LDAP Server Administration and Use,* SA24-5923-06
   - *z/OS V1R6.0 Security Server RACF Security Administrator's Guide,* SA24-7683-06

9.4 Oracle DB Advanced Security Option (ASO)

These are the steps to set up ASO:

1. Install ASO.

   This option can be installed by doing a custom install and it must be installed for the Oracle DB to do external authentication.

2. Verify Radius Adapters available.

   Oracle Advanced Security supports remote authentication through adapters. After the option is installed, the RADIUS adapters can be verified by running the **$ORACLE_HOME/bin/adapters** command and also running the command with an option $ORACLE_HOME/bin/adapters ./oracle. The first command will list the installed Oracle Advanced Security options and will include RADIUS authentication. Similarly, the second command will show RADIUS authentication is linked.

3. Create Radius secret file.

   This must exactly match the secret specified in the FreeRADIUS clients.conf file for the database connection defined above.

```
radius.key file in $ORACLE_HOME/network/security

   TESTING12345678
```

4. Configure Oracle for external authentication with FreeRADIUS.

   The Oracle Net Manager is used to configure the Oracle DB to use RADIUS for authentication, as shown in Figure 9-2 through Figure 9-4 on page 210. From the local profile window select **Oracle Advanced Security** from the pull-down menu and then select the **Authentication** tab and move RADIUS from the available to the selected methods window. After this is done, from the other parms tab, select **RADIUS** in the Authentication Service window. Fill in the host name, port name, and the secret file location with values. The results are in the sqlnet.ora file, and shown below.

```
sqlnet.ora in $ORACLE_HOME/network/admin
(...keywords changed were)
SQLNET.AUTHENTICATION_SERVICES = RADIUS
SQLNET.RADIUS_AUTHENTICATION = pazxxt03.us.oracle.com
SQLNET.RADIUS_AUTHENTICATION_PORT = 1812
SQLNET.RADIUS_SECRET = /oracle/10g/network/security/radius.key.
```

   Also, the two instance parameters, os_authent_prefix and remote_os_authent, need to be set so that the Oracle DB server will do external authentication. If using a server parameter file (/$ORACLE_HOME/dbs/spfileora1.ora), they can be set with the following **`alter system`** commands:

```
SQL> alter system set os_authent_prefix = " " scope = both;
SQL> alter system set remote_os_authent = false scope = both;
```

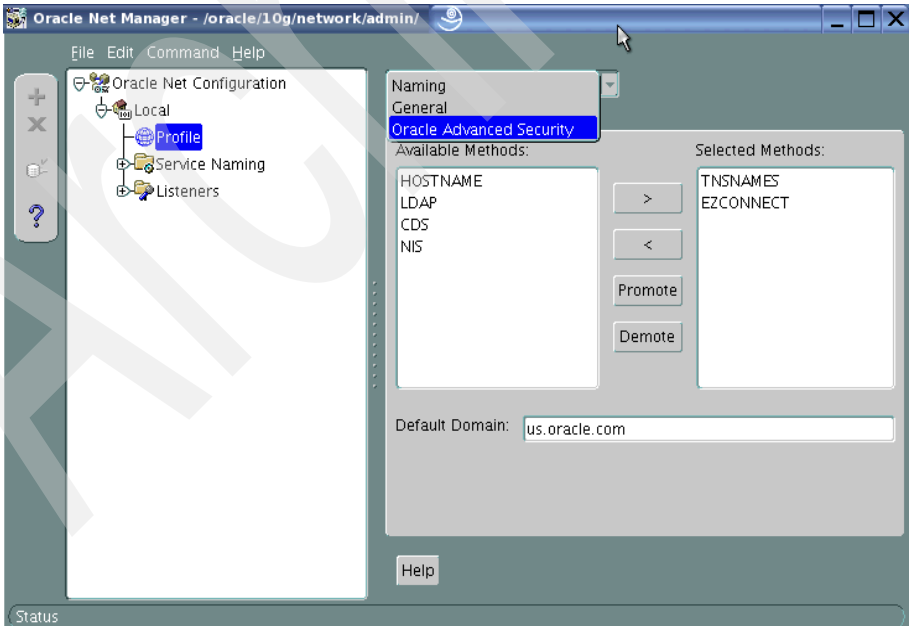


Figure 9-2 Oracle Advanced Security pull-down

Archived

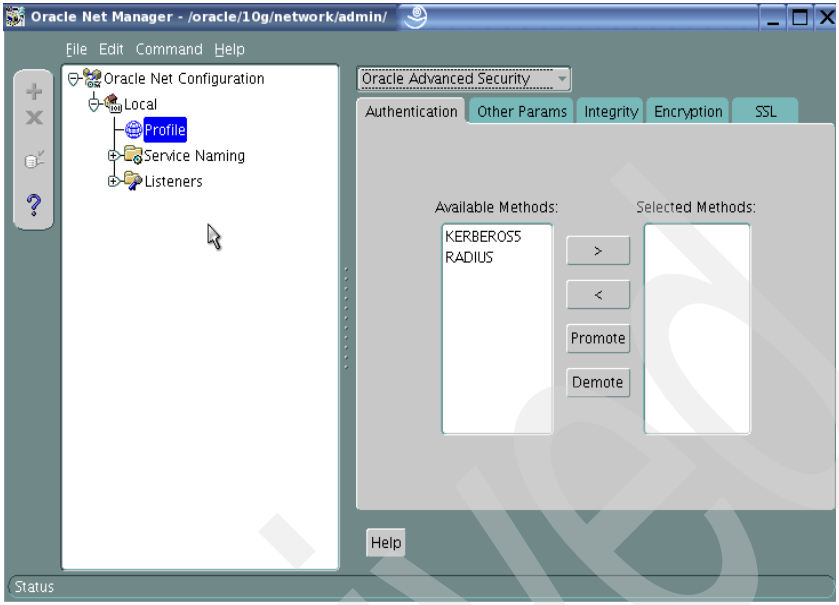


Figure 9-3 Authentication

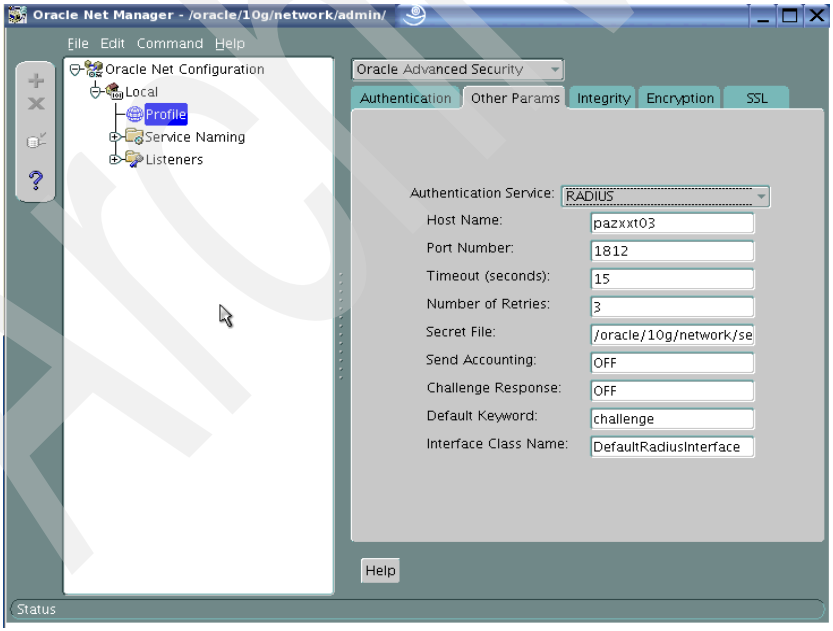


Figure 9-4 Oracle Advanced Security Other Parameters tab

Archived

5. Create users with identified external attributes.

```
create user TERRY1 identified externally;
grant connect, resource to TERRY1;
```

6. References
   - *Oracle. Database Advanced Security Administrator's Guide 10g Release 1 (10.1)* Part No. B10772-01
   - *Oracle Note: 272804.1 Installing and Configuring RADIUS and Oracle 9.2.0 Advanced Security Option (ASO) on Linux*

9.5 Oracle client

These are the steps to set up the Oracle client:

1. Install the Client.

   The Oracle client install on WIN/XP requires the selection of the Advanced Security Option to be able to do RADIUS authentication from an Oracle client.

2. Configure.

   The Oracle Net Manager is used to configure the Oracle client to use RADIUS for authentication. From the local profile window select **Oracle Advanced Security** from the pull-down, and then select the **Authentication** tab and move RADIUS from the available to the selected methods window. This will update the client sqlnet.ora file by adding RADIUS to the SQLNET.AUTHENTICATION_SERVICES parameter.

```
# sqlnet.ora Network Configuration File:
F:\Oracle\product\10.1.0\Client_1\network\admin\sqlnet.ora
# Generated by Oracle configuration tools.
SQLNET.AUTHENTICATION_SERVICES= (RADIUS,NTS)
NAMES.DIRECTORY_PATH= (TNSNAMES, EZCONNECT)
```

This is the final panel we went through to configure Oracle Net Manager.

Archived

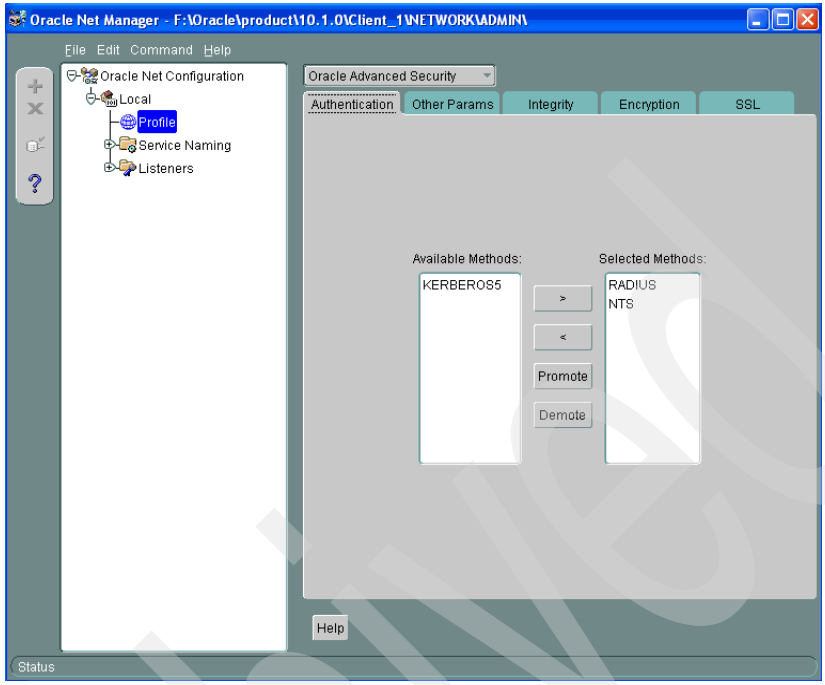


Figure 9-5 Promotion of radius

3. Connect to the Oracle DB.

   Connect TERRY1/test123@ora1 where the user ID has been created with the identified external attribute in the Oracle DB and the user ID and password are in the z/OS LDAP SDBM database.

Archived

A

VM setup and useful commands

This appendix describes the setup and configuration we used for our Linux Oracle systems. Our first concern, to avoid multiple installations and speed up our testing capacities, was to be able to start the same system either as VM guests or in LPAR mode. That implied some specific choices for the VM guest definitions, like the use of fullpack minidisks or dedicated disks, preferably to standard VM minidisks. The second concern was to be able to easily duplicate (clone) these systems. We describe the simple steps for cloning, as well as a couple of VM commands we found useful to clone and easily manage a number of Linux instances.

© Copyright IBM Corp. 2005. All rights reserved.

VM setup

This section describes our VM setup.

VM guest definition

We used the following sample definition in the USER DIRECT file on VM for our guests.

Example: A-1 VM guest user direct entry

```
USER ORACLE4 EXPRESS  1024M 16G G
   INCLUDE LINXPROF
*
*OWNER LINUX
   ACCOUNT 1 LINUX
   OPTION TODEN MIH  DEVI DEVM MAINTCCW RMCHINFO
*
   SPECIAL B000 QDIO 3 SYSTEM SWG40
   SPECIAL C000 QDIO 3 SYSTEM SWPSS              2
*
   LINK LINADMIN 0591 0191 RR
   MDISK 0100  FB-512 V-DISK 300000              3
   MDISK 0500  3390 DEVNO  1814 MR               4
   MDISK 0501  3390 DEVNO  1914 MR
   MDISK 0502  3390 DEVNO  1a14 MR
   MDISK 0503  3390 DEVNO  1b14 MR
   MDISK 0504  3390 DEVNO  1c14 MR
   MDISK 0505  3390 DEVNO  1d14 MR
   MDISK 0506  3390 DEVNO  1e14 MR
   MDISK 0507  3390 DEVNO  1f14 MR
   MDISK 0508  3390 DEVNO  1014 MR
```

Example: A-2 Linxprof entry

```
PROFILE LINXPROF
ACCOUNT LINUX
CPU 00 BASE
*PU 01
*PU 02
*PU 03
IPL CMS PARM AUTOCR
IUCV ALLOW
IUCV ANY PRIORITY
MACHINE ESA 16
SCR INA WHI NON STATA RED NON CPOUT YEL NON VMOUT GRE NON
SCR INRED TUR NON
```

Archived

```
XAUTOLOG OPERATOR HMFSERVE LINADMIN LINOPER
CONSOLE 0009 3215 T
SPOOL 000C 2540 READER *
SPOOL 000D 2540 PUNCH A
SPOOL 000E 1403 A
LINK MAINT 0190 0190 RR
LINK MAINT 019D 019D RR
LINK MAINT 019E 019E RR
LINK $$VMSYST 019B 019B RR
```

1. OSA devices (not used). We choose not to use real OSA, but to use VSWITCH instead.
2. Network Interface Card (NIC) definition for the VSWITCH adapter.
3. VDISK definition for the swap device.
4. FULLPACK Minidisk definition. We made sure all the disks were spread over different logical subsystems in the ESS (for example, over different Logical Control Units for VM. Here we use one disk on each control unit 1800 to 1F00 and 1000).

Important: The main difference between fullpack minidisk and *normal* minidisk is that a fullpack minidisk includes cylinder 0, and therefore is recognizable as a standard 3390 from an LPAR. The same would apply to a DEDicated disk, some of the differences between the two types of definition being:

- We found that a fullpack minidisk defined with DEVNO is eligible for MDC (minidisk caching), whereas a DEDicated volume is not. The SRL *z/VM 4.4 Performance*, SC24-5999, states that DEVNO is not supported; however, we found that it works. In any case, perhaps you should define your full pack minidisks with the cylinder 0 to END option, which is documented.
- DEDicated volume is the only way to use Parallel Access Volume (PAV). See: http://www10.software.ibm.com/developerworks/opensource/linux390/perf/tuning_rec_dasd_PAV.shtml#begin

VM System definition

We used the following definition in the VM SYSTEM CONFIG file to define a VSWITCH named SWPSS, to link it to the OSA adapter using OSA device 63D, under control of VM's TCPIP stack, and to authorize connection from our guest ORACLE4.

Example: A-3 VSWITCH definition in SYSTEM CONFIG file

```
Define Vswitch  SWPSS   RDEV 63D  CONN    CONTRoller TCPIP

MODIFY VSWITCH SWPSS  GRANT ORACLE4
```

The above definition also requires the statement VSWITCH CONTROLLER ON in the TCPIP configuration file.

See “Related publications” on page 227 and for more details.

Cloning

Our main concern for cloning is that we wanted it to be as simple and quick as possible. This was made possible by keeping all the guests very similar in terms of VM definition; they all use the same virtual devices numbers (disks or network adapters), so there is no need to change the Linux configuration files (especially those referring to hardware definitions, like chandev.conf and zipl.conf). We first set up a reference Linux image, and then used the following process to clone it:

1. Duplicate the source VM directory entry, keeping the same virtual devices number (disks, OSA device when used, and vswitch adapter), changing only the guest name, the real disks, and OSA devices numbers on the target guest definition.
2. Make sure both source and target guests are logged off.
3. Use the VM FLASHCopy command to flashcopy the real disks from our source Linux guest to our target Linux guest.
4. Start the new cloned guest and use YaST2 on SuSE to change network definitions (IP addresses, hostname, etc.).

Example: A-4 Cloned guest directory entry

```
USER ORACLE5 EXPRESS  1024M 16G G
   INCLUDE LINXPROF
*
*OWNER LINUX
   ACCOUNT 1 LINUX
   OPTION TODEN MIH  DEVI DEVM MAINTCCW RMCHINFO
*  DEDICATE 0600 0620
*  DEDICATE 0601 0621
*  DEDICATE 0602 0622
SPECIAL A000 HIPER 3 SYSTEM HIPER1
   SPECIAL B000 QDIO 3 SYSTEM SWG40
   SPECIAL C000 QDIO 3 SYSTEM SWPSS
*
```

```
   LINK LINADMIN 0591 0191 RR
   MDISK 0100  FB-512 V-DISK 300000
*  MDISK 0191  3390 0001 020 04ADM1 MR
MDISK 0500  3390 DEVNO  1015 MR
   MDISK 0501  3390 DEVNO  1115 MR
   MDISK 0502  3390 DEVNO  1215 MR
   MDISK 0503  3390 DEVNO  1315 MR
   MDISK 0504  3390 DEVNO  1415 MR
   MDISK 0505  3390 DEVNO  1515 MR
   MDISK 0506  3390 DEVNO  1615 MR
   MDISK 0507  3390 DEVNO  1715 MR
   MDISK 0508  3390 DEVNO  1815 MR
```

FLASHCOPY

We used the following method to flashcopy the disks under VM (requires an ESS with FLASHCOPY microcode installed). From the preceding definitions, here are the steps to flashcopy ORACLE4 disk 500 (real device 1814) to ORACLE5 disk 500 (device 1015).

1. Log on to a CMS user ID with class B authority.
2. Vary offline/online the devices to make sure the labels are consistent.
3. Attach the volumes.
4. Run the flashcopy command, and check the message for completion.

   ```
   FLASHC 1814 0 END to 1015 0 END
   ```

   Note: Once a disk is part of a flashcopy operation, either as source or target, this operation must be complete before a new flashcopy process can be initiated on the same disk.

5. Detach and the vary offline/online the volumes.

Example: A-5 Sequence of commands for FLASHCOPY operation (commands in bold)

```
Ready; T=0.01/0.01 15:51:11
vary off 1015 1814
1015 varied offline
1814 varied offline
2 device(s) specified; 2 device(s) successfully varied offline
Ready; T=0.01/0.01 15:51:20
vary on 1015 1814
1015 varied online
1814 varied online
2 device(s) specified; 2 device(s) successfully varied online
Ready; T=0.01/0.01 15:51:31
attach 1015 *
```

Archived

```
DASD 1015 ATTACHED TO MAINT 1015 WITH DEVCTL
Ready; T=0.01/0.01 15:51:48
att 1814 *
DASD 1814 ATTACHED TO MAINT 1814 WITH DEVCTL
Ready; T=0.01/0.01 15:51:52
flashc 1015 0 end to 1814 0 end
Command started: FLASHCOPY 1015 0 END TO 1814 0 END
Ready; T=0.01/0.01 15:56:08
Command complete: FLASHCOPY 1814 0 END TO 1015 0 END
det 1015 1814
1015 1814 DETACHED
Ready; T=0.01/0.01 15:56:50
vary off 1015 1814
1015 varied offline
1814 varied offline
2 device(s) specified; 2 device(s) successfully varied offline
Ready; T=0.01/0.01 15:57:01
vary on 1015 1814
1015 varied online
1814 varied online
2 device(s) specified; 2 device(s) successfully varied online
Ready; T=0.01/0.01 15:57:09
q 1015 1814
DASD 1015 ORA607  , DASD 1814 ORA607
Ready; T=0.01/0.01 15:57:14
```

```
                                                    RUNNING   MOPVM04
```

As shown by the last query command, the result of the flashcopy operations is two exactly identical volumes, having the same volume serial. To avoid duplicate volsers under VM, we used a “quick’n dirty” method to relabel the volume using ICKDSF through the CP utility CPFMTXA. The following command shows how to relabel attached device 1015 with new label ORA507.

```
cpfmtxa 1015 ORA507 label
```

The proper way of doing this would have been to use the **fdasd** command from Linux, as this would keep in sync both the VOLSER and the dummy data set name that defines the Linux partition on the disk. Some tools requires those to match.

```
fdasd -l ORA507
```

Archived

Note: Because cylinder 0 is part of the minidisk, a Linux formatting of the disk will reinitialize the volume label. Therefore it is strongly recommended to always use the **fdasd** command with -l options; otherwise a default volume label based on the virtual device number will be set, which may end up with a lot of duplicate volume serial numbers under VM.

Booting same Linux either as VM guest or LPAR

The following considerations apply for a Linux system to boot in both VM and LPAR modes:

- The hardware resources used (disks and network adapters) must match real devices; so a minidisk cannot be used unless it is a fullpack minidisk.
- Under VM, Linux deals only with virtual devices, so all Linux guests can use the same virtual devices for ease of configuration and administration. CP is doing the underlying matching between virtual and real devices, which must be different for all guests.
- Under LPAR, Linux deals only with real devices. VM-specific devices, like Vswitch or Guest LANs, cannot be used. Also, Linux configuration files defining hardware resources must be different for all systems.

The following configuration file is used to define the accessed disks.

Example: A-6 /etc/zipl.conf

```
# Generated by YaST2
[defaultboot]
default=ipl

[ipl]
target=/boot/zipl
image=/boot/kernel/image
ramdisk=/boot/initrd
parameters="dasd=500-53f,100 vmpoff='LOGOFF'  root=/dev/dasda1"

[ipllpar]
target=/opt/oracle/boot/zipl
image=/opt/oracle/boot/kernel/image
ramdisk=/opt/oracle/boot/initrd
parameters="dasd=1814,1914,1a14,1b14,1c14,1d14,1e14,1f14,1014 vmpoff='LOGOFF'
root=/dev/dasda1"
```

Archived

The [ipl] sections define the default boot parameters used for VM. Because all the guests use the same virtual disks, 500 to 508, this section will be identical for all Linux guests (for example, they will all IPL from disk 500).

The [iplpar] sections are used for LPAR mode. Here the real disks have to be specified in the dasd parameter; therefore, this section has to be adapted for each of the Linux systems.

The following example is used to define the network adapters.

Example: A-7 /etc/chandev.conf

```
noauto;
qeth1,0x0c00,0x0c01,0x0c02;add_parms,0x10,0x0c00,0x0c02,portname:PORTC0
qeth2,0xc000,0xc001,0xc002
```

We define our hardware configuration so that OSA devices C00 to C02 are shared by all LPARs on our system.

The VM guests are using a device C000 defined as a vswitch adapter. We did not include a definition for OSA device C00.

In Linux we defined with YaST2 both network devices eth1 and eth2 as having the same IP address and network definitions. When booting in any LPAR, only device c00 is detected, so the network comes up on device eth1. Under VM, only device C000 is detected, so the same network definitions come up this time on adapter eth2. The same file can be used for all the Linux images, whether they are starting in VM or LPAR mode. Only the network parameters need to be changed using YaST2.

Useful VM commands

We found the **xautolog** and **signal** VM commands useful when managing Linux from a single CMS user.

To automatically start a guest, we used the **xautolog** command:

```
xautolog oracle4 ipl 500 sto 512M
```

Where oracle4 is the guest name, 500 the IPL disk number, and 512M the amount of virtual memory we want for that guest.

To automatically shut down Linux, we used the **signal** command:

```
signal shutdown oracle4 within 120
```

Where 120 is the interval (in seconds) we allow for a proper shutdown.

Note: In order for Linux to react to a shutdown signal from VM, the following line must be present in /etc/inittab:

```
ca:12345:ctrlaltdel:/sbin/shutdown -t1 -h now
```

How to remove Oracle code

During the process of writing this document we have had to remove Oracle several times. To make the process easier we wrote a script to remove Oracle from the guest. It is a basic script but works well. It is written to remove the directories we used. You will have to modify it and test it to match the directories that you choose.

Example: A-8 Shell script to remove Oracle

```
#!/bin/sh
#################################################################
# make changes to files and directories as needed
# this needs to be run by root
#Written 09/16/2004 by Denny Dutcavich IBM
#Modified 10/20/2004 by Chris Little Oklahoma DHS
#Modified 12/20/2004 by Bruce Frank IBM
#################################################################

#Replace ORACLE-BASE with your ORACLE_BASE directory and ORACLE-HOME with your
#ORACLE_HOME directory
BASE_DIR=/ORACLE-BASE
ORACLE_HOME=/ORACLE-HOME

if [ "$UID" -ne "0" ]
    then echo 'You must be root to run this script'
   exit
fi

if test -e /var/opt/oracle/oraInst.loc
  then rm /var/opt/oracle/oraInst.loc
  echo 'oraInst.loc gone'
  else echo 'oraInst.loc not there'
fi

if test -e /etc/oratab
  then rm /etc/oratab
  echo 'oratab gone'
  else echo 'oratab not there'
fi

if test -e /usr/local/bin/oraenv
```

Archived

```
  then rm /usr/local/bin/oraenv
  echo 'oraenv gone'
  else echo 'oraenv not there'
fi

if test -e /usr/local/bin/coraenv
  then rm /usr/local/bin/coraenv
  echo 'coraenv gone'
  else echo 'coraenv not there'
fi

if test -d $BASE_DIR/admin
  then rm -r $BASE_DIR/admin
  echo 'admin directory gone'
  else echo 'admin directory not there'
fi

if test -d $BASE_DIR/flash_recovery_area
  then rm -r $BASE_DIR/flash_recovery_area
  echo 'flash_recovery_area gone directory'
  else echo 'flash_recovery_area directory not there'
fi

if test -d $BASE_DIR/oraInventory
  then rm -r $BASE_DIR/oraInventory
  echo 'oraInventory directory gone'
  else echo 'oraInventory directory not there'
fi

if test -d $ORACLE_HOME/O10g
  then rm -r $ORACLE_HOME/O10g
  echo 'O10g directory gone'
  else echo 'O10g directory not there'
fi


if test -d $BASE_DIR/oradata
  then rm -r $BASE_DIR/oradata
  echo 'oradata directory gone'
  else echo 'oradata directory not there'
fi

echo 'bye bye'
```

Archived

B

Overview of ESALPS

For the measurements and analysis for this book, ESALPS (Linux Performance Suite) from Velocity Software was used because of its capabilities and focus on the z/VM and Linux environments.

© Copyright IBM Corp. 2005. All rights reserved.

ESALPS overview

ESALPS, the Linux Performance Suite, is a suite of products provided by Velocity Software for measurement and tuning of z/VM, Linux under z/VM, and applications in this environment. The four products that make up this suite are:

- ESAMAP, the VM Monitor Analysis Program, providing performance reports on all aspects of VM/ESA and z/VM performance
- ESAMON, the VM Real Time Monitor, providing real-time data collection and analysis of performance
- ESATCP, the network and Linux data collection program
- ESAWEB, a very fast VM-based Web server

ESALPS features

ESALPS provides the following features:

- Full function z/VM performance monitor allows for monitoring of all z/VM subsystems, including full reporting, real-time analysis, and management reporting.
- Network Monitor: Allows monitoring of your Internet servers via SNMP.
- Linux Monitor: Allows monitoring of your Linux disks, storage, and processes. When running under z/VM, the CPU busy statistics from the /proc file system are incorrect, sometimes by as much as an order of magnitude. ESALPS has access to the true data, as measured by z/VM. It is the only monitor on the market today that corrects the data returned by the /proc file system, so that reports including CPU time by Linux guests under z/CVM are correct.
- Monitoring NT, SUN, HP: ESALPS will monitor any server that supports the standard SNMP mibs. This includes network information, disk and storage information, and process data. ESALPS also includes custom mibs through an rpm-based install that provides additional information.
- Web serving: ESALPS includes a full-function Web server to provide browser access to the ESALPS reports, and also to allow Web site support and CGI development on the VM platform.
- Alerts: Many different types of alerts can be set up for network availability monitoring. These include z/VM subsystem errors and threshold exceptions, and Linux errors and threshold exceptions.
- Full Performance Database (PDB): ESALPS provides both real-time data and historical data for in-depth analysis. The performance data is collected daily with a one-minute granularity based on the monitor interval. A longer-term

Archived

archive is collected, with a default granularity of 15 minutes. This PDB includes VM data, Linux data, and network data.

Critical agent technology

One of the powerful features of ESALPS is the agent technology. Agents running inside Linux may be either active agents or passive agents. An active agent will wake up every few seconds on a regular schedule, collect data, possibly write data to a log, and then sleep. A passive agent will wait for an external data request and will not otherwise wake up and use resource.

The issue is that under z/VM, we expect to have maybe hundreds of Linux servers, of which many will be idle. With an active agent in each of them, the Linux servers are not idle, and may consume excessive resources just by processing the active agent requests. Thus, the only way to conserve resources is to eliminate the active agent from most or all of the servers.

With the passive technology enabled by using SNMP agents, ESALPS will know that a server is idle because of the z/VM data, and will not have to wake up an agent inside Linux, thus causing additional paging and CPU consumption, to determine and record the fact that the server is idle. Thus, the idle servers will not be measured until they become active.

The operational cost of agents may be quite high. On other non-virtualized servers, the cost is relatively insignificant, as it only impacts the applications on that one server. In a virtualized environment where sharing resources is a primary objective, resources used by each agent reduce the resources available to all of the other servers. Minimizing the cost of running these agents is critical to the success of monitoring many Linux servers under z/VM. The SNMP passive agent technology is believed to be the lowest cost agent technology available.

NETSNMP extensions

The design of NETSNMP and SNMP host-specific MIBs was found lacking in a few areas. Velocity Software has extended the mibs supported by NETSNMP. This agent can be downloaded from Velocity Software's Web site. This corrects measured values such as swap rate, and provides added process data thought necessary to meet the needs of this environment. This agent is also modified to use less resource than the NETSNMP that would come with the normal Linux distributions.

Monitoring requirements

There are many requirements for data collection met by ESALPS. Data is provided for:

- Capacity planning: Long-term data in the form of a performance database (PDB) is needed as input to long-term capacity planning and trend analysis. Full historical data functions are provided with collection and may forms of data extraction tools.
- Performance analysis: Trend data allows the analyst to detect performance changes in any of thousands of potential problem areas. The performance database allows analysts to determine what of many potential changes occurred in the system. Reporting on specific periods of time can be performed, allowing in-depth performance analysis of performance problems.
- Real-time performance: Beyond the traditional entry-level real-time performance reporting of the top users and system utilization, in real time.
- Performance analysis is provided for all subsystems—user activity as well as Linux (and many other platforms) servers. Network data is also provided in real time.
- Linux data: With the advent of virtual Linux server farms on z/VM, performance data from the Linux guest machine is required. ESALPS provides a centralized repository for this data.

Standard interface

ESALPS uses standard interfaces for all data collection. The advantage to using the standard interfaces provided is that when there are a multitude of releases and distributions available, the standard interfaces will provide consistent data sources.

- z/VM provides a *monitor interface* that has been available since 1988. Since then, this interface has provided a consistent view of performance of VM systems.
- Network performance is collected using SNMP, the standard for network management.
- NETSNMP, an open source software package provides host data for Linux and other platforms.
- VM Application data interface is used by applications to insert data into the monitor stream consistent with the standard monitor interface. ESATCP uses this interface to ensure consistent data collection that allows full integration of Linux and VM data.

Related publications

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

IBM Redbooks

For information on ordering these publications, see “How to get IBM Redbooks” on page 228. Note that some of the documents referenced here may be available in softcopy only.

- *Experiences with Oracle Database on Linux on zSeries*, SG24-6552 (Oracle9*i*)
- *Linux for IBM zSeries and S/390: Distributions*, SG24-6264
- *Linux for S/390*, SG24-4987
- *Linux on IBM zSeries and S/390: ISP/ASP Solutions*, SG24-6299
- *Building Linux Systems under IBM VM*, REDP0120
- *Linux on zSeries and S/390: Systems Management*, SG24-6820
- *z/VM and Linux on zSeries: From LPAR to Virtual Servers in Two Days*, SG24-6695
- *Linux Handbook A Guide to IBM Linux Solutions and Resources*, SG24-7000

Other publications

These Oracle publications, available on otn.oracle.com, are also relevant as further information sources:

- IBM Tivoli Storage Manager for Linux, Quick Start Version 5.2
- Oracle Database 10g for Linux on zSeries—*Oracle Database Release Notes for Linux on zSeries,* B13964
- *Oracle Database Installation Guide for UNIX Systems*, B10811
- *Oracle Database Administrator’s Reference for UNIX Systems*, B10812
- *Oracle Real Application Clusters Installation and Configuration Guide 10g Release 1 (10.1) for AIX-Based Systems, hp HP-UX PA-RISC (64-bit), hp*

Archived

© Copyright IBM Corp. 2005. All rights reserved.

Tru64 UNIX, Linux, Solaris Operating System (SPARC 64-bit), and Windows Platforms Part No. B10766-08

- *Data Protection for Oracle for UNIX Installation and User's Guide Version 5.2*
- *Database Backup and REcovery Advanced User's Guide for 10g Release 1* (10.1) Document B10734-01
- Database Recovery Manager Reference Guide Document B10770-01

Online resources

These Web sites and URLs are also relevant as further information sources:

- Oracle Technical Network

  http://otn.oracle.com

- Oracle home page

  http://www.oracle.com

- Oracle Certification information

  http://www.otn.oracle.com/support/metalink/content.html

- VM performance information

  http://www.vm.ibm.com/perf/tips/

How to get IBM Redbooks

You can search for, view, or download Redbooks, Redpapers, Hints and Tips, draft publications and Additional materials, as well as order hardcopy Redbooks or CD-ROMs, at this Web site:

ibm.com/redbooks

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services

Archived

Index

Symbols
/adsmorc 145
/etc/host 94
/etc/init.d 87
/proc/sys/kernel 28
/tmp 43
/var/opt/oracle/oraInst.loc 92

Numerics
64-bit architecture 2

A
absolute share 15
ACUCOBOL-GT 157
Alerts 224
APAR VM63282 179
Archive and Retrieval 122
archive files 150
archive log files 124
ASM 55
ASM best practices 74
ASM instance 58, 60
ASM.LIB 57
async IO 50
authenicate externally 201
Automated Storage Manager 55

B
Backup and Restore 122
backup strategy 152
binding raw devices 86
bunzip 132

C
Capacity Planning 226
capping. see LPAR capping
capture ratio 182
catalog database 148
central storage 12
cleanup scripts 100
clients.conf file 204
cloning 213, 216
cluster nodes 88
Cluster Ready Services 77
Configuration Guidelines 172
costs
  networking 3
  software 3
  support 3
  system management 3
cpio 27, 132
CPU Allocation 11
CRS 77, 90–91, 99
CyberTrust 8

D
Data Mining 7
Database Creation Assistant 77
database instance 109
database password 41
DBCA 77
dbca command 110
DBCA utility 58
DCSS 184
dedicated processors 199
dedicated volume 17
Detecting Storage Problems 182
DIAG 44 190
disk discovery 68
diskgroup 68
DISPLAY variable 90, 110
dormant users 184

E
Enterprise Manager 56
Entrust 8
ESALPS 171, 223
ESAMAP 224
ESAMON 19, 224
ESATCP 224
ESAWEB 224
ESCON 18
Ethernet 77
Expanded Storage 180

Archived

© Copyright IBM Corp. 2005. All rights reserved.

expanded storage 12, 182
external authentication 201

F
failed CRS installation 100
FCP 18
fdasd command 218
FICON 18
file system 56
filesystem 22, 85
Fixed I/O buffers 173
Flash Recovery Area 66
flashcopy 217–218
FreeRadius 203
ftp 27
full database restores 124
FULLPACK Minidisk 215

G
Gigabit 77
group
    DBA 23
    OINSTALL 23
Guest Lan 14
Guest Lans 219

H
hardware costs 3
Hierarchical Space Management 122
high speed interconnect 77
HSM 122
Hyper sockets 77

I
I/O bottlenecks 20
I/O considerations 18
I/O Subsystem 187
IBM Performance Toolkit for VM 19
ICKDSF 218
ICUV 77
IFL 5
instance name 110
Integrated Facility for Linux 5
inter-server communication 3

K
Kerberos 203
kernel settings 28

L
Label Security 7
LD_LIBRARY_PATH 29
LDAP 201
Linux Data 226
Linux distributions 9
Linux Monitor 224
LOCK_SGA parameter 51
Logical Volume Manager 187
logical volumes 82
LPAR 23, 213
LPAR capping 198
LPAR options 198
LVM 187

M
media recovery 150
memory 19, 172
metadata 123
MicroFocus
    Server Express V4.0 Compiler 163
MicroFocus compiler 163
Minidisk Cache 182
monitor resources 20

N
NETSNMP Extensions 225
Network Monitor 224
networking costs 3
NIC 215
nventory Directory Panel 32

O
OCFS2 57
OCR disk 91
OEM 52
OLAP 7
OLTP 10, 18
Oracle
    Automated Storage Manager 55
    Cluster Ready Services 91
Oracle 11i EBusiness Suite 7
Oracle Advanced Security Option 202
Oracle Application Server 10g 7
Oracle Collaboration Suite 7–8

Archived

Oracle Database Server 7
Oracle E-Business Suite 8
Oracle Enterprise Manager. 109
Oracle Management Server (OMS) 8
Oracle production instance 23
Oracle Recovery Catalog 141
Oracle Universal Installer 21, 24, 91
ORACLE_BASE 29
ORACLE_HOME 29
ORACLE_INVENTORY 29
ORACLE_SID 29
Oracle9i Application Server 7–8
OracleText 8
oraninstRoot.sh script 34
OSA devices 215
OUI 91
overcommit storage 182

P

paging 12, 182
Paging considerations 11
paging space 17, 20
Parallel Access Volumes 75
Partitioning 7
password management 46
PATH 29
Performance Analysis 226
performance concerns 11
Performance Toolkit for VM 19
point in time 150
PR/SM 5
Private Node 91
Pro*C/C++ 166
Pro*COBOL 7, 154
Pro*Cobol precompiler 163
Program Global Area 12
Public Node 91
PuTTY 24

Q

qualities of service 3

R

RAC 77
RACF 201–202
Radius Server 201
radiusd.conf file 204
RAID 18
raw devices 57, 68, 82
raw disks 57
Real Application Clusters 7, 77
recover database 147
recovery log 123
Redbooks Web site 228
  Contact us xvii
RedHat 4 U1 9
RedHat 4.0 U1 22
relative share 15
relinking oracle 50
Remove Oracle Code 221
restore database 146
RMAN 56, 124, 127, 140
root.sh script 93, 99, 106
rpm 132

S

schema password 41
scp 88
secureshell 88
sem 28
semopm 28
server consolidation 5
server farms 6
setting up the xWindows connection 21
SGA 28, 38
share of virtual machines 15
shared
  DASD 77
  database 77
  disk 77
shared processors 199
sharing resources 172
shmmax 28
shmmni 28
shutdown 220
SID name. 110
signal command 220
sizing 9, 13
sizing a guest 12
Sizing and Tuning memory 11
sizing workloads 9
slapd.conf file 207
SLES8 9
SLES9 9, 22
SNMP agent 225

Archived

software costs 3
space requirements 42
Spatial 7
spin locks 190
SSH 88
ssh command 90
Statspack 53
storage
central 12
storage management 56
storage management policies 123
storage requirement 172
striped disks 187
swap space 17, 20
symbolic links 85
system security 123
Systems Global Area 12, 28
systems management costs 3

T

tablespaces 124
TCPIP 215
t-disk 17
TDP
Web Administrator 142
TDPO client 139
tdpoconf 135
Tivoli Data Protect for Oracle (TDPO) 121
Tivoli Storage Manager 121–122
traditional minidisk 17
TSM Server 139

V

var/opt/oracle 100
VDISK 17, 215
Velocity Software 19
Verisign 8
VIP Node 91
virtual CP 14
virtual devices, 219
Virtual Switch. see VSWITCH
VM 213
commands 213
Guest definition 214
minidisks 213
Setup 214
VM FLASHCopy 216
VM Monitor Analysis Program 224
VM Real Time Monitor 224
vmstat 14
VNC server 24
VNC viewe 24
vncserver 90
Volume Manager 56
Voluntary Time Slice End. see DIAG 44
Voting Disk 91
VSWITCH 179, 215
Vswitch 14, 219

W

Welcome panel 31
workload characteristics 13

X

xautolog 220
X-Terminal 25
xWindows connection 21
xWindows interface 24

Y

YAST command 23
YaST2 220

Z

z/VM 5

Archived

Experiences with Oracle 10g Database for Linux on zSeries

Archived

Archived

Archived

Experiences with Oracle 10g Database for Linux on zSeries

Installing a single instance of Oracle Database 10g

Installing Oracle 10g RAC

Using ASM

Linux on zSeries offers many advantages to customers who rely upon IBM mainframe systems to run their businesses. Linux on zSeries takes advantage of the qualities of service in the zSeries hardware—making it a robust industrial strength Linux. This provides an excellent platform for consolidating Oracle databases that exist in your enterprise.

This IBM Redbook describes experiences gained while installing and testing Oracle10g for Linux on zSeries, such as:

- Installing a single instance database of Oracle10g instances for Linux on zSeries
- Installing Cluster Ready Services (CRS) and Real Application Clusters (RAC)
- Performing basic monitoring and tuning exercises
- Using options such as:
  - IBM's Tivoli Data Protector (TDP) and Tivoli Storage Manager (TSM)
  - Automated Storage Manager (ASM)
  - LDAP and Radius Server for security
  - COBOL and C programs with Oracle

Interested readers include database consultants, installers, administrators, and system programmers.

INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION

BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

For more information:
ibm.com/redbooks

SG24-6482-00 ISBN 0738493899