

Oracle on LinuxONE

Susan Adamovich

Sam Amsavelu

Armelle Chev

Helene Grosch

Guilherme Nogueira

Jagannath Raju

Moshe Reder

Mariana Roveri

David J Simpson

Narjisse Zaki





International Technical Support Organization

Oracle on LinuxONE

August 2019

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

Second Edition (August 2019)

This edition applies to Oracle Database 12cR1, Oracle Database 12cR2, SUSE Linux Enterprise Server and Red Hat Enterprise Linux Servers.

© Copyright International Business Machines Corporation 2019. 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

Notices	vii
Trademarks	viii
Preface	ix
Authors	ix
Now you can become a published author, too!	xi
Comments welcome	xi
Stay connected to IBM Redbooks	xi
Chapter 1. A shared commitment to client value with IBM and Oracle Alliance	1
1.1 Discover the possible	2
1.2 Solution overview	2
1.3 LinuxONE family overview and architecture	2
1.4 How it works	3
1.5 Competitive advantage	3
Chapter 2. Setting up Linux guests to install Oracle Database 12c Release 1	5
2.1 Obtaining Oracle documentation, Oracle code, and My Oracle Support notes.	6
2.1.1 Master Note of Linux OS Requirements for Database Server	6
2.1.2 Pre-installation requirements documentation	6
2.1.3 Obtaining code for Oracle 12cR1 on IBM LinuxONE	7
2.1.4 Oracle database pre-installation tasks in Oracle online Documentation.	7
2.1.5 Linux Large Pages and Oracle Databases	7
2.2 Red Hat Enterprise Linux and SUSE Linux Enterprise Server setup	8
2.2.1 Important information	8
2.2.2 Pre-installation checking	9
2.3 Red Hat Enterprise Linux Server 6 or 7 Specific Setup	16
2.3.1 Important information	16
2.3.2 Pre-installation checking	16
2.4 SUSE Linux Enterprise Server specific setup	17
2.4.1 Important information	18
2.4.2 Pre-installation checking	18
2.5 Installation	20
2.5.1 Oracle Universal Installer	20
2.5.2 Updating the Oracle user profile	22
2.5.3 Creating the database	22
Chapter 3. Using Oracle Enterprise Manager Cloud Control Agent to manage Oracle Database 12c Release 1	23
3.1 Basic Enterprise Manager Cloud Control Architecture	24
3.2 Creating the Enterprise Manager Cloud Control Architectures	25
3.2.1 Downloading the installation files	26
3.2.2 Installing and configuring Enterprise Manager Cloud Control 13c	26
3.3 Updating the agent at the Enterprise Manager Cloud Control to monitor Oracle Databases on LinuxONE (online)	29
3.3.1 Configuring the Software Library storage space	30
3.3.2 Acquiring the LinuxONE agent in online mode	30
3.3.3 Setting up My Oracle Support Credentials	30
3.3.4 Verifying that Enterprise Manager Cloud Control is in online mode	30

3.3.5 Downloading and applying the LinuxONE Management Agent Software	31
3.4 Updating the agent at the Enterprise Manager Cloud Control to monitor Oracle Databases on LinuxONE (offline)	31
3.4.1 Acquiring the LinuxONE agent in offline mode	32
3.4.2 Verifying that Enterprise Manager Cloud Control is in offline mode	32
3.4.3 Updating the agents in Enterprise Manager Cloud Control using the Self Update feature	32
3.5 Deploying the agents from the Cloud Control console	35
3.6 Deploying the agents in silent mode	37
3.7 Adding the databases for monitoring	40
3.8 Summary	41
Chapter 4. Oracle encryption on LinuxOne	43
4.1 Oracle Transparent Data Encryption on Oracle Database 12cR2	44
4.1.1 Benefits of the use of Transparent Data Encryption	44
4.1.2 Installing Transparent Data Encryption	44
4.1.3 Configuring and administering Transparent Data Encryption	44
4.2 Enabling encryption on Oracle Database	48
4.3 Managing TDE master keys and wallets	54
4.4 Performance	56
4.5 Using Oracle TDE on Oracle RAC	59
4.6 Summary	60
Chapter 5. Successful consolidation project: From sizing to migration	61
5.1 Delimitation of the project scope	62
5.1.1 Certification and support	62
5.1.2 Best candidates for consolidation	62
5.1.3 Non-functional requirements	62
5.1.4 Business value analysis	63
5.2 Sizing	63
5.2.1 CPU	63
5.2.2 Memory	64
5.2.3 I/O	66
5.3 Proof of concept	67
5.3.1 PoC qualification	67
5.3.2 Execution	68
5.3.3 Analyzing and reporting	70
5.4 Migration	71
5.4.1 Considerations before migration	71
5.4.2 Available techniques for cross-platform migration	73
5.4.3 Considerations when migrating from File System to ASM or vice versa	77
5.4.4 Best practices	77
5.5 Successful production	79
Chapter 6. Performance management for IBM z/VM, Linux, and Oracle on LinuxONE	81
6.1 Oracle performance monitoring overview	83
6.2 Oracle database tools	84
6.2.1 Oracle Diagnostics Pack	84
6.2.2 Oracle Tuning Pack for Oracle Database	84
6.2.3 Oracle Enterprise Manager Cloud Control	85
6.2.4 STATSPACK	85
6.2.5 Oracle Real Application Testing	85
6.2.6 Oracle Orion tool	86
6.2.7 Downloadable tools from Oracle	86

6.3 Hypervisor monitoring tools	87
6.3.1 IBM Performance Toolkit for VM feature	87
6.4 zVPS	87
6.5 IBM Wave for z/VM	88
6.6 OMEGAMON XE on z/VM and Linux	88
6.7 Linux monitoring tools	90
6.8 Summary	91
Chapter 7. High Availability	93
7.1 Oracle HA options	94
7.1.1 Oracle Real Application Clusters	96
7.1.2 Oracle RAC One Node	96
7.1.3 Oracle Clusterware	96
7.1.4 Edition-based redefinition	97
7.1.5 Online Reorganization and Redefinition	97
7.1.6 Oracle Flashback	97
7.1.7 Oracle Data Guard/Active Data Guard	98
7.1.8 Application Continuity	98
7.1.9 Oracle GoldenGate	98
7.1.10 Global Data Services	99
7.1.11 Storage-based replication	99
7.1.12 Oracle Site Guard	100
7.1.13 Oracle Recovery Manager	100
7.1.14 Oracle Secure Backup	100
7.1.15 Oracle Multitenant	100
7.2 HA building blocks for Oracle Database on Linux running on LinuxONE	101
7.2.1 Hardware provided HA	102
7.2.2 Operating system HA	102
7.2.3 Oracle provided HA	103
7.3 IBM LinuxONE with Oracle Database: Considerations and examples for HA	103
7.3.1 Networking	103
7.3.2 Oracle HA networking options	104
7.3.3 Oracle RAC Interconnect z/VM Link Aggregation	105
7.3.4 Oracle RAC interconnect considerations	105
7.3.5 I/O channel failover considerations	106
7.3.6 Government Client Example: Oracle MAA RAC and Data Guard	108
7.4 Summary and recommendations	109
Chapter 8. Surrounding Oracle with open source software	111
8.1 Open source database managers on LinuxONE	112
8.1.1 Database types	112
8.1.2 PostgreSQL	113
8.1.3 MongoDB overview	114
8.1.4 Integrating Apache Spark with Oracle and MongoDB databases that are running on LinuxONE	116
8.2 Summary	119
Appendix A. Oracle Database 11gR2 on Red Hat Enterprise Linux Servers 7	121
A.1 Pre-Installation requirements documentation	122
A.2 Obtaining the code for Oracle 11gR2 on IBM LinuxONE	122
A.3 Java version for 11gR2 installations on Red Hat 7	122
A.4 Oracle RAC installation issue when version 11.2.0.4 is used with Red Hat Enterprise Linux server 7	123
A.5 Security-Enhanced Linux	124

A.6 Pre-installation checking.....	124
A.6.1 Operating system requirements for IBM LinuxONE	124
A.6.2 Oracle RPM Checker with Red Hat Enterprise Linux Server on LinuxONE	124
A.7 Oracle Universal Installer	125
Related publications	127
IBM Redbooks	127
Online resources	127
Help from IBM	127

Notices

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

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 grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing, IBM Corporation, North Castle Drive, MD-NC119, Armonk, NY 10504-1785, US

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 jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

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

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

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

The performance data and client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

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.

Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

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 actual people or business enterprises is entirely coincidental.

COPYRIGHT LICENSE:

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

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at <http://www.ibm.com/legal/copytrade.shtml>

The following terms are trademarks or registered trademarks of International Business Machines Corporation, and might also be trademarks or registered trademarks in other countries.

AIX®
DB2®
Domino®
ECKD™
FICON®
FileNet®
HiperSockets™
IBM®
IBM LinuxONE™
IBM LinuxONE Rockhopper™
IBM Z®
Informix®
InfoSphere®
Interconnect®
OMEGAMON®
Optim™
Rational®
Redbooks®
Redbooks (logo) ®
Resilient®
S/390®
System z®
Tivoli®
WebSphere®
z Systems®
z/VM®

The following terms are trademarks of other companies:

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 and other countries.

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

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

Java, and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

UNIX is a registered trademark of The Open Group in the United States and other countries.

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

Preface

Oracle Database 12c running on Linux is available for deployment on IBM® LinuxONE. The enterprise-grade Linux on LinuxONE solution is designed to add value to Oracle Database solutions, including the new functions that are introduced in Oracle Database 12c.

In this IBM Redbooks® publication, we explore the IBM and Oracle Alliance and describe how Oracle Database benefits from LinuxONE. We then explain how to set up Linux guests to install Oracle Database 12c. We also describe how to use the Oracle Enterprise Manager Cloud Control Agent to manage Oracle Database 12c Release 1.

Additionally, we discuss encryption for Oracle using Oracle Transparent Data Encryption (TDE) on Oracle 12c Release 2.

We also describe a successful consolidation project from sizing to migration, performance management topics, and high availability.

Finally, we end with a chapter about surrounding Oracle with Open Source software.

The audience for this publication includes database consultants, installers, administrators, and system programmers. This publication is not meant to replace Oracle documentation, but to supplement it with our experiences while installing and using Oracle products.

Authors

This book was produced by a team of specialists from around the world working with the IBM Redbooks group, Poughkeepsie Center.

Susan Adamovich is a business development executive bringing over 25 years of high tech experience covering many disciplines including alliance management, technical enablement, sales and marketing. Today Susan brings over a decade of experience as the Oracle LinuxONE and Z server lead and global business development focal point for the IBM and Oracle technology partnership within IBM Systems.

Sam Amsavelu is a IBM certified consulting I/T architect providing technical support for Oracle solutions on IBM platforms. During the last 22 years with IBM, he worked with many customers in developing solutions to resolve their business problems with IBM technology and products.

Armelle Chev is an IBM Z® Client Technical Specialist in the Advanced Technical Support (ATS) IBM Oracle Center (IOC), located in Montpellier, France. She has over 20 years of IT experience, including development in C/C++, SQL, and performance benchmark tests for European customer accounts. Since 2001, Armelle has been providing presales in support of new customer's workloads for IBM and, most recently, for Oracle Solutions that are running on IBM LinuxONE™. This support includes technical support, consolidation sizing, architecture, and performance studies.

Helene Grosch is a certified architect, working since 2010 in the IBM Oracle Center in IBM Client Center Montpellier, France. In this pre-sales organization, she leads projects with Oracle solutions on IBM Z and LinuxONE. Her previous focus was on consolidation projects on Linux on IBM Z solutions in the New Technology Center, Montpellier. She has 18 years of experience in IT. Before joining IBM in 2007, she was a software developer for business applications on Mainframes in a Consultant and System Integration company, mainly for public sector.

Guilherme Nogueira is an IT Specialist for IBM Global Technology and Services in Brazil. He has six years of experience in configuring Linux on IBM Z. He supported over 1,800 Linux on Z servers as part of his previous role in Linux on Z support for IBM internal account. He also worked in Auto Server Provisioning, supporting Linux on Z deployments for IBM internal accounts. He is now part of an internal application team to map IBM Z hardware connections in data centers. He holds a degree in Information Security and Computer Networking from Faculdade de Tecnologia de Americana (FATEC Americana). His areas of expertise include Linux, systems automation, security, and cryptography.

Jagannath Raju is an Oracle Lead Technical Specialist working at IBM in Melbourne Australia. He provides technical advice and expertise to support the development, implementation, support, and administration of technology solutions across Middleware and LinuxONE systems.

Moshe Reder provides Global Sales Support at the IOICC (IBM Oracle International Competency Center). Moshe has over 30 years of experience with the IT industry and spent several years in the IBM Sales organization as a technical sales support for IBM Large Systems (IBM Brazil and IBM USA). He also spent a few years at the IBM Programming Laboratory (performance and benchmark for IBM S/390®, now IBM Z, and at the Technical Office for IBM Storage products). Moshe has been with the IOICC since its inception in 1997 and he focuses on the Oracle Technology for IBM Systems.

Mariana Roveri is a Senior IT Specialist for IBM Global Technology and Services in Brazil. She has more than 10 years of experience working as a Database Administrator. She is working as a Database Administrator for the American Express account. She holds a degree in Information Technology from UNICAMP and post-graduate degree MBA in Database Management. She is IBM L1 IT Specialist and her areas of expertise includes databases (Oracle, MongoDB, and IBM DB2®), high availability for Databases, and Docker. Mariana co-authored Getting Started with Docker Enterprise Edition on IBM Z, SG24-8429

David J Simpson is part of the IBM North America Channels and ISV Solutions technical sales team with almost 30 years of experience in the IT Industry. He is an Oracle certified Oracle specialist working with Oracle Database since version 5 of this product.

Narjissee Zaki is working in IBM Oracle Center in Montpellier, France. She provides technical sales support for Oracle projects on IBM Z and LinuxONE, Most recently, she provides for Open Source Databases, including architectural studies, consolidation sizing, performance analysis, and conducting briefings and workshops for IBM clients. Narjissee is certified in IBM Z Technical Support V6. She holds a Master of Science degree in Computer Science and Management. Before joining the IOC in 2016, Narjissee was involved in an Agile Project Management leading projects as a Scrum Master in the IBM Client Center Montpellier.

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

Lydia Parziale, Robert Haimowitz
IBM Redbooks, Poughkeepsie Center

Wayne T. Martin
IBM USA

Andrew Braid, Nora Kissari, Majidkhan Remtoula
IBM Montpellier

Now you can become a published author, too!

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

Find out more about the residency program, browse the residency index, and apply online at:
ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

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

- ▶ Use the online **Contact us** review Redbooks form found at:
ibm.com/redbooks
- ▶ Send your comments in an email to:
redbooks@us.ibm.com
- ▶ Mail your comments to:

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

Stay connected to IBM Redbooks

- ▶ Find us on Facebook:
<http://www.facebook.com/IBMRedbooks>
- ▶ Follow us on Twitter:
<http://twitter.com/ibmredbooks>
- ▶ Look for us on LinkedIn:
<http://www.linkedin.com/groups?home=&gid=2130806>
- ▶ Explore new Redbooks publications, residencies, and workshops with the IBM Redbooks weekly newsletter:
<https://www.redbooks.ibm.com/Redbooks.nsf/subscribe?OpenForm>
- ▶ Stay current on recent Redbooks publications with RSS Feeds:
<http://www.redbooks.ibm.com/rss.html>



A shared commitment to client value with IBM and Oracle Alliance

Since 1986, Oracle and IBM have been providing clients with compelling joint solutions, combining Oracle's technology and application software with IBM's complementary hardware, software and services solutions. More than 100,000 joint clients benefit from the strength and stability of the Oracle and IBM alliance. Through this partnership, Oracle and IBM offer technology, applications, services and hardware solutions that are designed to mitigate risk, boost efficiency and lower total cost of ownership.

IBM is a Platinum level partner in the Oracle Partner Network, delivering the proven combination of industry insight, extensive real-world Oracle applications experience, deep technical skills and high performance servers and storage to create a complete business solution with a defined return on investment. From application selection, purchase and implementation to upgrade and maintenance, we help organizations reduce the total cost of ownership and the complexity of managing their current and future applications environment while building a solid base for business growth.

This chapter includes the following topics:

- ▶ “Discover the possible” on page 2
- ▶ “Solution overview” on page 2
- ▶ “How it works” on page 3
- ▶ “Competitive advantage” on page 3

1.1 Discover the possible

Growth affects your business. Coupled with the transformative nature of Open Source and Linux, this growth puts stress on infrastructure. More clients, transactions, servers, mobile devices, networking, and databases that are supporting business growth can create significant challenges for any enterprise, leading to infrastructure with unnecessary complexity and costs.

IBM and Oracle provide a premier enterprise Linux infrastructure that is simple, secure, scalable, and cost-effective in supporting your transformation into the digital economy.

1.2 Solution overview

Enterprises require a trusted Linux infrastructure that is secure, scalable, and available to support both mission-critical work and the development and deployment of new workloads. Coupled with innovative solutions from Oracle, the LinuxONE Emperor II and IBM LinuxONE Rockhopper™ II can deliver advanced performance, security, resiliency, availability, and extreme virtualization to provide high levels of qualities of service.

1.3 LinuxONE family overview and architecture

Linux is available on various computing platforms, from set top boxes and handheld devices to the largest servers. After your applications are running on Linux, you are no longer tied to a specific hardware platform. You control the choice of hardware platform that supports your application.

Workloads that are running on LinuxONE benefits from a hardware platform that includes specialized processors, cryptographic cards with dedicated RISC processors, and a combination of hypervisors that allow unparalleled flexibility in Linux deployment.

A major benefit of Linux is that it is open source. Hundreds of Linux distributions are available for almost every computing platform. The following enterprise distributions¹ of Linux are tested on LinuxONE (Red Hat and SUSE are the certified and supported Linux distributions for deploying Oracle solutions on LinuxONE):

- ▶ [Red Hat: Red Hat Enterprise Linux \(RHEL\)](#)
- ▶ [SUSE: SUSE Linux Enterprise Server](#)
- ▶ [Canonical: Ubuntu Server](#)

These Linux distributions provide customers who are using Linux with various support options, including 24 x 7 support with one-hour response time worldwide for customers who are running production systems. Including the Linux operating system, all of the major Linux distributions offer several other open source products that they also fully support.

LinuxONE is a portfolio of hardware, software, and solutions for an enterprise-grade Linux environment. It is designed to run more transactions faster and with more security and reliability specifically for the open community. It fully embraces open source-based technology.

¹ A Linux distribution is a complete operating system and environment including compilers, file systems, and applications, such as Apache (Web server), SAMBA (file and print), Sendmail (mail server), Tomcat (Java application server), and MySQL (database).

The following are some key points to consider that tie to the value of deploying Oracle on IBM LinuxONE:

- ▶ Scale and grow Oracle Database 12c applications and data with confidence
- ▶ Simplify IT operations with advanced virtualization, multi-architecture workload support and operations management capabilities
- ▶ Minimize network security vulnerabilities, and reduce latency

1.4 How it works

IBM LinuxONE can help deliver advanced performance, security, resiliency, availability and virtualization for a high quality of service. Additionally, offering massive scalability for secure data serving, high-volume transaction processing and large-scale consolidation.

LinuxONE is exceptionally good for deploying Oracle data-serving workloads. The Emperor II delivers outstanding transaction processing and data serving performance for excellent economies of scale and more-efficient use of critical data. With up to 170 LinuxONE cores up to 32 TB of memory, up to 85 logical partitions for secure workload isolation and simultaneous multithreading (SMT) support, the Emperor II is ideally suited for consolidating large-scale distributed environments and for new in-memory and Java workloads.

The IBM LinuxONE Rockhopper II, the newest member of the LinuxONE family, is a single-frame system in a 19-inch industry standard rack allowing it to sit side-by-side with any other platform in a data center. The Rockhopper II also supports SMT, up to 30 Linux cores and up to 8 TB of memory and up to 40 logical partitions for secure workload isolation. This system is ideal for any growing business that seeks to use LinuxONE technologies' qualities of service, flexibility and performance.

These servers can support up to thousands of virtual Linux servers on a single footprint. That means that the virtualization capabilities in a single LinuxONE system can result in a less complex Linux infrastructure with fewer components, less management, less space requirements, and optimized core workload efficiency.

For more information about the value of deploying Oracle Database Release 12cR1 on IBM LinuxONE servers, see the solution data sheet available on the [IBM Techdocs support website](#).

1.5 Competitive advantage

Oracle and IBM experts work together to provide design, development, testing, technical support, and education to support your Oracle solutions that are running Linux on LinuxONE servers. These resources are provided to help ensure your solution performance and provide training to your IT staff.

For more information about how LinuxONE provides the perfect platform for running Oracle and Open Source mission-critical Linux applications for fast deployment and cost savings, see the [IBM LinuxONE website](#).



Setting up Linux guests to install Oracle Database 12c Release 1

In this chapter, we describe best practices in setting up Linux guests to install Oracle Database 12cR1. We include information about how to obtain Oracle documentation, code, and My Oracle Support Notes.

We also describe setting up Red Hat Enterprise Linux Servers and configuring Security Enhanced Linux (SELinux) and SUSE Linux Enterprise Servers.

This chapter includes the following topics:

- ▶ 2.1, “Obtaining Oracle documentation, Oracle code, and My Oracle Support notes” on page 6
- ▶ 2.2, “Red Hat Enterprise Linux and SUSE Linux Enterprise Server setup” on page 8
- ▶ 2.3, “Red Hat Enterprise Linux Server 6 or 7 Specific Setup” on page 16
- ▶ 2.4, “SUSE Linux Enterprise Server specific setup” on page 17
- ▶ 2.5, “Installation” on page 20

2.1 Obtaining Oracle documentation, Oracle code, and My Oracle Support notes

This section includes the following topics:

- ▶ “Master Note of Linux OS Requirements for Database Server”
- ▶ “Pre-installation requirements documentation”
- ▶ “Obtaining code for Oracle 12cR1 on IBM LinuxONE” on page 7

2.1.1 Master Note of Linux OS Requirements for Database Server

The Master Note is intended to provide an index and references to the most frequently used My Oracle Support articles concerning Linux OS Requirements for Oracle Database Software Installation.

For more information, see [My Oracle Support Doc ID 851598.1](#), which is available by logging in to your Oracle Support account.

Figure 2-1 shows an example of the latest Linux distributions that were certified on LinuxONE.

```
OL7/RHEL7 zLinux:  
9.2.0 - Not certified, not supported, not planned.  
10.1.0 - Not certified, not supported, not planned.  
10.2.0 - Not certified, not supported, not planned.  
11.1.0 - Not certified, not supported, not planned.  
11.2.0 - Document 1967531.1 Requirements for Installing Oracle 11.2.0.4 RDBMS on RH7 on IBM: Linux on System z (s390x)  
12.1.0 - Document 2213265.1 Requirements for Installing Oracle 12c RDBMS on RH7 on IBM: Linux on System z (s390x)  
  
SLES11 zLinux:  
10.2.0 - Document 1290360.1 Requirements for Installing Oracle 10gR2 RDBMS on SLES 11 on zLinux (s390x)  
11.2.0 - Document 1290644.1 Requirements for Installing Oracle 11gR2 on SLES 11 on IBM: Linux on System z (s390x)  
12.1.0 - Document 1574414.1 Requirements for Installing Oracle Database 12c Release 1 on SLES 11 on IBM: Linux on System z (s390x)
```

Figure 2-1 Latest Linux distributions that are certified on LinuxONE

The Release Schedule of Current Database Releases is described in [Doc ID 742060.1](#), which is available by logging in to your Oracle Support account.

Oracle delivered the next Oracle database version 12.2.0.1 in the second quarter of calendar year 2017 (Q2CY2017) on IBM LinuxONE.

Note: At time of this writing, the SLES12 distribution is not certified for Oracle database.

2.1.2 Pre-installation requirements documentation

For more information about the pre-installation phase, see the Oracle Database (RDBMS) on UNIX, IBM AIX®, HP-UX, Linux, Mac OS X, Solaris, Tru64 Unix Operating Systems Installation, and Configuration Requirements Quick Reference (8.0.5 to 12.1) (Doc ID 169706.1), which is available at the [Oracle Support website](#).

For more information about Oracle Database 12cR1, see [Doc ID 1587357.1](#).

2.1.3 Obtaining code for Oracle 12cR1 on IBM LinuxONE

For more information about Oracle documentation that is used to obtain the code, see [Doc ID 1290644.1](#).

For more information about more recent Oracle Database code for Oracle Database 12cR1, Install the latest patchset version of 12.1 software (for example, Version 12.1.0.2), which is available under Patch 17694377 or from the [Oracle Technical Network website](#).

Patchsets are full releases and you do not need to install Oracle 12.1.0.1 first.

2.1.4 Oracle database pre-installation tasks in Oracle online Documentation

The Oracle Documentation website describes the tasks that must be completed before the Oracle Universal Installer is started. Specific information about IBM LinuxONE is included in the [Oracle Database Preinstallation Tasks](#) document (particularly bullet points 6 and 9).

2.1.5 Linux Large Pages and Oracle Databases

It is recommended for performance and availability reasons to implement Linux large pages for Oracle databases running on IBM LinuxONE systems. Linux large pages are particularly beneficial for systems where the database's Oracle SGA is greater than 10 GB or if there are many DB connections greater than 50.

Complete the following steps to implement large pages:

1. Each database that is planned for Linux large pages cannot utilize Automatic Memory Management (AMM) by setting the MEMORY_MANAGEMENT parameter. It is recommended to utilize Automatic Shared Memory Management (ASMM) by setting the SGA_TARGET and PGA_AGGREGATE_TARGET Oracle parameters.
2. The Oracle use_large_pages parameter can be set to true, false, or only. If you have a LinuxONE system with one large database require large pages, and other smaller databases that do not require large pages, you can set the larger SGA database to "only" and the smaller databases to "false". The default setting for use_large_pages is "true".
3. At the Linux level, it is recommended to set the /etc/security/limits.conf to unlimited to allow for changes to the Oracle SGA/Linux large page values dynamically, as shown in the following example:
 - soft memlock unlimited
 - hard memlock unlimited
4. Set or update the following parameters in the /etc/sysctl.conf file:
 - vm.nr_hugepages = ((sum of all large page SGA's)* 1024) + 16 = N
 - vm.hugetlb_shm_group = <Linux group number from /etc/group>

Example:

```
#cat /etc/group | grep oracle
dba:x:54322:oracle,grid
#((SGA = 119GB)*1024) + 16 ' (119 * 1024) + 16 = 121872
vm.nr_hugepages = 121872
vm.hugetlb_shm_group = 54322
```

5. Restart your Linux Image and restart Oracle for the changes to take effect.
6. Review your Oracle database's alert log to verify the database was started with large pages enabled.

2.2 Red Hat Enterprise Linux and SUSE Linux Enterprise Server setup

This section describes the common steps that are needed to implement an Oracle Single Instance Database on SUSE Enterprise Server or Red Hat Enterprise Linux on LinuxONE.

For more information about GRID or RAC implementations, see *Experiences with Oracle Database 12c Release 1 on Linux on System z*, SG24-8159.

2.2.1 Important information

In this section, the following pre-installation verification information is provided to give you a basic understanding before starting an Oracle Database installation on LinuxONE:

- ▶ Graphical user interface (GUI)

A GUI is required to run the Oracle Installers interactively. We use the Linux vncserver on LinuxONE to start an X server and a VNC viewer client to establish a terminal emulator xterm session.

To use GUIs, such as the Oracle Universal Installer (OUI), configuration assistants, and Oracle Enterprise Manager, set the display to a system with X Window System server packages.

In our environment, we start a vncserver on LinuxONE, and use a VNC viewer client to establish the connection with the vncserver.

- ▶ Linux minimal installation

A minimal Linux installation lacks many RPMs that are required for database installation. Therefore, you must use an RPM package for Linux release to install the required packages.

- ▶ ASM

To use Oracle Automatic Storage Management (ASM), you must first install Oracle Grid Infrastructure for a stand-alone server before you install and create the database.

To provide persistence of the storage devices across reboots, ASM on LinuxONE requires specific configuration of udev rules. To use Oracle ASM on LinuxONE, you must first install the Oracle Grid Infrastructure for a stand-alone server before you install and create the database.

For Red Hat, see *How to Manually Configure Disk Storage devices for use with Oracle ASM 11.2 and 12.1 on IBM: Linux on System z under RedHat* (Doc ID 1377392.1).

For SLES, see *How to Manually Configure Disk Storage devices for use with Oracle ASM 11.2 and 12.1 on IBM: Linux on System z under SLES* (Doc ID 1350008.1).

Although the SUSE **orarun** RPM package can be used to configure the user or groups, ulimits, and kernel parms automatically, our experiences are based on manually configuring these settings based on guidance from the Oracle Database Quick Installation Guide 12c Release 1 (12.1) for IBM: Linux on System z®, E18443. Changes in the recommended Oracle RPMs and kernel parameters can occur between release levels; therefore, review the latest Oracle support Getting Started notes for the latest updates.

- ▶ Large pages

Use of large (1 M) pages is recommend and requires specific configuration of the system.

2.2.2 Pre-installation checking

The general pre-installation checking tasks that you must perform after ensuring a GUI is established are described in this section.

General requirements and information for Oracle Database on LinuxONE

To obtain general LinuxONE information from a Linux session, run the following commands, ensuring that the system was started with a run level 3 or 5:

- ▶ Check information about the CPU that is defined in LinuxONE:
13oradb2:~ head -2 /proc/cpuinfo
- ▶ Check information about the RAM that is defined in LinuxONE:
13oradb2:~ # grep MemTotal /proc/meminfo

3oradb2:~ # grep SwapTotal /proc/meminfo
- ▶ Check the amount of space that is available in the /tmp directory:
13oradb2:~ # df -h /tmp
- ▶ Check the amount of free disk space on the system:
13oradb2:~ # df -h
- ▶ Check the amount of free RAM and disk swap space on the system:
13oradb2:~ # free
- ▶ Check whether the system architecture can run the software:
13oradb2:~ # uname -m

This command should return: s390x

Server minimum requirements

The following minimum requirements for the server must be met:

- ▶ Disk Space Requirements for Oracle 12cR1 on IBM LinuxONE
 - Ensure that the system meets the disk space requirements for software files:
 - Enterprise Edition database: 5.5 GB.
 - Grid Infrastructure: 3.5 GB
- ▶ Recommended memory
 - A total of 4 GB of virtual memory is recommended for Oracle 12cR1 installations of Oracle Database, including the Grid Infrastructure requirements.

Note: Although 2 GB is the minimum amount of virtual memory that is required, the Oracle OPatch utility requires 3072 MB (3 GB) for Oracle patching.

- ▶ Swap

Swap disk space is proportional to the system's physical memory as listed in Table 2-1.

Table 2-1 Swap size recommendation

RAM	Swap space
1 - 8 GB	Twice the size of RAM
8 - 32 GB	Equal to the size of RAM

RAM	Swap space
More than 32 GB	32 GB

More disk space (on a file system or an Oracle ASM disk group) is required for the fast recovery area.

- ▶ Temporary directory
1 GB of space in the /tmp directory on LinuxONE systems.
- ▶ Linux kernel parameters

Verify that you have the required operating system kernel and packages installed.

For more information about kernel requirements, [see the online version](#).

An example of kernel parameters is listed in Table 2-2.

Table 2-2 Kernel parameters

Variable	Example value	File
shmall: Percent of the size of physical memory in pages. If the server supports multiple databases, or uses a large SGA, set this parameter to a value that is equal to the total amount of shared memory in 4 K pages that the system can use at one time. For most systems, this value is the value 2097152. For more information, see My Oracle Support note 301830.1	2097152	/proc/sys/kernel/shmall
shmmmax: Half the size of physical memory in bytes. For more information, see My Oracle Support note 567506.1	2147483648	/proc/sys/kernel/shmmmax
semmsl	kernel.sem=250 32000 100 128	/proc/sys/kernel/sem
semmns		/proc/sys/kernel/sem
semopm		/proc/sys/kernel/sem
semnni		/proc/sys/kernel/sem
shmmni	4096	/proc/sys/kernel/shmmni
panic_on_oops setting is the value that allows <i>n</i> seconds delay before a node eviction/reboot.	1	/proc/sys/kernel/panic_on_oops
file-max 512 x number of processes	6815744 for 13312 processes	/proc/sys/fs/file-max
aio-max-nr: This value limits concurrent outstanding requests and should be set to avoid I/O subsystem failure.	3145728	/proc/sys/fs/aio-max-nr
net.ipv4.ip_local_port_range	9000 65500	/proc/sys/net/ipv4/ip_local_port_range
rmem_default	262144	/proc/sys/net/core/rmem_default
rmem_max	4194304	/proc/sys/net/core/rmem_max

Variable	Example value	File
wmem_default	262144	/proc/sys/net/core/wmem_default
wmem_max	1048576	/proc/sys/net/core/wmem_max

After kernel parameter values are updated, validate kernel parameters by using the following **sysctl** command:

```
13oradb2:~ # /sbin/sysctl -p
```

User login security and limits configuration

The following settings must be verified before an installation is performed:

- ▶ Limits.conf

If necessary, update the resource limits in the `/etc/security/limits.conf` configuration file for the installation owner.

Assuming that the “oracle” Linux user performs the installation, add the following settings to `/etc/security/limits.conf`:

```
oracle soft nproc 2047
oracle hard nproc 16384
oracle soft nofile 1024
oracle hard nofile 65536
oracle soft stack 10240
oracle hard stack 10240
```

If you are planning to use large pages, as recommended, add the following settings to `/etc/security/limits.conf`:

```
oracle soft memlock unlimited
oracle hard memlock unlimited
```

If GRID is used, define the following same limits for the GRID Linux user:

```
grid soft nproc 2047
```

Note: When the `limits.conf` file is changed, these changes take effect immediately. However, if the “oracle” users are logged in, these changes do not take effect until you log out these users and log them back in. You must log these users out and back in before you use these accounts for installation.

- ▶ Oracle user shell limits

Check the current shell limits and raise them if necessary to required values.

In oracle Linux user home, update the profile or bashrc file to define oracle user limits value, as shown in Example 2-1.

Example 2-1 Oracle user limits

```
if [ $USER = "oracle" ]; then
    if [ $SHELL = "/bin/ksh" ]; then
        ulimit -u 16384
        ulimit -n 65536
    else
```

```
        ulimit -u 16384 -n 65536
    fi
fi
```

► **Pluggable Authentication Modules (PAM)**

PAM is a system of libraries that manage user authentication tasks for applications. On Linux, external scheduler jobs require PAM. Oracle strongly recommends that you install the latest Linux-PAM library for your Linux distribution.

Edit the following line in the security services files in /etc/pam.d, and add the line at the bottom if needed. For example, add the following line in the sshd file:

```
session required pam_limits.so
```

Warning: Use care when the services files in pam.d are modified. Test the effect of the modification before closing the current work session by opening a new session.

Shared memory file system

Ensure that the /dev/shm mount area is of type tmpfs and is mounted with the following options:

- With RW and execute permissions set on it
- With noexec or nosuid not set on it

Use the following command to check the shared memory file system:

```
13oradb2:~ # cat /etc/fstab |grep tmpfs
```

The output for this command looks similar to following example:

```
tmpfs      /dev/shm      tmpfs      defaults      0 0
```

If needed, change the mount settings. As the root user, open the /etc/fstab file with a text editor and modify the tmpfs line. If you are planning to use Automatic Memory Management (AMM), you should set the tmpfs file system size to the sum of all the MEMORY_TARGET on the system, as shown in the following example:

```
tmpfs      /dev/shm      tmpfs      rw,exec,size=30G 0 0
```

LinuxONE can use large pages of 1 M in size with Oracle. The Oracle MEMORY_TARGET parameter is not eligible for large pages; instead, SGA_TARGET should be used for the Oracle memory setting when Linux large pages are used.

Host name

Use the ping command to ensure that the computer host name is resolvable, as shown in Example 2-2.

Example 2-2 Example of the ping command and its output

```
13oradb2:/etc # ping 13oradb0.mop.fr.ibm.com
PING 13oradb0.mop.fr.ibm.com (10.3.58.144) 56(84) bytes of data.
64 bytes from 13oradb0.mop.fr.ibm.com (10.3.58.144): icmp_seq=1 ttl=64 time=0.290
ms
--- 13oradb0.mop.fr.ibm.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 999ms
rtt min/avg/max/mdev = 0.090/0.190/0.290/0.100 ms
```

If needed, add the Fully Qualified Domain Name to the HOSTNAME file, by using the **hostname** command. Example 2-3 demonstrates showing the hostname command and its results.

Example 2-3 Show the hostname

```
13oradb2:~ # hostname  
13oradb2.mop.fr.ibm.com
```

Example 2-4 shows the command and results of displaying the contents of the HOSTNAME file.

Example 2-4 Show the contents of the HOSTNAME file

```
13oradb2:~ # cat /etc/HOSTNAME  
13oradb2.mop.fr.ibm.com
```

To update the hostname, enter the fully qualified hostname by using the **hostname** command, as shown in Example 2-5.

Example 2-5 Updating the hostname

```
113oradb2:~ # hostname  
13oradb2  
3oradb2:~ # hostname 13oradb2.mop.fr.ibm.com  
3oradb2:~ # hostname  
13oradb2.mop.fr.ibm.com
```

Creating the database installation owner user

The installation process requires at least an Oracle database installation owner (oracle), an Oracle Inventory group (oinstall), and an Oracle administrative privileges group (dba).

Example 2-6 shows the commands that are used to create the oracle user and groups.

Example 2-6 Creating the oracle user and groups

```
13oradb2:~ # /usr/sbin/groupadd oinstall -g 1001  
13oradb2:~ # /usr/sbin/groupadd dba -g 1002  
13oradb2:~ # /usr/sbin/useradd -m -g oinstall -G dba oracle  
13oradb2:~ # echo "oracle:newPassw0rd" | /usr/sbin/chpasswd
```

Storage options for Oracle database

In this section, we describe the following disk storage configuration options that are available for database files on LinuxONE:

- ▶ Open Storage FCP/SCSI LUNs
- ▶ IBM Extended Count Key Data (ECKD™) DASD with HyperPAV

To configure disk storage for LinuxONE, assign the LUNs (with multiple paths) to Linux to ensure high availability and better performance by spreading the I/O workload across more channel paths/host bus adapters (HBAs).

Configure a UDEV rule that ensures device persistence and set the following file permissions for Oracle:

- ▶ FCP/SCSI: /etc/udev/rules.d/12-dm-permissions.rules file
- ▶ ECKD/DASD: /etc/udev/rules.d/99-udev-oracle.rules

A restart is required for modifications of UDEV rules to take effect. For Red Hat 6+ / SLES 11+, you can run the command **udevadm trigger** for any udev changes to take effect. This change can be done dynamically.

Then, run the **ls** commands to confirm that the file permissions are set correctly (based on the devices configured in your UDEV rules), as shown in the following example:

```
# ls -la /dev/dasde1
```

Running this command results in the following output:

```
brw-rw---- 1 oracle oinstall 94, 5 Sep 25 08:59 /dev/dasde1
```

- ▶ **File systems and Shared Storage**

For a stand-alone database on Linux for LinuxONE, database files and recovery files are supported on Linux file systems and Oracle ASM.

When using database with GRID or RAC, ASM provides shared storage and is certified on LinuxONE.

As described in *Oracle Supported and Recommended File Systems on Linux* (Doc ID 236826.1), the recommended file systems when not utilizing Oracle ASM for database files are:

- For SUSE 12 SP1+ systems, use xfs (Requires Oracle DB 12.1.0.2+)
- For Suse 11 SP3+ systems, use xfs
- For Red Hat 6 systems, use ext4
- For Red Hat 7 systems, use xfs

Required software directories

Identify or create the following directories that are needed by the Oracle software installer. The examples in this section use /u01 for the mount point directory:

- ▶ **Oracle Base Directory**

The following directory is used:

```
ORACLE_BASE = /mount_point/app/software_owner
```

In this installation example, we created a 20GB Logical Volume to host Oracle software and data.

Where *software_owner* is the operating system user name of the software owner that is installing the Oracle software; for example, oracle or grid.

- ▶ **Oracle Inventory Directory**

This directory stores an inventory of all software that is installed on the system. It is required and shared by all Oracle software installations on a single system. If Oracle Inventory paths exist, the Oracle Universal Installer continues to use that Oracle Inventory.

The Oracle Inventory path is in /var/opt/oracle/oraInst.loc.

Example 2-7 on page 15 shows the command that is used to check the contents and the existence of the oraInst.loc file with the results displaying the contents of the file.

Example 2-7 Examining the oraInst.loc

```
13oradb0:~ # cat /var/opt/oracle/oraInst.loc
inventory_loc=/oraInventory
inst_group=oinstall
```

- ▶ Oracle Home Directory

This directory is where the Oracle software product is installed:

```
ORACLE_HOME=$ORACLE_BASE/oracle_base/product/12.1.0/dbhome_1
```

Identifying or creating an Oracle Base Directory

For more information about if Oracle ASM is used, see the Oracle ASM documentation, *How to Manually Configure Disk Storage devices for use with Oracle ASM 11.2 and 12.1 on IBM: Linux on System z under SLES* (Doc ID 1350008.1).

For more information about Red Hat systems, see, *How to Manually Configure Disk Storage devices for use with Oracle ASM 11.2 on IBM: Linux on System z under RedHat 5* (Doc ID 1351746.1).

If ASM is not used, create a Linux file system by using the following Yast or Linux commands:

- ▶ Create physical volumes, a volume group, and add physical volumes to it and create a Logical Volume with stripping. Create an ext3 file system.
- ▶ Create the following mount point:

```
13oradb2:~ # mkdir /u01
13oradb2:~ # chown -R oracle:oinstall /u01
13oradb2:~ # chmod -R 775 /u01
```

- ▶ Mount the logical volume on ORACLE_BASE=/u01, as shown in the following example:

```
13oradb2:~ # mount -t ext3 /dev/mapper/vg2181-lv2181 /u01
```

For resiliency after restart, insert the mount statement into /etc/fstab.

Setting the disk I/O scheduler

The following I/O schedulers (also known as *elevators*) are available:

- ▶ Noop: No operation; only last request merging.
- ▶ Deadline: A latency-oriented I/O scheduler. The algorithm preferably serves reads before writes.
- ▶ Complete fair queuing (CFQ): All users of a particular drive can execute approximately the same number of I/O requests over a specific time.

The default scheduler in the current distributions is Deadline. Do not change the default scheduler or default settings without reason. Changing these settings can reduce throughput or increase processor consumption.

For more information about which elevator is the current default, run the command that is shown in Example 2-8. The selected scheduler is listed in brackets.

Example 2-8 Checking the scheduler

```
13oradb2:/sys/devices # cat ./css0/0.0.0007/0.0.0100/block/dasdb/queue/scheduler
noop [deadline] cfq
```

2.3 Red Hat Enterprise Linux Server 6 or 7 Specific Setup

This section describes the other steps that are needed to install Oracle Single Instance Database on Red Hat Enterprise Linux for Red Hat 6 or 7. These steps are in addition to the generic Linux setup steps that are described in 2.2, “Red Hat Enterprise Linux and SUSE Linux Enterprise Server setup” on page 8 that apply for Red Hat Linux systems on LinuxONE only.

2.3.1 Important information

In this section, we describe some information important to note before installation.

Security-Enhanced Linux

Security-Enhanced Linux (SELinux) is a Linux kernel security module that provides a mechanism for supporting access control security policies for security of Red Hat operating systems.

By default, Red Hat Enterprise Linux server versions 6 and 7 are installed with SELinux in **enforcing** mode (meaning that the SELinux policy is in effect and it enforces access denials and audits them). This configuration is acceptable for the 12cR1 installation process for Red Hat Enterprise Linux server version 6 and 7.

2.3.2 Pre-installation checking

This section describes more pre-installation checking that you must perform after you ensure that a GUI is established.

Operating system requirements for IBM LinuxONE

The following operating system requirements must be met when IBM LinuxONE is used:

- ▶ SSH Requirement: OpenSSH is the required SSH software.
- ▶ RHEL 7 servers must be running Red Hat kernel 3.10.0-229.el7 (s390x) or higher. (Red Hat 7.2+ is recommended).
- ▶ RHEL 6 Kernel 2.6.32-279.el6 RHEL6 UPDATE3 is the minimum level for Oracle Database 12c Release 1 Version 12.1.0.2.

Oracle RPM Checker with Red Hat Enterprise Linux Server on LinuxONE

In this section, we describe how to use the Oracle RPM Checker utility to verify that the following required Linux packages are installed on the operating system before starting the Oracle Database or Oracle Grid Infrastructure installation. The RPM Checker does not install any missing package:

- ▶ Checking missing RPMs

Download the appropriate Oracle RPM Checker utility from one of the following My Oracle Support websites based on Oracle Database 12c Release 1 Version 12.1.0.2 and the Red Hat Enterprise Linux Server distribution level (RHEL 6 / 7):

- [RedHat 6 12.1.0.2](#)
- [RedHat 7 12.1.0.2](#)

After you download the RPM checker on LinuxONE, decompress the RPM and install it as root. To assess the missing RPMs, use one of the following commands (depending on your distribution level):

```
yum install ora-val-rpm-RH7-DB-12.1.0.2-1.s390x.rpm
```

Running the command automatically pulls in the required RPMs if your yum repository is set up. If you do not have your yum repository set up, you can install the RPM dependency checker by using the following command (use the command corresponding to your distribution level):

```
rpm -ivh ora-val-rpm-RH7-DB-12.1.0.2-1.s390x.rpm
```

Figure 2-2 shows the result of running the command to install the RPM dependency checker.

```
[root@oralab01 oracode]# rpm -ivh ./ora-val-rpm-RH7-DB-12.1.0.2-1.s390x.rpm
Preparing...                                         #####
Updating / installing...
1:ora-val-rpm-RH7-DB-12.1.0.2-1      ##### [100%
*****
*   Validation complete - please install any missing rpms          *
*   The following output should display both (s390) - 31-bit and    *
*   (s390x) 64-bit rpms - Please provide the output to Oracle       *
*   Support If you are still encountering problems.                 *
*****
Found      glibc (s390)
Found      glibc (s390x)
Found      libaio (s390)
Found      libaio (s390x)
Found      libaio-devel (s390)
Found      libaio-devel (s390x)
Found      libXi (s390)
Found      libXi (s390x)
Found      libXtst (s390)
Found      libXtst (s390x)
Found      glibc-devel (s390)
Found      glibc-devel (s390x)
Found      libgcc (s390)
Found      libgcc (s390x)
Found      libstdc++ (s390)
Found      libstdc++ (s390x)
Found      libstdc++-devel (s390)
Found      libstdc++-devel (s390x)
```

Figure 2-2 Red Hat rpm checker for 12.1.0.2 Red Hat 7 Installation

2.4 SUSE Linux Enterprise Server specific setup

This section describes the steps that are used to set up Oracle Single Instance Database on SUSE Enterprise Server on LinuxONE.

These steps are in addition to the generic Linux setup steps that are described in 2.2, “Red Hat Enterprise Linux and SUSE Linux Enterprise Server setup” on page 8 and apply to SUSE Linux Enterprise systems on LinuxONE only.

2.4.1 Important information

Although the SUSE `orarun` RPM package can be used to configure the user or groups, ulimits, and kernel parms automatically, our experiences are based on manually configuring these settings based on guidance from *Oracle Database Quick Installation Guide 12c Release 1 (12.1) for IBM Linux on System z*, E18443. Changes in the recommended Oracle RPMs and kernel parameters can occur between release levels; therefore, review the latest Oracle support Getting Started notes for updates.

2.4.2 Pre-installation checking

This section describes other pre-installation checking that you must perform after ensuring that a GUI is established.

Operating system requirements for IBM LinuxONE

The following operating system requirements must be met when IBM LinuxONE is used:

- ▶ SSH Requirement: OpenSSH is the required SSH software.
- ▶ SUSE Linux Enterprise Server: SUSE Linux Enterprise Server 11 SP2: 3.0.13-0.27-default s390x or later.

Oracle RPM Checker with SUSE Linux Enterprise Server on LinuxONE

Use the Oracle RPM Checker utility to verify that the following required Linux packages are installed on the operating system before starting the Oracle Database or Oracle Grid Infrastructure installation. The RPM checker does *not* install any missing package:

- ▶ Check for missing RPMs.

Download the Oracle RPM Checker utility from the link that is provided in the My Oracle Support note 1574412.1.

After downloading the utility to the Suse Enterprise Linux Server on LinuxONE, decompress the RPM and install it as root. To assess the missing RPMs, use the following command:

```
13oradb2:~ # rpm -ivh ora-val-rpm-S11-DB-12.1.0.1-1.s390x.rpm
```

The output of the command is shown in Figure 2-3.

```
13oradb2:/mnt/oracle/oracle.12c # rpm -ivh /mnt/oracle/oracle.12c/ora-val-rpm-S11-DB-12.1.0.1-1.s390x.rpm
error: Failed dependencies:
      ksh >= 93t-9.9.8 is needed by ora-val-rpm-S11-DB-12.1.0.1-1.s390x
      libaio-devel >= 0.3.109-0.1.46 is needed by ora-val-rpm-S11-DB-12.1.0.1-1.s390x
      libaio-32bit >= 0.3.109-0.1.46 is needed by ora-val-rpm-S11-DB-12.1.0.1-1.s390x
      libaio-devel-32bit >= 0.3.109-0.1.46 is needed by ora-val-rpm-S11-DB-12.1.0.1-1.s390x
      libstdc++-33 >= 3.3.3-11.9 is needed by ora-val-rpm-S11-DB-12.1.0.1-1.s390x
      libstdc++-33-32bit >= 3.3.3-11.9 is needed by ora-val-rpm-S11-DB-12.1.0.1-1.s390x
      libstdc++-43-devel-32bit >= 4.3.4_20091019-0.22.17 is needed by ora-val-rpm-S11-DB-12.1.0.1-1.s390x
      libgcc46 >= 4.6.1_20110701-0.13.9 is needed by ora-val-rpm-S11-DB-12.1.0.1-1.s390x
      sysstat >= 8.1.5-7.32.1 is needed by ora-val-rpm-S11-DB-12.1.0.1-1.s390x
      glibc-devel >= 2.11.3-17.31.1 is needed by ora-val-rpm-S11-DB-12.1.0.1-1.s390x
      linux-kernel-headers >= 2.6.32-1.4.13 is needed by ora-val-rpm-S11-DB-12.1.0.1-1.s390x
      gcc >= 4.3-62.198 is needed by ora-val-rpm-S11-DB-12.1.0.1-1.s390x
      gcc43 >= 4.3.4_20091019-0.7.35 is needed by ora-val-rpm-S11-DB-12.1.0.1-1.s390x
      glibc-devel-32bit >= 2.11.3-17.31.1 is needed by ora-val-rpm-S11-DB-12.1.0.1-1.s390x
      gcc-32bit >= 4.3-62.198 is needed by ora-val-rpm-S11-DB-12.1.0.1-1.s390x
      gcc43-32bit >= 4.3.4_20091019-0.7.35 is needed by ora-val-rpm-S11-DB-12.1.0.1-1.s390x
      libomp43-32bit >= 4.3.4_20091019-0.7.35 is needed by ora-val-rpm-S11-DB-12.1.0.1-1.s390x
      libstdc++-43-devel >= 4.3.4_20091019-0.22.17 is needed by ora-val-rpm-S11-DB-12.1.0.1-1.s390x
      gcc-c++ >= 4.3-62.198 is needed by ora-val-rpm-S11-DB-12.1.0.1-1.s390x
      gcc43-c++ >= 4.3.4_20091019-0.7.35 is needed by ora-val-rpm-S11-DB-12.1.0.1-1.s390x
      libstdc++-43-devel-32bit >= 4.3.4_20091019-0.7.35 is needed by ora-val-rpm-S11-DB-12.1.0.1-1.s390x
      libstdc++-46 >= 4.6.1_20110701-0.13.9 is needed by ora-val-rpm-S11-DB-12.1.0.1-1.s390x
      libstdc++-46-32bit >= 4.6.1_20110701-0.13.9 is needed by ora-val-rpm-S11-DB-12.1.0.1-1.s390x
      libstdc++-devel >= 4.3-62.198 is needed by ora-val-rpm-S11-DB-12.1.0.1-1.s390x
      libcap1 >= 1.10-6.10 is needed by ora-val-rpm-S11-DB-12.1.0.1-1.s390x
13oradb2:/mnt/oracle/oracle.12c #
```

Figure 2-3 RPM checker output showing missing RPMs

In SUSE Linux Enterprise Server, you can also use the zypper installation utility by executing the following command:

```
13oradb2:~ # zypper install ora-val-rpm-S11-DB-12.1.0.1-1.s390x.rpm
```

The missing RPMs that are shown by the tool must be installed before the Oracle Installer is used.

With SUSE Linux Enterprise Server, software can be added by using the YAST interface or from the Linux command line (**rpm -i** or **zypper**). Figure 2-4 shows the installation of a package using zypper.

```
13oradb2:/etc/zypp/repos.d # zypper install libaio-devel
Loading repository data...
Reading installed packages...
Resolving package dependencies...

The following 3 NEW packages are going to be installed:
  glibc-devel libaio-devel linux-glibc-devel

3 new packages to install.
Overall download size: 1.6 MiB. Already cached: 0 B. After the operation, additional 9.3 MiB will be used.
Continue? [y/n? show all options] (y): y
Retrieving package linux-glibc-devel-3.12-6.54.noarch
Retrieving package glibc-devel-2.19-31.9.s390x
Retrieving package libaio-devel-0.3.109-17.15.s390x
Checking for file conflicts: .....(1/3) Installing: linux-glibc-devel-3.12-6.54 .....(done)
(2/3) Installing: glibc-devel-2.19-31.9 .....(done)
(3/3) Installing: libaio-devel-0.3.109-17.15 .....(done)
13oradb2:/etc/zypp/repos.d #
```

Figure 2-4 Package installation using zypper

Note: Repositories for zypper can be found in the `/etc/zypp/repos.d` file system.

- ▶ Automatically install all missing RPMs in a single shell command.

To provision LinuxONE Oracle instances and simplify automation, a shell can be used to check and install the missing packages.

Create a shell command file and paste the contents, as shown in Example 2-9 on page 20.

Example 2-9 Contents of shell command file

```
rpm -ivh /mnt/oracle/oracle.12c/ora-val-rpm-S11-DB-12.1.0.1-1.s390x.rpm  
2>/tmp/tempo;  
var=$(wc -l /tmp/tempo | awk '{print $1}');  
tail -$((var - 1)) /tmp/tempo | awk '{print "zypper install " $1}'  
>/tmp/irpm.sh;sh /tmp/irpm.sh;  
rm -f /tmp/tempo;  
cat /tmp/irpm.sh;  
rm -f /tmp/irpm.sh ;  
rpm -ivh /mnt/oracle/oracle.12c/ora-val-rpm-S11-DB-12.1.0.1-1.s390x.rpm
```

This shell first identifies the missing RPMs and writes their name in a temporary file. Then, it starts the RPM installation by using the **zypper** command and removes temporary work files. Finally, it starts the RPM checker utility to ensure that all files were installed.

2.5 Installation

In this section, we describe the use of the OUI as a silent installation and an interactive installation. We then provide the steps to update the oracle user profile. Finally, we describe the steps to create the database.

2.5.1 Oracle Universal Installer

By using OUI, you can install software only, or install and create a database. Other installers are available, such as netca, to create the Oracle listener services, or dbca, to create databases. The Oracle Installers can be run interactively or in silent mode.

Silent installation

Complete the following steps to run a silent installation:

1. The silent mode is useful for automating and provisioning Oracle instances.
2. A response file contains all information that is required for a successful installation.
3. Samples of responses files are on the installation media. The response file can also be created during an interactive installation by clicking **Save Response File**.

The Silent Installation command is shown in the following example. It was started by the IBM Wave User Script Manager to provision Oracle Database as a Service:

```
13oradb2:~ # sudo -u oracle /media/database/runInstaller -silent -responseFile  
/tmp/silentInstallFile.rsp
```

Options, such as **-debug**, **-force**, and **-ignorePrereq** can be added to the command, as needed.

Interactive installation

Complete the following steps to run an interactive installation:

1. Run the Oracle Database Installation program through a VNC viewer by using the oracle user.
2. Mount the Installation DB under **/media** (for example), and go to the following database directory:
`cd /media/database/`

3. Start the interactive OUI as shown in the following example:

```
./runInstaller
```

4. Complete the required fields in each panel that are raised by the Installer. The panels that you see are listed in Table 2-3.

Table 2-3 Installation panels in order of appearance with their options

Panel	Comments and options
Configure security updates	Use this panel to enter your My Oracle Support account information if you want to receive the latest product information and security updates by using your My Oracle Support account.
Installation Options	Choose one of the following installation options: <ul style="list-style-type: none">▶ Create and configure a database▶ Install database software only▶ Upgrade an existing database
Grid Installation Options.	Select the following type of database installation: <ul style="list-style-type: none">▶ Single Instance▶ Oracle RAC▶ Oracle RAC One Node
Product Languages	Select your language
Database Edition	Only the Enterprise Edition is selectable on LinuxONE
Install Locations	Provide your Oracle base and software paths
Oracle Inventory	Location of the Oracle inventory
Operating System Groups	Specify the access privileges
Summary	A summary of the chosen configuration is displayed. This configuration can be saved in a response file by clicking Save Response File .

5. Click **Install**.

6. Continue with the installation panels that are listed in Table 2-4.

Table 2-4 Continuation of Installation panels

Panel	Comments and options
Product Installation	<ul style="list-style-type: none">▶ Prepare▶ Copy files▶ Link binaries▶ Setup▶ Setup Oracle Base▶ Execute Root Scripts. <p>The installer asks for running orainstRoot.sh and root.sh postinstallation scripts.</p>
Oracle Database configuration	(no customization needed)
Finish	Click Finish .

2.5.2 Updating the Oracle user profile

Following a successful database software installation, set \$ORACLE_HOME, and then set the \$PATH to include \$ORACLE_HOME/bin at the beginning of the \$PATH string, as shown in Example 2-10.

Example 2-10 Setting variables

```
export ORACLE_BASE=/u01  
export ORACLE_HOME=$ORACLE_BASE/product/12.1.0/dbhome_1  
export PATH=$ORACLE_HOME/bin:$PATH
```

2.5.3 Creating the database

Complete the following steps to create the database:

1. Log in with the Oracle user.
2. Start one of the following installers interactively or in silent mode:
 - netca installer to create an Oracle listener service
 - dbca installer to create a database
3. Add or uncomment ORACLE_SID in the Oracle user's profile, which is in the user's HOME directory using the **export ORACLE_SID=orc1** command.



Using Oracle Enterprise Manager Cloud Control Agent to manage Oracle Database 12c Release 1

This chapter describes how to deploy Oracle Enterprise Manager Cloud Control 13c Release 2 (13.2.0.0.0) agents on a LinuxONE environment to manage Oracle databases.

Enterprise Manager Cloud Control can monitor many databases that are running on multiple environments from a single console. In addition to databases monitoring, other applications, such as E-Business Suite, and support for monitoring certain non-Oracle products, for example, IBM WebSphere® Application Server, can be monitored from a single Cloud Control Console. Although the Enterprise Manager Cloud Control does not run on LinuxONE, the agents and the Oracle Database run on the LinuxONE guests and they communicate with the Enterprise Manager Cloud Control.

In this chapter, we share our experiences of installing Enterprise Manager Cloud Control on an x86 based Linux server and describe how the agents can be deployed from the server to monitor the Oracle databases on Linux that are running on LinuxONE. The agents can be deployed from the Enterprise Manager Cloud Control Console or by using the silent agent deployment option at the Linux guests. These procedures also are described in this chapter.

This chapter includes the following topics:

- ▶ 3.1, “Basic Enterprise Manager Cloud Control Architecture” on page 24
- ▶ 3.2, “Creating the Enterprise Manager Cloud Control Architectures” on page 25
- ▶ 3.3, “Updating the agent at the Enterprise Manager Cloud Control to monitor Oracle Databases on LinuxONE (online)” on page 29
- ▶ 3.4, “Updating the agent at the Enterprise Manager Cloud Control to monitor Oracle Databases on LinuxONE (offline)” on page 31
- ▶ 3.5, “Deploying the agents from the Cloud Control console” on page 35
- ▶ 3.6, “Deploying the agents in silent mode” on page 37
- ▶ 3.7, “Adding the databases for monitoring” on page 40

3.1 Basic Enterprise Manager Cloud Control Architecture

The Enterprise Manager Cloud Control Architecture features the following components:

- ▶ Oracle Management Services (OMS)

The OMS is a web-based application that coordinates with the management agents and plug-ins, stores the collected information in a repository, and renders the user interface for Enterprise Manager Cloud Control. The OMS is deployed to the Oracle Middleware home (Middleware home), and also installs Oracle WebLogic Server if it does not exist in the environment.

- ▶ Oracle Management Repository

When the Enterprise Manager Cloud Control is installed, the Management Repository is configured in an Oracle database. The Management Repository is where all of the information that is collected by the Management Agent is stored.

- ▶ Plug-ins

Plug-ins are deployed to the OMS and the Management Agent. The Management Agent, OMS, and the plug-ins work together to monitor all targets in the environment. By default, the plug-ins that are needed for Oracle Databases are installed automatically when the Enterprise Manager Cloud Control is installed.

- ▶ Oracle JVMD Engine (Java virtual machine Diagnostics)

Starting with Oracle Enterprise Manager 13c, one JVMD Engine is installed and configured by default on the OMS as part of the Oracle Fusion Middleware Plug-in deployment. JVMD Engine enables that ability to diagnose performance problems in Java applications in the production environment.

- ▶ Oracle BI Publisher

Oracle Business Intelligence (BI) Publisher is Oracle's reporting tool. Starting with Oracle Enterprise Manager 13c, Oracle BI Publisher is installed and configured by default on the OMS.

- ▶ Oracle Management Agents

The Management Agent is deployed to the hosts that are monitored by the Enterprise Manager Cloud Control. It works with the plug-ins to monitor the targets that are running on that managed host. When OMS is installed, a Management Agent that is named the Central Agent is installed by default and is used for monitoring the first OMS host, the first OMS, and the other targets that are running on the first OMS host. To monitor other hosts and targets that are running on those hosts, you must install a separate Standalone Management Agent on each of those hosts.

- ▶ Cloud Console

Cloud Console is the user interface and is the centralized location that is used to monitor and manage the systems and services.

Figure 3-1 shows the Enterprise Manager Cloud Control Architecture, which was originally shown in the Oracle Enterprise Manager Cloud Control Introduction 13c Release 2 (E74226-03).

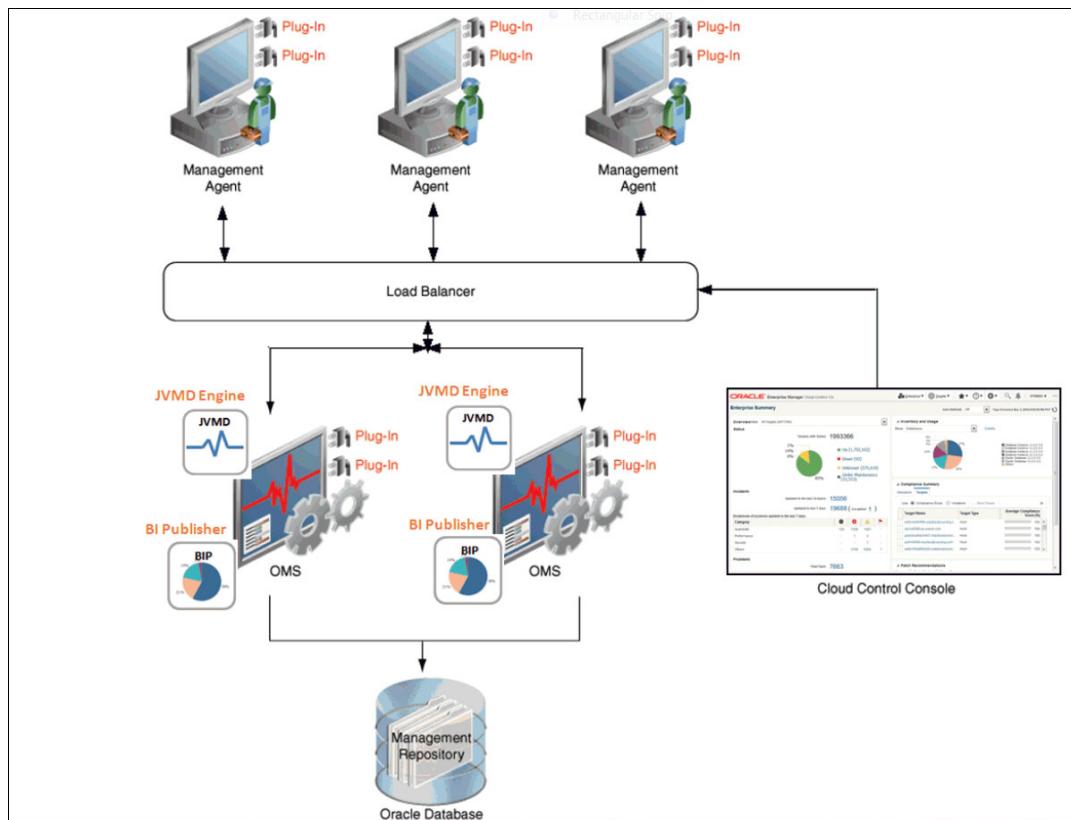


Figure 3-1 Cloud Control Architecture

3.2 Creating the Enterprise Manager Cloud Control Architectures

Enterprise Manager Cloud Control must be created on a supported architecture, this may be AIX or Linux on Intel.

Before you begin this task, see the following My Oracle support documents:

- ▶ *Oracle Enterprise Manager 13c Configuration Best Practices*, Doc ID 2139083.1
- ▶ *How to Install Enterprise Manager Cloud Control 12c on Linux*, Doc ID 1359176.1

The Enterprise Manager Cloud Control 13c and EM12c installation procedures are same. For more information, see *Experiences with Oracle Database 12c Release 1 on Linux on System z*, SG24-8159.

In our example environment, we created Oracle Enterprise Manager Cloud Control on an x86-64 SLES 11 sp2 Linux server.

For more information about the hardware and software requirements for the server architecture where the Cloud Control is going to be installed, see Oracle Enterprise Manager Cloud Control Basic Installation Guide 13c Release 2 (E73473-04).

Note: Although the installation procedure is beyond the scope of this book, we used the database templates for Enterprise Manager 13.2.0.0 on Linux x86-64 (376,876,746 bytes, cksum 3941973314).

The DB templates are for the 12.1.0.2 version of Oracle database 12.1.0.2 only and are available at the [Oracle Database Template website](#).

In the Linux server, we created the Oracle Enterprise Manager Cloud Control Architectures by completing the following steps:

1. Download the required installation files.
2. Start the installation in a graphical terminal, such as a VNC client.
3. Customize the Enterprise Manager Cloud Control during installation.
4. Run `allrootot.sh`.
5. Ensure that Enterprise Manager Cloud Control Console is accessible.

3.2.1 Downloading the installation files

Download the following Enterprise Manager Cloud Control 13c Release 2 (13.2.0.0.0) for Linux x86-64 files from the Oracle download site. Ensure that you are authorized to download the software:

- ▶ `em13200_linux64.bin` (554,607,894 bytes) (cksum - 2013200947)
- ▶ `em13200_linux64-2.zip` (2,118,110,455 bytes) (cksum - 1145259313)
- ▶ `em13200_linux64-3.zip` (553,563,556 bytes) (cksum - 2319677119)
- ▶ `em13200_linux64-4.zip` (2,084,299,941 bytes) (cksum - 4253735100)
- ▶ `em13200_linux64-5.zip` (2,128,070,240 bytes) (cksum - 4235966149)
- ▶ `em13200_linux64-6.zip` (474,441,637 bytes) (cksum - 2197985098)

Download all of the files (including .bin files) in same directory. There is need to decompress the downloaded software. As the installation user (in our case, as oracle user) who is installing the product, set the execute permission for the .bin file. For example: `chmod +x em13200_linux64.bin`. The directory structure that is shown in Figure 3-2 is created.

```
oracle@atstools100:/dload/oracode/emcode> ls -alF em*
-r--r--r-- 1 oracle oinstall 2118110455 Oct  2 11:33 em13200_linux64-2.zip
-r--r--r-- 1 oracle oinstall 553563556 Oct  2 11:34 em13200_linux64-3.zip
-r--r--r-- 1 oracle oinstall 2084299941 Oct  2 11:34 em13200_linux64-4.zip
-r--r--r-- 1 oracle oinstall 2128070240 Oct  2 11:35 em13200_linux64-5.zip
-r--r--r-- 1 oracle oinstall 474441637 Oct  2 11:35 em13200_linux64-6.zip
-r-xr-xr-x 1 oracle oinstall 554607894 Oct  2 11:35 em13200_linux64.bin*
oracle@atstools100:/dload/oracode/emcode>
```

Figure 3-2 Directory structure

3.2.2 Installing and configuring Enterprise Manager Cloud Control 13c

In general, the following operations are carried out by the installation wizard as part of the installation of an Enterprise Manager system:

- ▶ Install the following middleware components in the Middleware home (in our example, /d1/mw).
 - Java Development Kit (JDK) 1.7.0_80
 - Oracle WebLogic Server 12c Release 1 (12.1.3.0)
 - Oracle Management Service 13c Release 2
 - Oracle JRF 12c Release 1 (12.1.3.0), which includes the oracle common directory

- Oracle Web Tier 12c Release 1 (12.1.3.0)
- Oracle BI Publisher 12c Release 1 (12.1.3.0), which includes the /bi directory
- ▶ Install Oracle plug-ins (in the plug-ins directory):
 - Oracle Database plug-in
 - Oracle Fusion middleware plug-in
 - Oracle Exadata Plug-in
 - Oracle Cloud Framework Plug-in
 - Oracle System Infrastructure Plug-in
 - Any other plug-ins you choose to deploy
 - One JVMD Engine
- ▶ Install Oracle Management Agent 13c Release 2 (13.2.0.0.0) in the agent base directory that is specified during installation (outside the Middleware home; in our case, (/d1/agent)).
- ▶ Create an Oracle WebLogic domain that is called GCDomain, which is a default user account (weblogic) that is used as the administrative user, and a node manager account.
- ▶ Configure Oracle Management Service in the Instance Base location (gc_inst) in the Middleware home for storing all configuration details that are related to Oracle Management Service 13c (in our case, /d1/gc_inst).
- ▶ Configure Oracle Management Repository in the Oracle Database. In our case, the database instance “orcl” was created by using the database template that is offered by Oracle; therefore, this step was skipped.
- ▶ Configure the various installed components.

Complete the following steps:

1. The installer requires at least 10 GB of /tmp directory space. In our case, we created a directory with 10 GB space as /d1/tmp and passed that information as a parameter.
2. Start the installation wizard as the Oracle user in the directory location of the downloaded installation files from a graphical client, such as VNC, as shown in the following example:

```
./em13200_linux64.bin -J-Djava.io.tmpdir=/d1/tmp
```
3. The installation starts. You are prompted to specify the Oracle Support Credentials; in our example, we skipped the Software Updates option.
4. During the prerequisite checks, the installer verifies the requirements for installation. In our example, we ensured that all the steps are successful.
5. In the Installation Types window, select **Create a new Enterprise System Manager System** and the **Advanced** option.
6. In the Installation Details window, specify the middleware home location, agent base directory location, and the host name where the installation is done.
7. In the Select Plug-ins window, the mandatory plug-ins are automatically disabled. You can select any other plug-ins that are needed. In our case, we used the default selections.
8. In the WebLogic Server Configuration Details window, specify the required information for the Oracle WebLogic Server configuration.
9. In the Database Connection Details window, specify the required information for the installed database connection in the server.

The Enterprise Manager Shared Location details are populated in the Enterprise Manager Shared Location Details window.

In our example, we accepted the default port values in the Port Configuration Details.

10. Review the configuration values in the next window and click **Install**.

11. The installation starts and you can see the progress in the Installation Progress Details window. When the installation and configuration is complete, run `allroot.sh` as the root user. For more information, see the [Postinstallation Configuration Tasks website](#).

In our example, the Enterprise Manager Cloud Control configuration and installation took approximately 100 minutes to complete. The status is displayed at the end of the process. The Finish window shows the URL and port number access information for Enterprise Manager Cloud Control and the Admin Server.

12. When you use the Enterprise Manager Cloud Control URL for the first time, you are prompted to trust, certify, and add the exception. Then, the Enterprise Manager Cloud Control window opens, as shown in Figure 3-3. Now, you can log on as sysman by using the assigned password.

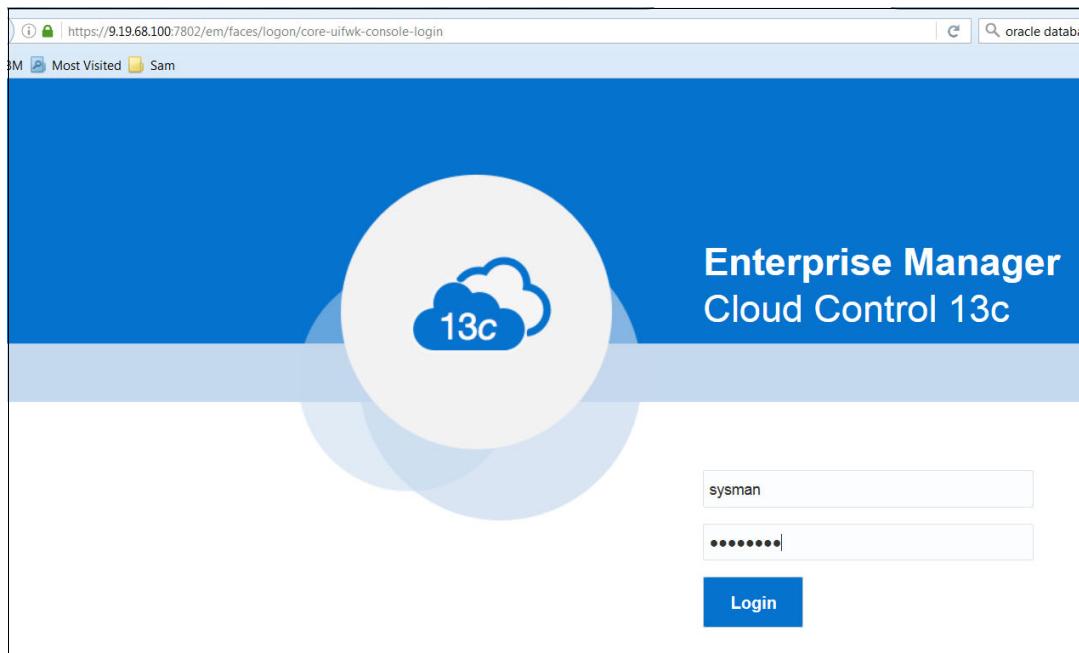


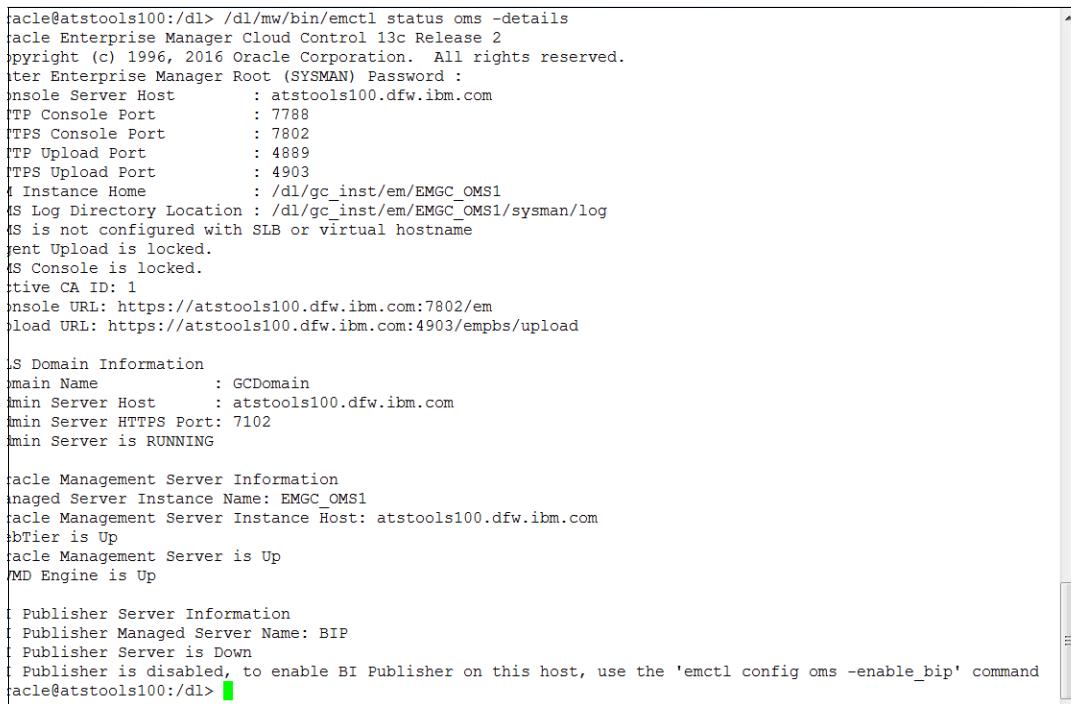
Figure 3-3 Enterprise Manager Cloud Control 13c

13. Accept the license requirement, which completes the Enterprise Manager Cloud Control 13c Server installation process.

Also, at the Linux Server, you can use the `emctl` command to display the OMS status, as shown in the following example:

```
/d1/mw/bin/emctl status oms -details
```

The sample output is shown in Figure 3-4.



```
oracle@atstools100:/dl> /dl/mw/bin/emctl status oms -details
oracle Enterprise Manager Cloud Control 13c Release 2
Copyright (c) 1996, 2016 Oracle Corporation. All rights reserved.
Enter Enterprise Manager Root (SYSMAN) Password :
Console Server Host      : atstools100.dfw.ibm.com
HTTP Console Port        : 7788
TPS Console Port         : 7802
TP Upload Port           : 4889
TPS Upload Port          : 4903
Instance Home             : /dl/gc_inst/em/EMGC_OMS1
Log Directory Location   : /dl/gc_inst/em/EMGC_OMS1/sysman/log
Log is not configured with SLB or virtual hostname
Log Upload is locked.
Log Console is locked.
Active CA ID: 1
Console URL: https://atstools100.dfw.ibm.com:7802/em
Upload URL: https://atstools100.dfw.ibm.com:4903/empbs/upload

Domain Information
Domain Name       : GCDomain
Admin Server Host : atstools100.dfw.ibm.com
Admin Server HTTPS Port: 7102
Admin Server is RUNNING

Oracle Management Server Information
Managed Server Instance Name: EMGC_OMS1
Oracle Management Server Instance Host: atstools100.dfw.ibm.com
dbTier is Up
Oracle Management Server is Up
IMD Engine is Up

Publisher Server Information
Publisher Managed Server Name: BIP
Publisher Server is Down
Publisher is disabled, to enable BI Publisher on this host, use the 'emctl config oms -enable_bip' command
oracle@atstools100:/dl>
```

Figure 3-4 OMS status details

The steps in this section are for a simple configuration installation process. Enterprise Manager Cloud Control offers multiple configurations. For more information about advanced installation and configuration, see the relevant Oracle manuals and documentation.

3.3 Updating the agent at the Enterprise Manager Cloud Control to monitor Oracle Databases on LinuxONE (online)

Enterprise Manager Cloud Control 13c by default includes the Cloud Control Agent software installed for the operating system where Enterprise Manager Cloud Control 13c is installed. In our case, the agent for the Linux x86-64 is available when the Enterprise Manager Cloud Control 13c is installed on the Linux x86-64 server. However, the agent to monitor the Oracle databases on LinuxONE architecture is not available in the base installation.

Management Agent software for LinuxONE architecture must be downloaded and applied by using the Self Update feature. The Self Update feature is a dashboard option that is used to obtain information about new updates. It provides a process flow to review, download, and apply those updates.

The Software Library is a repository that stores software entities, such as software patches and application software. The software library stores its repository on a file system that is accessible by the OMS. Depending on the usage, the size of this file can reach 50 GB. For more information about advanced features and the requirements of the Software Library, see the MOS document Understanding and Configuring Software Library In 12c Cloud Control, ID 1368035.1.

3.3.1 Configuring the Software Library storage space

The storage location in the Software Library represents a repository of files. These files are uploaded by the Software Library, and you must add at least one upload file storage location. This location can be an OMS Shared location or OMS Agent location. This step is the first step to be carried out, and in our case, we decided to use an OMS Shared location. We used the default location, which was configured in the system during the Enterprise Manager Cloud Control 13c installation.

Click **Setup → Provisioning and Patching → Software Library** to configure the Software Library. In our case, /d1/swlib was configured as a Software Library location for the OMS Shared File System; therefore, we used that location.

3.3.2 Acquiring the LinuxONE agent in online mode

The Enterprise Manager Cloud Control is in online mode when it has access to My Oracle Support through the internet. The example in this section shows how the Linux on LinuxONE agent is acquired in online mode on an Enterprise Manager Cloud Control running on an x86-64 Linux architecture. Oracle refers to this solution in certify for LinuxONE architecture as “IBM Linux on System z”.

In general, the following tasks are completed:

- ▶ Set up My Oracle Support Credentials in Enterprise Manager Cloud Control.
- ▶ Verify that Enterprise Manager Cloud Control is in online mode.
- ▶ Update the available agents in Enterprise Manager Cloud Control by using the Self Update feature.
- ▶ Download and applying the LinuxONE Management Agent Software (labeled as “IBM Linux on System z”).

3.3.3 Setting up My Oracle Support Credentials

Complete the following steps to set up My Oracle Support Credentials in Enterprise Manager Cloud Control:

1. Log on to Enterprise Manager Cloud Control 13c.
2. Click **Setup → My Oracle Support → Set Credentials**.
3. Enter the My Oracle Support Credentials and click **Apply**.

The My Oracle Support Preferred Credentials are set up in Cloud Control 13c.

3.3.4 Verifying that Enterprise Manager Cloud Control is in online mode

Complete the following steps to ensure that Cloud Control is set to the online mode:

1. Click **Setup → Provisioning and Patching → Offline Patching**. Then, change the setting for Connection to Online.
2. Update the available agents in Enterprise Manager Cloud Control by using the Self Update feature. By default, only the Management Agent software for the OMS host platform is downloaded and applied. The other host agents’ availability must be checked by using the Self Update feature of the Enterprise Manager Cloud Control.
3. Click **Setup → Extensibility → Self Update**.
4. Click **Check Updates** to get the complete list of available updates for the Agent Software.

A background job is submitted to get the new updates from My Oracle Support.

3.3.5 Downloading and applying the LinuxONE Management Agent Software

After the list of available updates is populated in the system, you can download the required Management Agent Software and then apply it to the OMS host.

Complete the following steps:

1. Click **Setup** → **Extensibility** → **Self Update**.
2. Select the entity type Agent Software and select Open from the Action menu. The entity type page shows agent software for different platforms.

In our example, we selected **IBM: Linux on System z OS Platform** and **Oracle Enterprise Manager Cloud Control Agent 13c Release 2 Version 13.2.0.0.0** from the list of available updates.

3. Click **Download** and schedule the download job for immediate execution.

You can monitor the job. After the job completes, the status is now “Downloaded”.

4. Click **Apply** for the Downloaded Agent.

This step stages the agent software in the Software Library and makes it available to the Add Targets wizard, which we used to install the agent on the LinuxONE host machines.

After the job completes, the status is now Applied, as shown in Figure 3-5.

Status	OS Platform	Revision/Version	Vendor	Size(MB)	Description
Available	IBM AIX on POWER Systems	0	Oracle	365.295	Agent Software (12.1.0.1.0) for IBM AIX on POWER Systems (64-bit)
Available	IBM AIX on POWER Systems	0	Oracle	318.497	Agent Software (12.1.0.2.0) for IBM AIX on POWER Systems (64-bit)
Available	IBM AIX on POWER Systems	0	Oracle	325.340	Agent Software (12.1.0.3.0) for IBM AIX on POWER Systems (64-bit)
Available	IBM: Linux on System z	0	Oracle	294.392	Agent Software (12.1.0.5.0) for IBM: Linux on System z
Applied	IBM: Linux on System z	0	Oracle	531.625	Agent Software (13.2.0.0.0) for IBM: Linux on System z
Available	IBM: Linux on System z	0	Oracle	293.015	Agent Software (12.1.0.4.0) for IBM: Linux on System z
Available	IBM: Linux on System z	0	Oracle	286.743	Agent Software (12.1.0.3.0) for IBM: Linux on System z

Figure 3-5 List of agents that are applied

The online update process for the Management Agent Software at the Enterprise Manager Cloud Control is complete. Now, the agent is available to deploy and monitor the Oracle databases on the LinuxONE hosts.

3.4 Updating the agent at the Enterprise Manager Cloud Control to monitor Oracle Databases on LinuxONE (offline)

In section 3.3, “Updating the agent at the Enterprise Manager Cloud Control to monitor Oracle Databases on LinuxONE (online)” on page 29, we described the steps to update the Management Agent Software library in online mode. If the internet connection is not available on the OMS host server, you must follow the offline procedures that are outlined in this section to update the Management Agent Software library.

In the following example, we show how the LinuxONE agent is acquired in offline mode on an Enterprise Manager Cloud Control running on an x86-64 Linux architecture.

3.4.1 Acquiring the LinuxONE agent in offline mode

In the example that is shown in this section, we show how the Software Library can be updated in offline mode on an Enterprise Manager Cloud Control that is running on an x86-64 Linux architecture to get the management agent for LinuxONE. Oracle requires that you use the Enterprise Manager Command-Line Interface (emcli) to update the Enterprise Manager Cloud Control Software updates.

In general, the following tasks are performed:

- ▶ Verify that Enterprise Manager Cloud Control is in offline mode.
- ▶ Update the available agents in Enterprise Manager Cloud Control by using the Self Update feature.
- ▶ Use emcli to update the available software.
- ▶ Acquire Software Updates offline.
- ▶ Acquire Management Agent Software offline.

3.4.2 Verifying that Enterprise Manager Cloud Control is in offline mode

You must ensure that Cloud Control is set to the offline mode. Click **Setup → Provisioning and Patching → Offline Patching**. Then, change the setting for Connection to Offline.

3.4.3 Updating the agents in Enterprise Manager Cloud Control using the Self Update feature

By default, only the Management Agent software for the OMS host platform is downloaded and applied. The other host agents' availability must be checked by using Enterprise Manager Cloud Control by using the Self Update feature.

Complete the following steps:

1. To get the complete list of available updates for the agent software, click **Setup Extensibility → Self Update**. Then, click **Check Updates**. A window opens and shows the URL that you use to download a catalog of all the updates for the agents (see Figure 3-6).

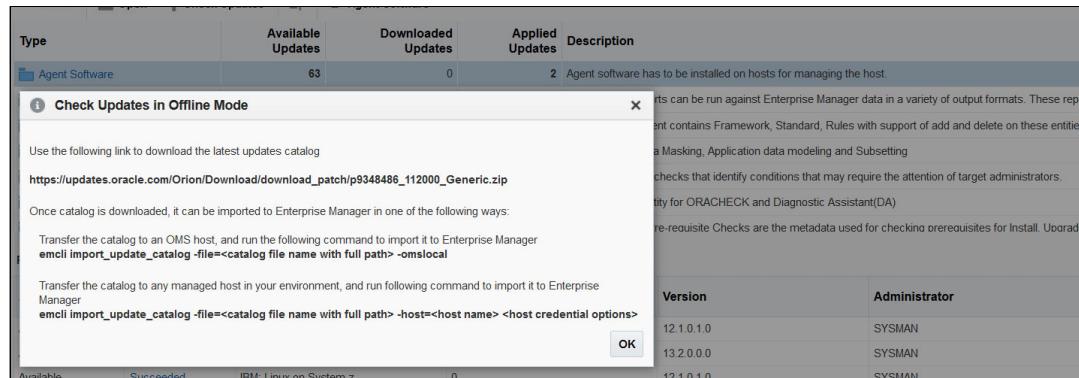


Figure 3-6 Check updates in offline mode

2. From any computer that has an Internet connection, download the catalog file by using the following URL:

https://updates.oracle.com/Orion/Download/download_patch/p9348486_112000_Generic.zip

3. Log on with the oracle credentials for MOS.
4. Figure 3-7 shows the window for downloading the catalog updates. Copy the downloaded compressed file to the Enterprise Manager Cloud Control Server where the OMS is running.

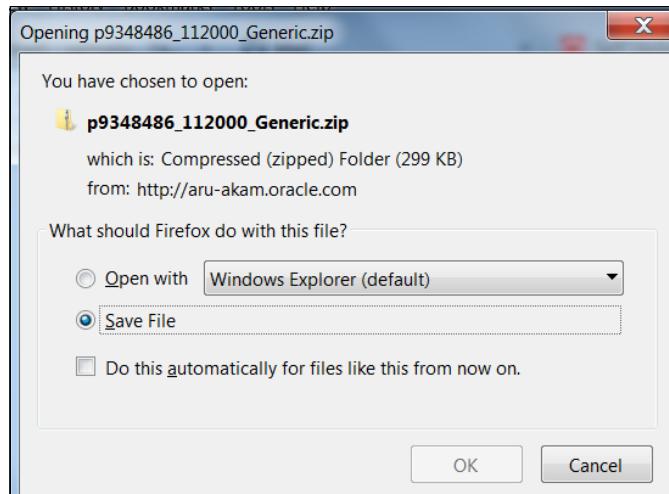


Figure 3-7 Saving the compressed file

The name of the compressed file in our example is p9348486_112000_Generic.zip. That file is copied into the/d1/tmp location at the OMS server.

Using emcli to update the available software

Oracle provides the emcli command-line utility to perform some of the Enterprise Manager Cloud Control functions. This utility is installed and configured during OMS installation.

Complete the following steps to update the available software by using emcli:

1. Run emcli to log in (from OMS home) (see Figure 3-8).

```
oracle@atstools100:/d1/mw> bin/emcli login -username=sysman
Enter password :
Login successful
oracle@atstools100:/d1/mw>
```

Figure 3-8 emcli login

2. Synchronize Enterprise Manager CLI by running emcli sync (see Figure 3-9).

```
oracle@atstools100:/d1/mw> bin/emcli sync
Synchronized successfully
oracle@atstools100:/d1/mw>
```

Figure 3-9 emcli sync

3. Ensure that the Software Library is configured in the system by clicking **Setup** → **Select Provisioning and Patching** → **Software Library** (in our case, /d1/swlib is used for Software Library).
4. To import the downloaded compressed archive in to the Oracle Management Service instance, run emcli import_update_catalog, as shown in Figure 3-10.

```

oracle@atstools100:/dl/mw> bin/emcli import_update_catalog -omslocal -file=/dl/tmp/p9348486_112000_Generic.zip
Processing catalog for Agent Software
Processing update: Agent Software - Agent Software (13.2.0.0.0) for IBM: Linux on System z
Processing catalog for Middleware Profiles and Gold Images
Processing catalog for Data Masking and Subsetting templates
Processing catalog for Extensibility Development Kit
Processing catalog for Plug-in
Processing catalog for Management Connector
Processing catalog for Compliance Content
Processing catalog for Informational
Processing catalog for Provisioning Bundle
Processing catalog for Diagnostic Tools

Successfully uploaded the Self Update catalog to Enterprise Manager. Use the Self Update Console to view and manage updates.
Time taken for import catalog is 24.219 seconds.
oracle@atstools100:/dl/mw>

```

Figure 3-10 emcli import_update_catalog

- After the command completes, you can see that the agents are shown as available in the Self Update window. Click **Setup** → **Extensibility** → **Self Update**, select the entity type **Agent Software**, and select **Open** from the Action menu. The entity type page shows agent software for different platforms.

Acquiring the Management Agent Software offline

To acquire the Management Agent Software offline, complete the following steps:

- Select your OS platform and version. In our example, we select **IBM: Linux on System z** for the OS Platform and **13.2.0.0.0** for the version from the list of available updates.
- A window opens and provides a link to download the selected agent and instructions about how to import the agent software updates into Enterprise Manager, as shown in Figure 3-11. Using this link, download the compressed file to any computer with an internet connection and copy the downloaded compressed file to the Enterprise Manager Cloud Control Server where the OMS is running.

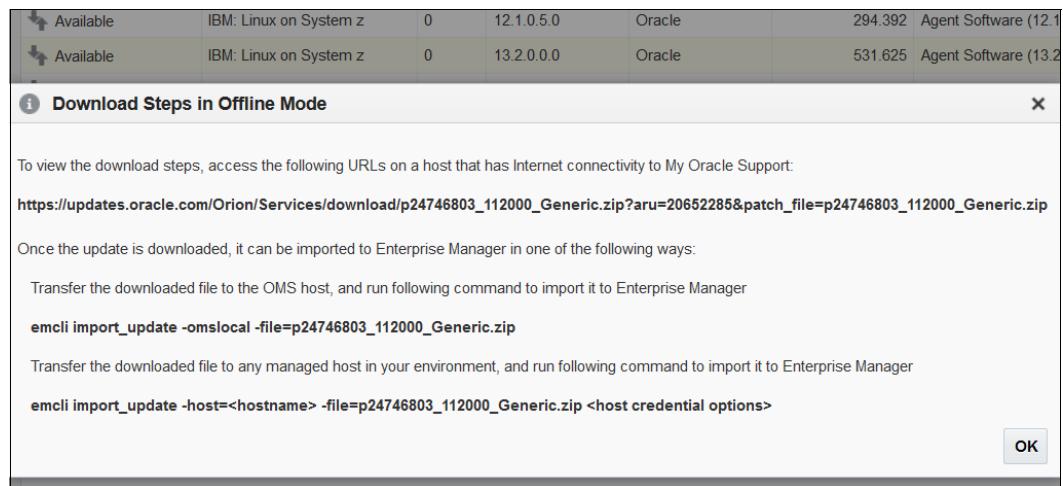


Figure 3-11 Offline download

- Run emcli to log in (from OMS home).
- Synchronize the Enterprise Manager CLI by running emcli sync.
- Run emcli import_update to specify the absolute location for the downloaded patch file and omslocal options, as shown in Figure 3-12.

```
oracle@atstools100:/dl/mw> bin/emcli import_update -omslocal -file=/dl/tmp/p24746803_112000_Generic.zip
Processing update: Agent Software - Agent Software (13.2.0.0.0) for IBM: Linux on System z
Successfully uploaded the update to Enterprise Manager. Use the Self Update Console to manage this update.
oracle@atstools100:/dl/mw>
```

Figure 3-12 emcli import_update

This step completes the Agent Updates in offline mode. Now, the status in the Self Update window is shown as “Downloaded” for the IBM: Linux on System z agent type.

6. Click **Apply** for the Downloaded Agent. This step stages the agent software in the Software Library and makes it available to the Add Targets wizard, which we used to install the agent on the host machines in our example.

The agent is now available to deploy and monitor the Oracle databases on LinuxONE hosts.

3.5 Deploying the agents from the Cloud Control console

Oracle Management Agent 13c for LinuxONE hosts can be deployed from the Enterprise Manager Cloud Control 13c console or by using the silent installation method. This section describes the processes to deploy the agents from the Enterprise Manager Cloud Control 13c console. Oracle recommends the use of the Add Host Targets wizard, which converts an unmanaged host to a managed host in the Enterprise Manager system by installing an Oracle Management Agent 13c. For Oracle Real Application Clusters (RACs) with multiple nodes, the Management Agent must be installed on each of the nodes separately in the Add Host Targets wizard by adding each node explicitly as a destination host.

For more information about the hardware and software requirements for installing the Cloud Control Agent, see the [Installing Oracle Management Agents website](#).

The network between the Cloud Control Server, where the OMS is running, and the destination hosts should be accessible.

In our example, we used the ping by host name method to ensure that the OMS Server and hosts can be reached.

To install Oracle Management Agent 13c for LinuxONE from Enterprise Manager Cloud Control 13c, complete the following steps:

1. To add or install an agent on a host, the software distribution of the agent that corresponds to the host’s platform must be available in the Software Library. In our example, we verified the availability of LinuxONE agents’ availability in the Enterprise Manager Cloud Control Server by completing the following steps:
 - a. Log on to Enterprise Manager Cloud Control 13c.
 - b. Click **Setup** → **Extensibility** → **Self Update** in the Status section of the Self Update window. Click the Agent Software type.

You can see that the Agent Software for Linux on System z has a status of “Applied” in the Agent Software Updates section. When we highlighted the rows, the bottom pane showed a status, such as when the agent software was available, downloaded, and applied.

2. Click **Setup** → **Add Target** → **Add Targets Manually** to deploy the agent.

3. In the Add Targets Manually window, select **Add Host Targets** and click **Add Host**. Then, **Install Agent on Host**.
4. In the Host and Platform window, we completed the following steps for our example:
 - a. Accept the default name that is assigned for this session.
 - b. Click **Add** and enter the fully qualified name of the host. We select **IBM: Linux on System z** as the platform of the host on which we wanted to install the Management Agent and click **Next**.
5. In the Installation Details window (see Figure 3-13), complete the following steps:
 - a. In the Installation Details section, for Installation Base Directory, enter the absolute path to the base directory (the software binary files, security files, and inventory files of Management Agent are copied here). In our case, it is /u01/app/oracle/agentHome.
 - b. For Instance Directory, enter the directory where you want all Management Agent-related configuration files to be stored. We accepted the default instance directory location (in our case, /u01/app/oracle/agentHome/agent_inst).

Note: Maintain the instance directory inside the installation base directory.

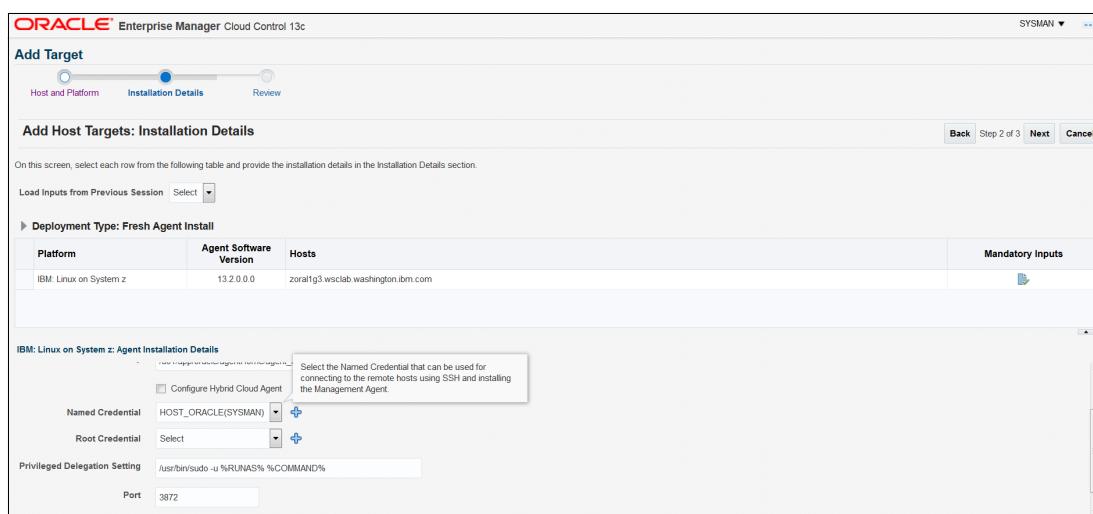


Figure 3-13 Installation information

- c. From Named Credential list, add a profile whose credentials are used for setting up the SSH connectivity between the OMS and the remote host, and for installing a Management Agent.
- d. In the Review window, review the details and then, click **Deploy Agent** to install the Management Agent.

You can monitor the progress of the installation in the Add Hosts Status window.

- e. If the deployment fails during the prerequisite check stage (as was the case with our installation), root.sh authorization messages might be displayed. We continued the installation by selecting the **Continue all Hosts** option.

The Agent Deployment Summary message displays when the process completes.

- f. Run root.sh in the host location as recommended. Click **Done**.

- To check the availability of the hosts, click **Targets** → **Hosts** in the Cloud Control window, as shown in Figure 3-14.

Name	Status	Pending Activation	Incidents	Compliance Violations	Average Compliance Score	CPU Util %	Mem Util %	Total IO/sec
			● ✖ ⚠ 🔴	Critical Warning Minor				
atstools100.dfw.ibm.com	⬆️	-	0 ✖️ 0 0	1 Critical 0 Warning 0 Minor	51	16.03	19.11	16.03
zoral1g4.wsclab.washington.ibm.com	⬆️	-	0 ✖️ 0 0	1 Critical 0 Warning 0 Minor	51	0	0	0

Figure 3-14 Availability of hosts

Oracle Management Agent 13c for LinuxONE hosts is now deployed from Enterprise Manager Cloud Control 13c console.

3.6 Deploying the agents in silent mode

Oracle Management Agent 13c for LinuxONE hosts can be deployed from the Enterprise Manager Cloud Control 13c console or by the silent installation method. This section describes the processes to deploy it in silent mode.

Installing a Management Agent in silent mode requires the Enterprise Manager CLI utility. Using this utility, you can install fresh Management Agents, install Shared Agents, and clone existing Management Agents.

To install Oracle Management Agent 13c for LinuxONE in silent mode, complete the following steps:

- Run the emcli commands to determine the list of supported platforms.
- Run **emcli** and to log in and then sync Enterprise Manager CLI by running **emcli sync**.
- Ensure that the Software Library is configured and available in the system by running **emcli list_add_host_platforms** (see Figure 3-15). This command lists the platforms for which the Management Agent software is available in Software Library.

```
oracle@atstools100:/dl/mw> bin/emcli list_add_host_platforms
Platform ID  Platform Name
226          Linux x86-64
209          IBM: Linux on System z
oracle@atstools100:/dl/mw>
```

Figure 3-15 emcli list add_host_platforms

From the displayed output, you can see that IBM: Linux on System z is available in the Enterprise Manager Cloud Control Server.

- Create named credentials for the “oracle” user on the target server, which are monitored by the emcli **create_named_credential** command with the parameters **-cred_name**, **-auth_target_type**, **-cred_type**, and **-attributes**, as shown in Example 3-1.

Example 3-1 emcli create_named_credential

```
bin/emcli create_named_credential -cred_name=orahost -auth_target_type=host
-cred_type=HostCreds -attributes="HostUserName:oracle;HostPassword:oracle"
```

- Run **submit_add_host** to submit the Add Host command, as shown in Example 3-2.

Example 3-2 submit_add_host

```
bin/emcli submit_add_host -host_names=zoral1g3.wsclab.washington.ibm.com  
-platform=209 -installation_base_directory=/u01/app/oracle/agentHome  
-credential_name=orahost
```

- Now, you can track the running of the **submit_add_host** command, as shown in Example 3-3.

Example 3-3 submit_add_host status command

```
bin/emcli get_add_host_status  
session_name=ADD_HOST_SYSMAN_Jan_24_2017_8:16:56_PM_CST
```

The results are shown in Figure 3-16.

ST					
OverAll Status : Agent Deployment Succeeded					
Host	Platform Name	Initialization	Remote Prerequisite	Agent Deployment	
zoral1g3.wsclab.washington.ibm.com	IBM: Linux on System z	Succeeded	Succeeded	Succeeded	
oracle@atstools100:/dl/mw> █					

Figure 3-16 submit_add_host status

After the job status is Succeeded, you can see that the host is added in the Enterprise Manager Cloud Control.

The logs for the agent deployment at the target server can be found under the agent installation base directory /oracle/agentHome/agent_inst/install/logs.

You can also check the agent status by browsing to the Management Agent home at the target server and running the **status** command, as shown in Example 3-4.

Example 3-4 Agent status

```
agentHome/agent_13.2.0.0.0/bin/emctl status agent
```

The results are shown in Figure 3-17.

```
Oracle Enterprise Manager Cloud Control 13c Release 2
Copyright (c) 1996, 2016 Oracle Corporation. All rights reserved.

Agent Version      : 13.2.0.0.0
OMS Version       : 13.2.0.0.0
Protocol Version  : 12.1.0.1.0
Agent Home        : /u01/app/oracle/agentHome/agent_inst
Agent Log Directory : /u01/app/oracle/agentHome/agent_inst/sysman/log
Agent Binaries    : /u01/app/oracle/agentHome/agent_13.2.0.0.0
Core JAR Location : /u01/app/oracle/agentHome/agent_13.2.0.0.0/jlib
Agent Process ID : 18788
Parent Process ID : 18748
Agent URL         : https://zoral1g3.wsclab.washington.ibm.com:3872/emd/main/
Local Agent URL in NAT : https://zoral1g3.wsclab.washington.ibm.com:3872/emd/main/
Repository URL   : https://atstools100.dfw.ibm.com:4903/empbs/upload
Started at        : 2017-01-24 20:26:04
Started by user   : oracle
Operating System  : Linux version 3.0.76-0.11-default (s390x)
Number of Targets : 2
Last Reload       : (none)
Last successful upload : 2017-01-24 20:28:59
Last attempted upload : 2017-01-24 20:28:59
Total Megabytes of XML files uploaded so far : 0.34
Number of XML files pending upload : 0
Size of XML files pending upload(MB) : 0
Available disk space on upload filesystem : 32.62%
Collection Status : Collections enabled
Heartbeat Status  : Ok
Last attempted heartbeat to OMS : 2017-01-24 20:29:11
Last successful heartbeat to OMS : 2017-01-24 20:29:11
Next scheduled heartbeat to OMS : 2017-01-24 20:30:12

-----
Agent is Running and Ready
oracle@zoral1g3:/u01/app/oracle>
```

Figure 3-17 Agent status results

7. You can check that the EMD upload completed successfully by browsing to the Management Agent home at the target server and running the upload agent, as shown in Figure 3-18.

```
oracle@zoral1g3:/u01/app/oracle> agentHome/agent_13.2.0.0.0/bin/emctl upload agent
Oracle Enterprise Manager Cloud Control 13c Release 2
Copyright (c) 1996, 2016 Oracle Corporation. All rights reserved.

EMD upload completed successfully
oracle@zoral1g3:/u01/app/oracle>
```

Figure 3-18 EMD Upload

8. To verify that the Management Agent is running, click **Setup** → **Manage Cloud Control Agents**.

You can see that the Management Agent is running.

3.7 Adding the databases for monitoring

After the agent for LinuxONE is deployed, complete the following steps to discover the Oracle databases on that host for monitoring:

1. In the Cloud Control window, click **Targets → Databases**, select **Search List** in the Databases section, and then, click **Add**.
2. In the Add Database Instance Target window, specify the Host window. In the Host field in our example, we specify the following fully qualified host name:

zoral1g3.wsclab.washington.ibm.com

Click **Continue** (see Figure 3-19).

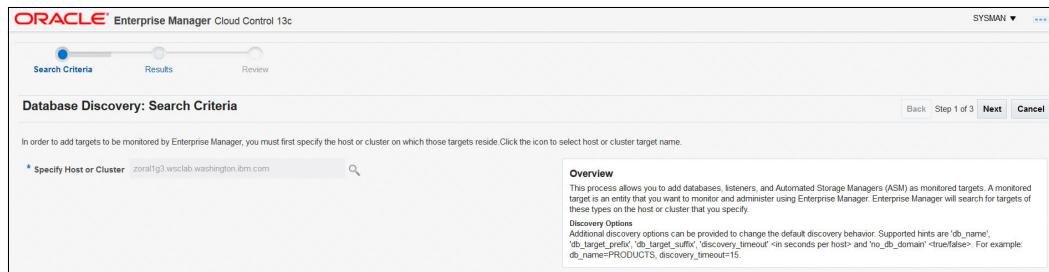


Figure 3-19 Selecting target

The agent discovers the database oratest and the Listener.

3. Configure the database by selecting the **Configure** window at the oratest line and specifying the Database-related parameters. Ensure that you can connect to the database, as shown in Figure 3-20.

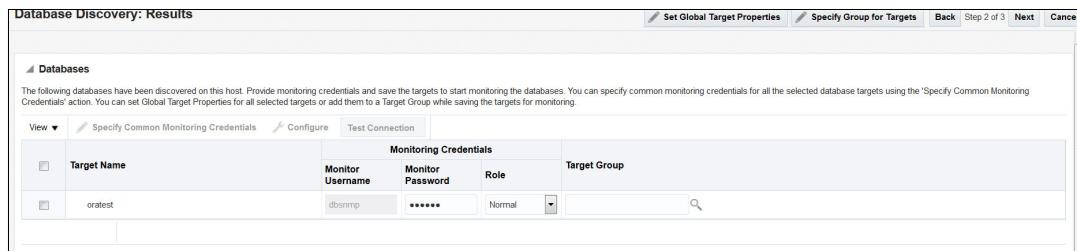


Figure 3-20 Database Discovery: Results

4. Save the database configuration in the Database Discovery Review window, as shown in Figure 3-21.

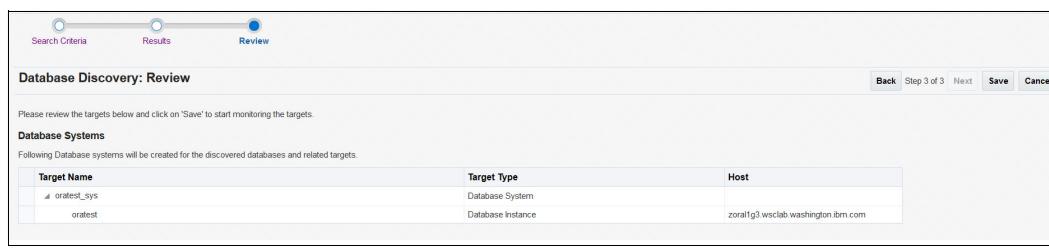


Figure 3-21 Database Discovery: Review

3.8 Summary

In this chapter, we shared our experiences with installing a Cloud Control Server on an x86-based Linux server. We deployed the agents from that server to monitor the databases that are running on LinuxONE.

Before deploying the agents, we also updated the Cloud Control Software Library with the required levels of agent software and plug-ins by connecting to the Oracle repository site online. Then, starting with Oracle Enterprise Manager Cloud Control 13c Release 2 (13.2.0.0.0), we showed how the command-line utility emcli can also be used to update the Grid Control Software Library in offline mode.

The agents were deployed from the Cloud Control Console and by using the silent agent deployment option at the Linux guests. We concluded this chapter by showing how to enable the Cloud Control to monitor Oracle databases.



Oracle encryption on LinuxOne

In this chapter, we describe the steps that are used to implement Oracle Transparent Data Encryption (TDE) on Oracle 12c Release 2 (12cR2) with Red Hat Enterprise Linux Servers 7 and Oracle workloads on LinuxONE.

This chapter includes the following topics:

- ▶ 4.1, “Oracle Transparent Data Encryption on Oracle Database 12cR2” on page 44
- ▶ 4.2, “Enabling encryption on Oracle Database” on page 48
- ▶ 4.3, “Managing TDE master keys and wallets” on page 54
- ▶ 4.4, “Performance” on page 56
- ▶ 4.5, “Using Oracle TDE on Oracle RAC” on page 59
- ▶ 4.6, “Summary” on page 60

4.1 Oracle Transparent Data Encryption on Oracle Database 12cR2

Oracle Transparent Data Encryption (TDE) was introduced on version 10gR2 and is available with the Oracle Advanced Security Option. TDE protects sensitive data that is stored in tables and tablespaces and it is transparent to the application. It helps to protect data at rest (that is, the data that is stored on media) if that storage media or data file is stolen.

On Oracle version 12c, TDE provides a different interface that is used to manage the wallet and master key.

For more information about Oracle Transparent Encryption, see [Oracle Note Doc ID 1228046.1](#).

4.1.1 Benefits of the use of Transparent Data Encryption

TDE ensures that sensitive data is encrypted and that the data is safe if storage media or data files are stolen.

Nothing needs to be changed in the application side to handle encrypted data. The encryption and decryption is handled by the database.

Data is transparently decrypted for database users and applications that access this data. Database and application users are not aware that the data is stored in encrypted form.

Data can be encrypted with zero downtime by using online table redefinition. Data also can be encrypted offline during a maintenance window.

The TDE master encryption key and keystore management is automatically done by Oracle database and no action is required by any user or application.

4.1.2 Installing Transparent Data Encryption

Oracle TDE is available by default in Oracle RDBMS Enterprise Edition. An Oracle Advantage Security License must be purchased to use this feature.

For more information about Oracle Licensing, see the [Database Licensing Information User Manual](#).

4.1.3 Configuring and administering Transparent Data Encryption

To configure TDE, the database user must be granted the ADMINISTER KEY MANAGEMENT system privilege. If you must open the keystore at the mount stage, the SYSKM administrative privilege must be granted. This privilege includes ADMINISTER KEY MANAGEMENT system privilege and other necessary privileges.

To configure TDE tablespace or column encryption, the ADMINISTER KEY MANAGEMENT or SYSKM system privileges are not needed. Instead, only the CREATE TABLE, ALTER TABLE, and CREATE TABLESPACE privileges are needed.

To start the configuration, the keystore must be configured. Keystore software is a container that stores TDE master encryption key. A location for your keystore must be defined on the sqlnet.ora file of your database. One keystore is available for each database and their location is defined on the sqlnet.ora file.

In our lab, we used RedHat Enterprise Linux 7 on IBM LinuxOne. We also used Oracle Database Enterprise Edition 12cR2 to install the database. For more information, see Chapter 2, “Setting up Linux guests to install Oracle Database 12c Release 1” on page 5.

The first step is to set up the ENCRYPTION_WALLET_LOCATION parameter in \$ORACLE_HOME/network/admin/sqlnet.ora to define the keystore location. Oracle Database searches for the keystore location. In our lab environment, we set up this parameter to point to a local ext4 file system in /home/oracle/tde12c.

Ensure that the folder is created on the server and then, update sqlnet.ora file, as shown in Example 4-1.

Example 4-1 Example of SQLNET.ORA file

```
[oracle@rdbkora3] :/home/oracle> cat  
/u01/app/oracle/grid/12.2.0.1/network/admin/sqlnet.ora  
# sqlnet.ora Network Configuration File:  
/u01/app/oracle/grid/12.2.0.1/network/admin/sqlnet.ora  
# Generated by Oracle configuration tools.  
  
NAMES.DIRECTORY_PATH= (TNSNAMES, EZCONNECT)  
  
ENCRYPTION_WALLET_LOCATION=  
(SOURCE=  
  (METHOD=FILE)  
  (METHOD_DATA=  
    (DIRECTORY=/home/oracle/tde12c)))  
  
[oracle@rdbkora3] :/home/oracle>
```

If the parameter is not set on the sqlnet.ora file, Oracle Database searches for a keystore in the default location of ORACLE_BASE/admin/DB_UNIQUE_NAME/wallet or ORACLE_HOME/admin/DB_UNIQUE_NAME/wallet where DB_UNIQUE_NAME is the unique name of the database that is specified in the initialization parameter file.

The default location that is inside the database can be checked by using V\$ENCRYPTION_WALLET, which is the same view that is used to check status and location of the wallet.

The next step is to create a keystore, which also is known as a *wallet*. The following types of keystores are available:

- ▶ Password-based software keystores are protected by using a password that you create. You must open this type of keystore before the keys can be retrieved or used.
- ▶ Auto-login software keystores are protected by a system-generated password, and do not need to be opened by a security administrator.
- ▶ Local auto-login software keystores are auto-login software keystores that are local to the computer on which they are created. Local auto-login keystores cannot be opened on any computer other than the one on which they are created.

In our lab, we used password-based software keystores. The steps that were used to create the wallet as recommended by Oracle official documentation are shown in Example 4-2.

Example 4-2 Creating keystore on the database

```
SQL> select * from v$encryption_wallet;

WRL_TYPE          WRL_PARAMETER          STATUS
WALLET_TYPE        WALLET_OR_FULLY_BAC    CON_ID
-----
-----          -----
FILE              /home/oracle/tde12c/      NOT_AVAILABLE
UNKNOWN           SINGLE      UNDEFINED      0

SQL> administer key management create keystore '/home/oracle/tde12c/' identified
by "Xgt6870Kj";

keystore altered.
```

After running the command to create the keystore, the wallet file ewallet.p12 is created in the location that is defined in the sqlnet.ora file.

The file must be opened manually before it can be used, as shown in Example 4-3.

Example 4-3 Checking keystore status and open it

```
SQL> select * from v$encryption_wallet;

WRL_TYPE          WRL_PARAMETER          STATUS
WALLET_TYPE        WALLET_OR_FULLY_BAC    CON_ID
-----
-----          -----
FILE              /home/oracle/tde12c/      CLOSED
UNKNOWN           SINGLE      UNDEFINED      0

SQL> administer key management set keystore open identified by "Xgt6870Kj";

keystore altered.
```

Note: The keystore (wallet) must be opened to create keys. If the keystore is accessible because of the auto_login feature, keys *cannot* be added to the wallet.

When the wallet status is set as OPEN_NO_MASTER_KEY on the database, the master key is not yet created in the keystore (see Example 4-4).

Example 4-4 Checking wallet status is now updated

```
SQL> select * from v$encryption_wallet;

WRL_TYPE          WRL_PARAMETER          STATUS
WALLET_TYPE        WALLET_OR_FULLY_BAC    CON_ID
-----
-----          -----
FILE              /home/oracle/tde12c/      OPEN_NO_MASTER_KEY
PASSWORD           SINGLE      UNDEFINED      0
```

The step that must be run to create the master key is shown in Example 4-5. A best practice is to create it by using a backup.

Example 4-5 Creating master key with backup

```
SQL> administer key management create key identified by "Xgt6870Kj" with backup;
```

keystore altered.

```
SQL> select * from v$encryption_wallet;
```

WRL_TYPE WALLET_TYPE	WRL_PARAMETER WALLET_OR_FULLY_BAC	STATUS CON_ID
FILE	/home/oracle/tde12c/	OPEN_NO_MASTER_KEY
PASSWORD	SINGLE UNDEFINED	0

To create an encrypted table or tablespace, the master key must be activated. To activate the master key, the key_id must be imported onto the database. Then, the master key is activated, as shown on Example 4-6.

Example 4-6 Activate key

```
SQL> select key_id from v$encryption_keys;
```

```
KEY_ID
```

```
ATa9LKxd408zvyorUNpFZsMAAAAAAAAAAAAAAAAAAAAAAAA
```

```
SQL> administer key management use key  
'ATa9LKxd408zvyorUNpFZsMAAAAAAAAAAAAAAAAAAAAAAAA' identified by "Xgt6870Kj"  
with backup;
```

keystore altered.

```
SQL> select key_id,activation_time from v$encryption_keys;
```

```
KEY_ID
```

```
ACTIVATION_TIME
```

```
ATa9LKxd408zvyorUNpFZsMAAAAAAAAAAAAAAAAAAAAAAAA
```

```
20-JUN-19 02.22.35.396868 PM -04:00
```

After the master key is activated, the status of the wallet changes to OPEN. This status can be confirmed by checking the view v\$encryption_wallet, as shown in Example 4-7.

Example 4-7 Open wallet and check status

```
SQL> administer key management set keystore open identified by "welcome1";
```

keystore altered.

```
SQL> select * from v$encryption_wallet;
```

WRL_TYPE	WRL_PARAMETER	STATUS
WALLET_TYPE	WALLET_OR_FULLY_BAC	CON_ID
FILE	/home/oracle/tde12c/	OPEN
PASSWORD	SINGLE NO	0

After these steps are completed, TDE is enabled on the database. Next, the database objects are encrypted.

For more information about Configuring TDE for 12c Database version, see [Oracle Note Doc ID 1964158.1](#).

4.2 Enabling encryption on Oracle Database

After the keystore configuration process is completed, Oracle TDE encryption can be enabled for a column in a table or an entire tablespace. In this section, some use cases are described that were performed in our lab to encrypt database data at the database level with the server. In a real-world scenario, your organization must decide whether you want a broad or granular level of encryption.

Encrypting columns in a table

To encrypt a column, the `ALTER TABLE` command is used. If the table features many rows, this process can take some time to complete because all of the values that are stored in the column that is encrypted must be replaced by encrypted strings. In our test, the column `CUSTOMER_NAME` on table `CUSTOMER` was encrypted, as shown in Example 4-8.

Example 4-8 Encrypt column CUSTOMERS_NAME on ORDDATA.CUSTOMER table

```
SQL> desc ORDDATA.customers
Name          Null?    Type
-----        -----
CUSTOMER_ID      NOT NULL NUMBER(10)
CUSTOMER_NAME    NOT NULL VARCHAR2(50)
CITY            VARCHAR2(50)

SQL> alter table ORDDATA.customers modify( customer_name encrypt SALT);

Table altered.

SQL> desc ORDDATA.customers
Name          Null?    Type
-----        -----
CUSTOMER_ID      NOT NULL NUMBER(10)
CUSTOMER_NAME    NOT NULL VARCHAR2(50) ENCRYPT
CITY            VARCHAR2(50)
```

If encryption of a column on a table must be disabled, use the **ALTER TABLE MODIFY** command that uses a decrypt clause, as shown in Example 4-9.

Example 4-9 Decrypting a table column

```
SQL> alter table ORDDATA.customers modify( customer_name decrypt);
```

```
Table altered.
```

Another approach is to add a column by using the encryption option on a table, as shown in Example 4-10.

Example 4-10 Adding an encrypted column to a table

```
SQL> ALTER TABLE ORDDATA.customers ADD (Address VARCHAR2(11) ENCRYPT);
```

```
Table altered.
```

```
SQL> desc ORDDATA.customers
Name          Null?    Type
-----
CUSTOMER_ID   NOT NULL NUMBER(10)
CUSTOMER_NAME NOT NULL VARCHAR2(50) ENCRYPT
CITY          VARCHAR2(50)
ADDRESS        VARCHAR2(11) ENCRYPT
```

It is important to know the data types of the objects that are planned for encryption because Oracle Database supports some types that can be encrypted by using TDE.

The following data types are supported by TDE:

- ▶ BINARY_DOUBLE
- ▶ BINARY_FLOAT
- ▶ CHAR
- ▶ DATE
- ▶ INTERVAL DAY TO SECOND
- ▶ INTERVAL YEAR TO MONTH
- ▶ NCHAR
- ▶ NUMBER
- ▶ NVARCHAR2
- ▶ RAW (older or extended)
- ▶ TIMESTAMP (includes TIMESTAMP WITH TIME ZONE and TIMESTAMP WITH LOCAL TIME ZONE)
- ▶ VARCHAR2 (older or extended)

Another important check is to confirm the maximum size of the data type. This check is important because it is impossible to encrypt a column if the encrypted column size is greater than the size that is allowed by the data type of the column.

The maximum allowable data type sizes are listed in Table 4-1.

Table 4-1 Maximum size for each data type

Data type	Maximum size
CHAR	1932 bytes
VARCHAR2 (older)	3932 bytes
VARCHAR2 (extended)	32,699 bytes
NVARCHAR2 (older)	1966 bytes
NVARCHAR2 (extended)	16,315 bytes
NCHAR	966 bytes
RAW (extended)	32,699 bytes

Because TDE encrypts at the SQL layer, TDE cannot be used on the following database features:

- ▶ Index types other than B-tree
- ▶ Range scan search through an index
- ▶ Synchronous change data capture
- ▶ Transportable tablespaces

TDE column encryption cannot be used for foreign key constraints because it generates an error message, as shown in Example 4-11.

Example 4-11 Error received when trying to encrypt a column that has a foreign key

```
SQL> alter table ORDDATA.ORDDCM_DOC_TYPES modify( DOC_TYPE_ID encrypt SALT);
alter table ORDDATA.ORDDCM_DOC_TYPES modify( DOC_TYPE_ID encrypt SALT)
*
```

ERROR at line 1:

ORA-28335: referenced or referencing FK constraint column cannot be encrypted

```
SQL>
```

Another available option is to create a table with an encrypted column by using the **CREATE TABLE** command, as shown in Example 4-12.

Example 4-12 Creating table with an encrypt column

```
SQL> CREATE TABLE ORDDATA.employee ( first_name VARCHAR2(128),last_name
VARCHAR2(128),empID NUMBER,salary NUMBER(6) ENCRYPT;
Table created.
```

```
SQL> desc ORDDATA.employee
 Name      Null?    Type
 -----
 FIRST_NAME  VARCHAR2(128)
 LAST_NAME   VARCHAR2(128)
 EMPID       NUMBER
 SALARY      NUMBER(6) ENCRYPT
```

Encrypting tablespaces

To encrypt a full tablespace, the **CREATE TABLESPACE** command can be used. It is possible to confirm if the tablespace is encrypted by checking the column ENCRYPTED on `dba_tablespaces`. In the tests we performed, we created encrypted tablespaces, as shown in Example 4-13.

Example 4-13 Creating an encrypted tablespace

```
SQL> CREATE TABLESPACE ORDDATA_TBS DATAFILE '+DORA31' SIZE 100M ENCRYPTION USING  
'AES256' DEFAULT STORAGE(ENCRYPT);
```

Tablespace created.

```
SQL> select TABLESPACE_NAME, ENCRYPTED from dba_tablespaces;
```

TABLESPACE_NAME	ENC
SYSTEM	NO
SYSAUX	NO
UNDOTBS1	NO
TEMP	NO
USERS	NO
RECLAIM_TS	NO
ORDDATA_TBS	YES

Up to Oracle version 12.1, a non-encrypted tablespace cannot be encrypted. An encrypted tablespace must be created and the data moved from the non-encrypted tablespace to the new encrypted tablespace. With Oracle version 12.2, tablespaces can be encrypted.

An online and offline encryption and decryption can be performed.

Offline encryption

To complete an offline encryption, it is necessary to set the tablespace to offline mode. Then, encrypt the tablespace by using the **ALTER TABLESPACE** command.

Individual data files of that tablespace can be encrypted by using the **ALTER DATABASE** statement. After the encryption is complete, the tablespace can be changed back to online status.

Performing an offline encryption on a tablespace is shown in Example 4-14.

Example 4-14 Performing an offline encryption

```
SQL> select TABLESPACE_NAME, ENCRYPTED from dba_tablespaces where  
TABLESPACE_NAME='DATA_IDX_TBS';
```

TABLESPACE_NAME	ENC
DATA_IDX_TBS	NO

```
SQL> alter tablespace DATA_IDX_TBS offline normal;
```

Tablespace altered.

```
SQL> alter tablespace DATA_IDX_TBS encryption offline encrypt;
```

Tablespace altered.

```
SQL> alter tablespace DATA_IDX_TBS online;
```

Tablespace altered.

```
SQL> select TABLESPACE_NAME,ENCRYPTED from dba_tablespaces where  
TABLESPACE_NAME='DATA_IDX_TBS';
```

TABLESPACE_NAME	ENC
DATA_IDX_TBS	YES

Offline decryption

Decryption can be done by decrypting the tablespace offline. The process includes taking the tablespace offline, decrypting the tablespace, and then, bringing the tablespace back online, as shown on Example 4-15.

Example 4-15 Offline decryption

```
SQL> alter tablespace DATA_IDX_TBS offline normal;
```

Tablespace altered.

```
SQL> alter tablespace DATA_IDX_TBS encryption offline decrypt;
```

Tablespace altered.

```
SQL> alter tablespace DATA_IDX_TBS online;
```

Tablespace altered.

```
SQL> select TABLESPACE_NAME,ENCRYPTED from dba_tablespaces where  
TABLESPACE_NAME='DATA_IDX_TBS';
```

TABLESPACE_NAME	ENC
DATA_IDX_TBS	NO

Online encryption

Online encryption became available with Oracle version 12.2. To perform an online encryption, the initialization parameter for tablespace encryption must be set to **compatible**.

When the **compatible** initialization parameter is not set in your parameter file, the **compatible** parameter value defaults to 12.2.0 for Oracle Database 12c release 2 (12.2). If you do not set the **compatible** initialization parameter to 12.2.0, you cannot use the new Oracle Database 12c release 2 (12.2) features because your upgraded database is not running in the required compatibility setting for Oracle Database 12c features.

Use the command that is shown in Example 4-16 to set this parameter.

Example 4-16 Set compatibility

```
ALTER SYSTEM SET COMPATIBLE = '12.2.0' SCOPE=SPFILE;
```

The tablespace online encryption is done by using the **ALTER TABLESPACE** command, as shown in Example 4-17.

Example 4-17 Online tablespace encryption

```
SQL> alter tablespace TBS_ENCRY03 encryption online encrypt;
```

Tablespace altered.

```
SQL> select TABLESPACE_NAME,ENCRYPTED from dba_tablespaces where TABLESPACE_NAME='TBS_ENCRY03' ;
```

TABLESPACE_NAME	ENC
TBS_ENCRY03	YES

Note: Online encryption requires at least the same amount of space as the largest data file in the tablespace you are encrypting.

Online decryption

To complete the online decryption on version 12.2 and up, the database must be set to be compatible with 12.2.0.0. Use the command that is shown in Example 4-16 to set this parameter.

The tablespace online decryption is done by using the **ALTER TABLESPACE** command, as shown in Example 4-18.

Example 4-18 Online tablespace decryption

```
SQL> alter tablespace TBS_ENCRY03 encryption online decrypt ;
```

Tablespace altered.

```
SQL> select TABLESPACE_NAME,ENCRYPTED from dba_tablespaces where TABLESPACE_NAME='TBS_ENCRY03' ;
```

TABLESPACE_NAME	ENC
TBS_ENCRY03	NO

It is possible to encrypt, decrypt, or rekey user tablespaces online, including SYSTEM, SYSAUX, and UNDO tablespaces. However, these tasks are not possible for temporary tablespaces.

For more information about encrypting and decrypting tablespaces, see Oracle note TDE 12.2: Encryption and Decryption of Existing Tablespaces ([Doc ID 2255611.1](#)).

4.3 Managing TDE master keys and wallets

It is important to know how to manage the wallet and the TDE master key if any maintenance is needed or any issues must be corrected with the wallet files.

As recommended, the wallet must be backed up in a secure location because it is a critical component. If the wallet that contains the master key is lost or the password is forgotten, the encrypted data becomes inaccessible.

Oracle recommends that the wallet is backed up in several scenarios, including the following examples:

- ▶ Immediately after creation
- ▶ When the master key is regenerated
- ▶ When the database is backed up
- ▶ Before the wallet password is changed

Oracle Wallet Manager or the **orapki** command can be used to update the wallet password, as shown in Example 4-19.

Example 4-19 Changing wallet password by using orapki command

```
orapki wallet welcome2 -wallet /home/oracle/tde12c
```

Oracle always recommends having the wallet outside of the \$ORACLE_BASE directory. It is also recommended to have the directory and folder with access restricted to avoid accidental removals. Therefore, after you create the folder and the wallet, update the permission of the file and the folder, as shown in Example 4-20.

Example 4-20 Changing permission for folder and wallet file

```
[oracle@rdbkora3] :/home/oracle> cd /home/oracle/
[oracle@rdbkora3] :/home/oracle> mkdir -pv /home/oracle/tde12c
[oracle@rdbkora3] :/home/oracle>chown -R oracle:oinstall tde12c
[oracle@rdbkora3] :/home/oracle>chmod -R 700 tde12c
[oracle@rdbkora3] :/home/oracle>chmod 600 ewallet.p12
```

It also is important to have a backup of the keystore, as shown in Example 4-21.

Example 4-21 Backing up the keystore

```
SQL> ADMINISTER KEY MANAGEMENT BACKUP KEYSTORE USING 'tde12c' IDENTIFIED BY
"welcome2" to '/home/oracle';
```

keystore altered.

```
[oracle@rdbkora3] :/home/oracle/tde12c> ls -ltr /home/oracle/ewallet*
-rw-----. 1 oracle oinstall 3995 Jun 24 14:07
/home/oracle/ewallet_2019062418074320_tde12c.p12
[oracle@rdbkora3] :/home/oracle/tde12c>
```

To manage the wallet, such as open and close, can be done by using the commands that are shown in Example 4-22 on page 55.

Example 4-22 Examples on how to manage a wallet

```
alter system set wallet open identified by "password";
alter system set wallet close identified by "password";
alter system set wallet close; -- This is used with auto_login wallets to remove
the cached keys from memory after making the cwallet.sso unavailable
```

The auto_login feature also can be implemented for the wallet, as shown in Example 4-23.

Example 4-23 Wallet files before auto_login is implemented

```
# ls -lthr /home/oracle/tde12c
total 20K
-rw-----. 1 oracle oinstall 2.5K Jun 24 13:16 ewallet_2019062417161925.p12
-rw-----. 1 oracle oinstall 3.8K Jun 24 13:16 ewallet_2019062417164011.p12
-rw-----. 1 oracle oinstall 0 Jun 24 14:16 ewallet.p12.lck
-rw-----. 1 oracle oinstall 0 Jun 24 14:16 cwallet.sso.lck
-rw-----. 1 oracle oinstall 8.4K Jun 24 14:18 ewallet.p12
```

After the wallet is created, auto_login can be enabled, which automatically opens after you create it, as shown in Example 4-24.

Example 4-24 Creating auto_login to the wallet

```
SQL> ADMINISTER KEY MANAGEMENT CREATE LOCAL AUTO_LOGIN KEYSTORE FROM KEYSTORE
'/home/oracle/tde12c/' IDENTIFIED BY welcome2;
keystore altered.
```

After auto_login is implemented, a new file can be seen in the wallet location that is named cwallet.sso. The wallet status and wallet_type also can be checked on the database, as shown in Example 4-25.

Example 4-25 Checking wallet file for auto_login and check wallet status on the database

```
ls -lthr /home/oracle/tde12c
total 32K
-rw-----. 1 oracle oinstall 2.5K Jun 24 13:16 ewallet_2019062417161925.p12
-rw-----. 1 oracle oinstall 3.8K Jun 24 13:16 ewallet_2019062417164011.p12
-rw-----. 1 oracle oinstall 0 Jun 24 14:16 ewallet.p12.lck
-rw-----. 1 oracle oinstall 0 Jun 24 14:16 cwallet.sso.lck
-rw-----. 1 oracle oinstall 8.4K Jun 24 14:18 ewallet.p12
-rw-----. 1 oracle oinstall 8.3K Jun 25 14:18 cwallet.sso <-- .sso file created
for autologin
```

```
SQL> SELECT * FROM v$encryption_wallet;
```

WRL_TYPE WALLET_TYPE	WRL_PARAMETER WALLET_OR_FULLY_BAC	STATUS CON_ID
FILE LOCAL_AUTOLOGIN	/home/oracle/tde12c/ SINGLE	OPEN 0

For more information, see [Oracle Doc ID 1228046.1](#) and [Oracle Database Advanced Security Guide \(E50333-18\)](#).

4.4 Performance

In this section, we describe a demonstration that was conducted in our lab that used Red Hat Enterprise Linux 7 on IBM LinuxOne with two CPU servers that used Oracle 12 Release 2 Enterprise Edition. This demonstration examined the performance effect of TDE and NON-TDE tablespaces.

Under our lab conditions, we tested with low I/O and subsequently low CPU overhead. In your enterprise environment, your results can differ and testing under your load conditions should be performed to evaluate the performance effect to your systems.

The lab test was set up with a low I/O load and most of the logical reads (blocks) from the SGA. Consider conducting proper load testing with realistic loads to determine the TDE performance effects in your environment.

For more information about TDE performance effects, see [Oracle Note Doc ID 1303412.1 Performance Impact of TDE](#).

For our demonstration, we used the AWR Report. For more information about AWR, see 6.2.1, “Oracle Diagnostics Pack” on page 84.

The AWR Load Profile of the demonstration results from the load test that was conducted by using TDE and NON-TDE on LinuxONE encrypted disks and did not result any significant differences in performance in our lab environment.

An AWR load was observed for a NON-TDE tablespace and no performance issue was reported, the results of that report were recorded and are shown in Figure 4-1.

Host	Platform	CPUs	Cores	Sockets	Memory (GB)
rdbkora3.cpolab.ibm.com	IBM zSeries Based Linux		2		125.63
	Snap ID	Snap Time	Sessions	Cursors/Session	
Begin Snap:	8	6/24/2019 16:37	53	0.7	
End Snap:	9	6/24/2019 16:50	57	0.7	
Elapsed:		12.52 (mins)			
DB Time:		12.57 (mins)			
Report Summary					
Top ADDM Findings by Average Active Sessions					
Finding Name	Avg active sessions of the task	Percent active sessions of finding	Task Name	Begin Snap time	End Snap time
Top SQL Statements	1	1.00E+02	ADDM:1539562585_1_9	6/24/2019 16:37	6/24/2019 16:50
Load Profile					
	Per second	Per transaction	Per exec	Cursors/Session	
DB time(s):	1	186.1	0.11	3.96	
DB CPU(s):	0.9	173.6	0.11	3.69	
Background CPU(s):	0	1.2	0	0	
Redo size (bytes):	5173.8	960311			
Logical read (blocks):	124,321 .3	23075530.5			
Block changes:	17.8	3300			
Physical read (blocks):	1.5	285.5			
Physical write (blocks):	1.4	267.8			
Read 10 requests:	1.5	282.5			
Write 10 requests:	0.6	111.5			
Read 10 (MB):	0	2.2			
Write 10 (MB):	0	2.1			
IM scan rows:	0	0			
Session Logical Read IM:	0	0			
User calls:	0.3	47			
Parses (SQL):	3	558			
Hard parses (SQL):	0.3	62.5			
SQL Work Area (MB):	0.9	160			
Logons:	0.1	17.8			
Executes (SQL):	8.8	1625.3			
Rollbacks:	0	0			
Transactions:	0				

Figure 4-1 Results recorded from an AWR Load Profile of non-TDE tablespace

An AWR report was then run after the TDE tablespace is enabled and the database remained with same behavior. The results were recorded and, as shown in Figure 4-2, no issues were encountered.

Host	Platform	Cpus	Cores	Sockets	Memory (GB)
rdbkora3.cpolab.ibm.com	IBM zSeries Based Linux	2			125.63
	Snap ID	Snap Time	Sessions	Cursors/Session	
Begin Snap:	8	6/24/2019 16:37	53	0.7	
End Snap:	9	6/24/2019 16:50	57	0.7	
Elapsed:		12.52 (mins)			
DB Time:		12.57 (mins)			
Report Summary					
Top ADDM Findings by Average Active Sessions					
Finding Name	Avg active sessions of the task	Percent active sessions of finding	Task Name	Begin Snap time	End Snap time
Top SQL Statements	1	1.00E+02	ADDM:1539562585_1_9	6/24/2019 16:37	6/24/2019 16:50
Load Profile					
	Per second	Per transaction	Per exec	Cursors/Session	
DB time(s):	1	188.5	0.09	3.38	
DB CPU(s):	0.9	173.1	0.09	3.1	
Background CPU(s):	0	1.1	0	0	
Redo size (bytes):	4740.2	890027			
Logical read (blocks):	123395.6	23169132.8			
Block changes:	13.8	2599.5			
Physical read (blocks):	1.9	364			
Physical write (blocks):	1.6	292			
Read 10 requests:	1.9	360.5			
Write 10 requests:	0.7	138			
Read 10 (MB):	0	2.8			
Write 10 (MB):	0	2.3			
IM scan rows:	0	0			
Session Logical Read IM:	0	0			
User calls:	0.3	55.8			
Parses (SQL):	3.1	578.3			
Hard parses (SQL):	0.5	98			
SQL Work Area (MB):	0.6	102.4			
Logons:	0.1	19.3			
Executes (SQL):	10.6	1998			
Rollbacks:	0	0			
Transactions:	0				

Figure 4-2 Results recorded from an AWR Load Profile of TDE tablespace

ORAchk tool

During our performance test, we used the ORAchk tool to check whether our settings were adequate for the database configuration, as recommended by Oracle.

Oracle also recommends running ORAchk - Health Checks for the Oracle Stack on regular basis or during any significant changes, such as Oracle Release Updates (RU) and any other feature usage, such as TDE or Advanced Compression.

As a best practice, attempt to get a score as close to 100 as possible and fix any reported failures or warnings when running ORAchk. It also is recommended that you use the most current version of ORAchk when running ORAchk - Health Checks.

The results of the ORAchk Assessment Report from our Demo Lab database server is shown in Figure 4-3. During our test, we found that our system had a score of 90 out of 100 after running ORAchk.

Assessment report results:	System Health Score is 90 out of 100
Summary	
OS/Kernel Version	Linux on System z OELRHEL 7 3.10.0- 957.21.2.el7.s390x
GRID Home	/ u0 1/app/oracle/grid/ 12.2.0.1
DB Home - Version - Names	/ u01/app/oracle/product/ 12.2.0.1/dbhome_1 - 12.2.0.1.0 - orcl
Database Server	rdbkora3
orachk Version	12.1.0.2. 7 20160526
Collection	orachk_rdbkora3_orcl_062519_1043 34.zip
Duration	3 mins, 14 seconds
Executed by	oracle
Arguments	- no upgrade
Collection Date	6/25/2019 10:43
Note! This version of orachk is considered valid for days from today or until a new version is available	

Figure 4-3 Recorded results from the ORAchk Assessment Report

The ORAchk Assessment Report includes suggestions for fixing any known issues to achieve a score of 100. This process helps to bring the database closer to the 100 score as recommended by Oracle. It also can help to avoid issues in the future.

For more information about how to use and install the tool, see [ORAchk - Health Checks for the Oracle Stack \(Doc ID 1268927.2\)](#).

4.5 Using Oracle TDE on Oracle RAC

A software keystore is a container for the TDE master encryption key, and it is in the software file system. In an Oracle RAC environment, Oracle recommends that you place the keystore on a shared file system, such as Oracle Automatic Storage Management (ASM) or NFS.

Managing the wallet and the keystore, such as open, close, or rekeying, can be performed on any of the instances of the Oracle RAC database.

If a shared file system is used, the ENCRYPTION_WALLET_LOCATION parameter must be set on sqlnet.ora file for all nodes. All steps that are described in 4.1.3, “Configuring and administering Transparent Data Encryption” on page 44 must be run on the Oracle RAC database.

For more information, check the Oracle Transparent Encryption document.

4.6 Summary

In this chapter, we described and demonstrated how to use Oracle Transparent Data Encryption on Oracle 12cR2.

Oracle TDE was used to configure the database data layer. These results show that Oracle TDE on version 12cR2 transparently encrypted databases. Under our lab conditions, with low I/O and subsequently low CPU overhead, no performance difference was found.

In your enterprise environment, your results might differ and testing under your conditions should be performed to evaluate the performance affect to your systems.



Successful consolidation project: From sizing to migration

This chapter focuses on the practical aspects of consolidation and guides you through the steps to achieve a successful IT optimization operation on LinuxONE with your Oracle databases.

This chapter includes the following topics:

- ▶ 5.1, “Delimitation of the project scope” on page 62
- ▶ 5.2, “Sizing” on page 63
- ▶ 5.3, “Proof of concept” on page 67
- ▶ 5.4, “Migration” on page 71
- ▶ 5.5, “Successful production” on page 79

5.1 Delimitation of the project scope

The first step of the consolidation project process is to delimitate the scope of consolidation: we need to choose which servers will be consolidated. Not all the servers are good candidates for consolidation. To select the right servers, consider the issues that are described next.

5.1.1 Certification and support

The middleware and software must be certified on the IBM LinuxONE system we plan to use. For Oracle database, the support can be checked on the My Oracle Support website (<https://support.oracle.com>). This website contains the latest certification information.

Note: Oracle does not certify their software to specific hardware. Oracle certifies their products to operating systems (OSs) and requires the hardware vendor support the wanted OS. In our case, we must check support for Oracle database on IBM: Linux on System z SLES (and the version) or IBM: Linux on System z Red Hat (and the version).

If the database is part of a specific application, we also must ensure that this application is certified (at least in split tier) on IBM LinuxONE. For example, Oracle e-Business Suite, Oracle Siebel, Oracle BIEE and Hyperion are certified in split tier (the database part) and can run on LinuxONE.

5.1.2 Best candidates for consolidation

The best candidates to be consolidated on IBM LinuxONE are low-end, under-utilized, and non-virtualized servers. Depending upon the workload in your current environment including level of virtualization, CPU utilization and other factors with these kind of servers, you can reach an interesting consolidation ratio (up to 1:10 and above). These ratios can lead to a better business case. Best candidates are also servers with complementary peaks (peaks of activity spread across large windows). Servers with a large quantity of concurrent peaks do not receive the best of consolidation capabilities of IBM LinuxONE.

5.1.3 Non-functional requirements

Architecture design must take into account and prioritize the following main non-functional requirements, depending on business requirements:

- ▶ Availability
- ▶ Reliability
- ▶ Scalability
- ▶ Security
- ▶ Performance
- ▶ System management

5.1.4 Business value analysis

A business value analysis is recommended to take into account the total cost of acquisition. The study includes the total cost of ownership for hardware, software, maintenance, administration, floor space, power, cooling, and so on. The results of this study demonstrate the financial advantages of running Oracle on LinuxONE. This study is done by specific teams within IBM. For more information about conducting this study, contact your IBM representative.

5.2 Sizing

Sizing is key for a successful consolidation project. This exercise determines, based on the existing environment, the quantity of resources (such as CPU and memory) that are needed to run Oracle databases on the new IBM LinuxONE servers.

In this section, we describe the various facets that should be examined when your Oracle databases are migrated to the LinuxONE servers.

5.2.1 CPU

The number of needed LinuxONE processors can be estimated with the help of IBM by using IBM SURF and IBM SCON tools. For this evaluation, you need the following information:

- ▶ Details about the source servers (vendor or manufacturer, server and type model, number of chips and cores, and model and speed of chips).
- ▶ The average and peak CPU usage (estimation).
- ▶ The type of workload (for example, production database).

This information provides you with a rough estimation of needed CPU. If you want to get a more accurate sizing, you can collect the real performance data (SAR, NMON, Perform for Windows, or an equivalent product) collected for each server at the same time during 24 - 48 hours with an interval of 5 - 10 minutes (to ensure we take into account activity peaks).

Note: It is important to collect this data during the peak of activity (for example, end of month batch, end of year marketing campaign) to ensure that we measure the peak of workload activity. An example of CPU sizing is shown in Figure 5-1 on page 64.

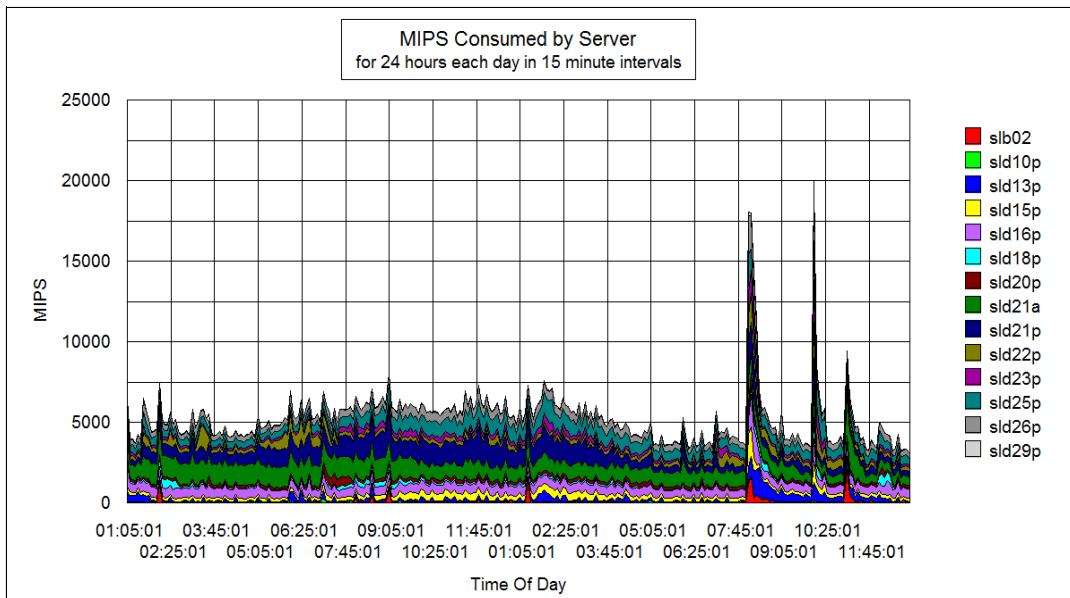


Figure 5-1 CPU sizing example

For more information about this CPU estimation, contact your IBM representative or IBM Techline.

5.2.2 Memory

On LinuxONE servers, it is recommended to carefully evaluate the memory required for the Linux guest in a virtualized environment. Oversizing memory is wasteful and may creating additional overhead.

It is not unusual in a distributed environment to allocate more memory than what is needed by the Oracle database workloads. You must ensure that the source database memory is optimized. You might want to use the SGA target advisory and PGA target advisory sections of the Oracle AWR reports to estimate the memory that is used by the Oracle database.

After verifying that the source system memory is optimized, the suggestion is to use the same amount of memory on the target database as on the source database. To get a representative Oracle AWR report, you must take your snapshots during the peak period of your workload. You must determine the peak period within the most loaded day of the week (or month at certain periods). You can run an ADDM report from your Oracle Enterprise Manager Cloud Control.

You can find the memory that is used by Oracle in AWR report in the Memory statistics section, as shown in Figure 5-2.

Memory Statistics		
	Begin	End
Host Mem (MB):	7,872.4	7,872.4
SGA use (MB):	2,048.0	2,048.0
PGA use (MB):	405.3	431.7
% Host Mem used for SGA+PGA:	31.16	31.50

Figure 5-2 Memory statistics in the Oracle AWR report

You can find the quantity of memory that is allocated to the Oracle database in AWR report in init.ora section, as shown in Figure 5-3.

parallel_min_servers	0
pga_aggregate_limit	0
pga_aggregate_target	1073741824
plsql_code_type	INTERPRETED
processes	200
recyclebin	off
sec_case_sensitive_logon	FALSE
service_names	GRL1, ebs_patch
session_cached_cursors	500
sessions	400
sga_target	2147483648

Figure 5-3 Quantity of memory allocated

You can check the AWR reports advisory section to ensure SGA and PGA are optimized. For SGA, you find the information in the AWR SGA Target Advisory section, as shown in Figure 5-4.

SGA Target Advisory			
SGA Target Size (M)	SGA Size Factor	Est DB Time (s)	Est Physical Reads
512	0.25	99,709	973,461
1,024	0.50	222,838	14,505,834
1,536	0.75	119,810	4,652,365
2,048	1.00	104,364	973,461
2,560	1.25	103,414	713,644
3,072	1.50	103,028	619,121
3,584	1.75	102,934	619,121
4,096	2.00	102,934	619,121

Figure 5-4 SGA optimization

For more information about PGA, see the AWR PGA Target Advisory section, as shown in Figure 5-5.

PGA Memory Advisory							
• When using Auto Memory Mgmt, minimally choose a pga_aggregate_target value where Estd PGA Overalloc Count is 0							
PGA Target Est (MB)	Size Factor	W/A MB Processed	Estd Extra W/A MB Read/Written to Disk	Estd PGA Cache Hit %	Estd PGA Overalloc Count	Estd Time	
128	0.13	63,143.90	7,658.54	89.00	250	30,011,946	
256	0.25	63,143.90	6,273.03	91.00	214	29,424,653	
512	0.50	63,143.90	840.69	99.00	0	27,121,974	
768	0.75	63,143.90	840.69	99.00	0	27,121,974	
1,024	1.00	63,143.90	840.69	99.00	0	27,121,974	
1,229	1.20	63,143.90	840.69	99.00	0	27,121,974	
1,434	1.40	63,143.90	840.69	99.00	0	27,121,974	
1,638	1.60	63,143.90	840.69	99.00	0	27,121,974	
1,843	1.80	63,143.90	840.69	99.00	0	27,121,974	
2,048	2.00	63,143.90	840.69	99.00	0	27,121,974	
3,072	3.00	63,143.90	840.69	99.00	0	27,121,974	
4,096	4.00	63,143.90	840.69	99.00	0	27,121,974	
6,144	6.00	63,143.90	840.69	99.00	0	27,121,974	
8,192	8.00	63,143.90	840.69	99.00	0	27,121,974	

Figure 5-5 PGA optimization

For the dedicated server processes, you can calculate the memory that is needed, as shown in the Example 5-1.

Example 5-1 Calculating needed memory

Memory needed for dedicated server processes = Max(logons current in AWR) X memory used per thread

(as a rule of thumb, dedicated connections use 4.5 MB per connection, this number is application dependent and can vary a lot).

You can find the concurrent logons in the Instance Activity Stats - Absolute Values section of the AWR report, as shown in Figure 5-6.

Instance Activity Stats - Absolute Values		
Statistic	Begin Value	End Value
logons current	107	111
opened cursors current	1,162	1,274
session cursor cache count	225,553	231,724
session pga memory	294,891,624	311,412,040
session pga memory max	578,097,800	603,317,448
session uga memory	8,843,571,560	9,032,453,632
session uga memory max	40,147,807,416	41,175,243,056
workarea memory allocated	10,872	10,872

Figure 5-6 Concurrent logons

5.2.3 I/O

You can find the I/O information in AWR reports in the load profile section, as shown in Figure 5-7. Physical reads and Physical writes values help you to size the I/O for this workload on IBM LinuxONE.

Load Profile				
	Per Second	Per Transaction	Per Exec	Per Call
DB Time(s):	0.7	0.7	0.04	0.04
DB CPU(s):	0.2	0.2	0.01	0.01
Background CPU(s):	0.0	0.0	0.00	0.00
Redo size (bytes):	36,288.1	36,852.1		
Logical read (blocks):	32,290.8	32,792.7		
Block changes:	136.7	138.8		
Physical read (blocks):	27.4	27.8		
Physical write (blocks):	4.9	5.0		
Read IO requests:	6.7	6.8		
Write IO requests:	1.8	1.8		
Read IO (MB):	0.2	0.2		
Write IO (MB):	0.0	0.0		
IM scan rows:	0.0	0.0		
Session Logical Read IM:				
User calls:	15.9	16.2		
Parses (SQL):	6.0	6.1		
Hard parses (SQL):	0.1	0.1		
SQL Work Area (MB):	0.7	0.7		
Logons:	0.6	0.6		
Executes (SQL):	17.6	17.9		
Rollbacks:	0.5	0.5		
Transactions:	1.0			

Figure 5-7 I/O information

5.3 Proof of concept

Before migrating the production on IBM LinuxONE, it is strongly recommended that a Proof of Concept (PoC) is run. This PoC allows you to validate the architecture from a functional point of view. If a performance test is done, the PoC confirms the sizing.

5.3.1 PoC qualification

Qualification is key for a successful and useful PoC. During this phase, we must determine the objectives, delimitate the scope of the PoC, set the Key Performance Indicators, design the architecture, define the type of workload we are going to evaluate, and define the test plan.

Objectives

The first question to answer when talking about a PoC is: What are my objectives? This question helps to determine the objectives from a business point of view that are translated into a technical point of view. The following objectives can be included:

- ▶ Validate Oracle on IBM LinuxONE delivers sufficient throughput to face the 200 business growth.
- ▶ Check that the response times to ensure the SLA meets user requirements.
- ▶ Ensure Oracle on IBM LinuxONE runs the same as it did on another platform in terms of functionalities.

The objectives orient the choice of functionality and type of workload to test, and the architecture to put in place for the test.

Scope

During a PoC, we do not test all of the functionalities or the full workload. Instead, we focus on a subset of functionalities and a representative workload.

The scope can include the following tasks:

- ▶ Test the duration of the end of year batch (critical for my company) on a subset of accounts.
- ▶ Evaluate the throughput for a specific type of transaction for 1000 concurrent users.
- ▶ Test the high availability mechanism if Operating System maintenance is needed.
- ▶ Test the full disaster recovery solution if a site is lost.

Key performance indicators

Asking the question “I want to check that the Oracle on LinuxONE servers is faster than my current environment” is not enough. Measurable metrics are needed to evaluate the success of the PoC. Collect performance measurements on the current environment.

The following metrics can be included:

- ▶ Response time for specific OLTP/Datawarehouse queries
- ▶ Throughput in terms of transactions per minute
- ▶ Time to recover after a failure
- ▶ Number of reports that is generated for one hour
- ▶ Duration of an export/import operation for a migration

Architecture

After the objectives, scope, and KPIs are set, we can design the architecture of the PoC. We recommend that this architecture is the closest possible than the final architecture, especially for Operating System version and database version. For performance measurements, dedicate the resources to the LPAR.

This architecture encompasses the following elements:

- ▶ Hardware part, including type of LinuxONE models, storage subsystem, and network
- ▶ Virtualization, if used
- ▶ Operating system
- ▶ Software, including database part and application server part

Type of workload

We strongly recommend the use of a real representative workload. For this workload, IBM Infosphere Optim™ Capture and Replay or Oracle Real Application Testing can be useful for OLTP workload. We do not recommend the use of generic workload generators, such as Swingbench or Hammer DB, because they produce generic results. Therefore, with these kind of tools, we do not handle the potential internal contentions so we cannot guarantee the scaling.

If you cannot use a subset of real workload (for example, if for regulation reasons you cannot bring the data, even anonymized, in a Benchmark Center to run your PoC), we suggest the use of benchmark tools, such as IBM Rational® Performance Testing.

If you plan to use real client data, you must decide the appropriate way to access the data. For example, for a subset of production data, export/import is the simplest way to migrate the data onto LinuxONE servers.

Test plan

With all the stakeholders of the project, you can work on the test plan, including validating the high-level time schedule and resources (IT and people) that are required for this PoC.

The high-level time schedule takes into account the following tasks:

- ▶ Hardware
- ▶ Execution of the various test cases, including tuning
- ▶ PoC report redaction

The availability of the resources during the PoC (especially people) is a key factor for success. For example, avoid holiday periods to run your PoC because you can be stuck in the middle of the project because of the lack of a critical skill.

5.3.2 Execution

When the qualification and preparation is complete, execution is the next step. This process includes the following tasks:

- ▶ Hardware and software setup
- ▶ Software setup
- ▶ Writing a test procedure
- ▶ Monitoring
- ▶ Tuning

Hardware setup

This phase includes the following tasks:

- ▶ Define the I/O connectivity
- ▶ Define the LPAR

If performance is part of the PoC, it is recommend that the test environment is isolated in an LPAR and dedicate the resources CPU, RAM, IOs, and network.

- ▶ If virtualization is a part of the PoC, define the IBM z/VM® installation and Linux VM.
- ▶ Install the operating system

Check that all of the packages that are needed for Oracle are available with RPM checker. Also, check the kernel parameters and change the I/O scheduler, if needed. Change user limits.

- ▶ Validate the storage validation

After the SAN is configured, it is recommend that the Orion disk I/O utility is run before any database is created to ensure that the I/O subsystem delivers as expected.

- ▶ Enable multipathing for SCSI/FCP or Hyperpav for ECKD/IBM FICON® devices.
- ▶ Validate network latency validation by using simple commands, such as **ftp** or **put**.

Software setup

Once the Oracle database is installed, we move on to installation of the application. If the architecture was designed in split-tier, the application is installed on other servers than LinuxONE.

Remember to install the latest level of patches to the database and to change the parameters, especially the `filesystemio_options` parameter. If file systems for Oracle database files are used, we recommend setting the Oracle parameter `filesystemio_options='setall'`, to enable both asynchronous and direct I/O for Oracle database files.

Monitoring

Before starting the testing, the following monitoring tools must be installed:

- ▶ PerfToolkit and IBM Wave for z/VM
- ▶ Storage monitoring tool
- ▶ NMON or any other Linux monitoring tool, such as SAR, VMSTAT, and IOSTAT
- ▶ Enterprise Manager Cloud Control (if you do not have this tool, Enterprise Express Cloud Control provides high-level monitoring) to monitor the Oracle database
- ▶ AWR Oracle statistics and alert log

The test procedure includes monitoring.

Writing a test procedure

A test procedure allows you to have automated and repeatable test cases.

You adapt the test procedure depending on what you want to test, based on the key performance indicators that are described in “Key performance indicators” on page 67. This test procedure can include the following tasks:

- ▶ Restore database, especially after a parameter change.
- ▶ Check DB size to ensure the restore process completed successfully.

- ▶ Set up the changes at the infrastructure level (various definitions of CPU, memory, and so on). Also, check CPU and memory setup in cat /proc/cpuinfo or meminfo.
- ▶ Check Oracle database parameters and adjust them, if needed (for example, number of processes).
- ▶ Gather system and schema statistics.
- ▶ Start Linux monitoring.
- ▶ Start AWR sampling.
- ▶ Launch the test:
 - Run test twice: 1 cold test and 1 warm test
 - Gather schema statistics between both (to assess the data growth)
- ▶ Collect statistics and results for analysis.
- ▶ Gather data in an independent manner.

Tuning

After each test case, you likely must tune the environment (infrastructure, including the operating system, Oracle database, and application). Only one change should be made at a time. Otherwise, you cannot evaluate which parameter brought the change. Keep a log book of each test case change and the results.

For more information about tuning advice, see Chapter 6, “Performance management for IBM z/VM, Linux, and Oracle on LinuxONE” on page 81.

5.3.3 Analyzing and reporting

In this section, we describe data collection and communicating the results.

Data collection

Each run generates a set of data that you must collect and gather in an independent manner. A recommendation is to keep a trace of all the tests in a table. The table can include the following information:

- ▶ Number of the test and run
- ▶ Hardware configuration (CPU, memory, and so on)
- ▶ Parameters that are involved
- ▶ Results:
 - Transactions per minute
 - Duration of the run
 - Number of reports generated

For each test, you have an associated folder that contains the monitoring data and a proof of results (for example, a screen shot of the duration of the run).

With this classification methodology, your results are not contested. Also, you can quickly spot the “golden run”, which is the test for which the best results were found, with the appropriated parameters.

Communicating the results

At the end of the PoC, a report is created to share the result with all the stakeholders.

It is suggested that the following information is included in this report:

- ▶ An executive summary that includes why this PoC was run and its main objectives
- ▶ Architecture design
- ▶ Testing methodology
- ▶ Test results summary
- ▶ Conclusion
- ▶ Appendix that includes monitoring data for the main tests

At the end of the presentation, the stakeholders will be in a position to decide (from a technical point of view based on the PoC results) the next steps for consolidation project on IBM LinuxONE servers.

5.4 Migration

After a decision is made to run your Oracle database environment on LinuxONE, it is time for the migration operation. There are several ways to migrate an Oracle database on LinuxONE. This migration, which is often a standard operation for most of the DBAs, can be risky if it is under evaluated.

Note: This section is based on our experiences and does *not* replace any IBM or Oracle documents.

For more information about migration, see *Practical Migration from x86 to LinuxONE*, SG24-8377.

5.4.1 Considerations before migration

In this section, we describe the following points that must be considered when an Oracle database is migrated to LinuxONE:

- ▶ Downtime
- ▶ Technical compatibility
- ▶ Application compatibility
- ▶ In-house administration scripts
- ▶ Disk space requirements
- ▶ Skills

Downtime

Each migration leads to a database downtime. Depending on the technique, this downtime can be a minimum of a few minutes to more than a day. For critical applications that must be always available, downtime is the main criteria for clients to choose the appropriate technique.

Technical compatibility

Depending on the technique you choose, you might encounter some of the following technical limitations:

- ▶ Endianness

Endianness describes how the bits are organized as seen from the outside. Depending on the platform, this endianness can be “Little Endian” or “Big Endian”. Big-endian means that numbers are stored with the most significant data earlier in memory. Conversely little-endian means that numbers are stored with the least significant data earlier in memory. IBM LinuxONE servers are “Big Endian”, where the most significant byte is put first.

Some cross platform migration methods, such as Transportable Databases, require the same endianness.

Also, some migration methods, such as Transportable Tablespaces, must convert the bits if the Endianness is different. This operation takes time and might lead to another migration technique being used.

Use the following command to check the Endianness for your platform:

```
select platform_id, platform_name from v$database;
```

- ▶ Objects

Some objects cannot be migrated with certain techniques, including the following examples:

- Export/Import Data Pump utilities cannot be used for XML types
- Streams cannot handle secure files, Character Large Object(CLOB), National Character Large Object (NCLOBs), Binary Large Object (BLOB) and other types

Application compatibility

Although Oracle database is supported on LinuxONE, you must check the supported configuration when this database is embedded into an application.

For example, Oracle database as part of Oracle E-Business Suite is supported on LinuxONE, whereas Oracle database as part of Oracle Financial Services (formerly I-FLEX) is not supported.

Tip: Always check the supported configuration on My Oracle Support to avoid the support issues.

In-house administration scripts

Be aware that you must customize the administration and automation scripts (for example, for backup and recovery operations) in your new environment. These scripts must be tested before the migration.

Network

If the migration technique uses the network (for example, replication techniques), you must ensure that the network is efficient in terms of bandwidth and latency; otherwise, this potential bottleneck dramatically increases the migration operation duration. The chosen technique must also consider the location of the servers, target, and source. Constraints can include that these factors are geographically dispersed or they cannot communicate with each other.

Disk space requirements

Some migration techniques need the disk space for staging or dump files. You should consider this technique if you have storage constraints, as shown in the following examples:

- ▶ If the migration is done by using Export/Import, you need some space on the source system to store the dump files.
- ▶ If the migration is done by using Recovery Manager (RMAN), you need some space to store the redo logs that are created after the copy starts.

Skills

Database migration can be considered a risky operation. Depending on the products and techniques that are used in your environment, you might prefer one technique over another.

Tip: Whenever possible, perform the migration with known products to mitigate the risks.

5.4.2 Available techniques for cross-platform migration

The following main cross-platform migration techniques are available:

- ▶ Export/Import with Data Pump
- ▶ Transportable Tablespaces
- ▶ Transportable database (if same Endianness)
- ▶ Create Table As Select
- ▶ IBM Infosphere Data Replication
- ▶ Oracle Streams
- ▶ Oracle DataGuard
- ▶ Oracle GoldenGate

In this section, we describe the two most-utilized cross-platform migration techniques: Export/Import with Data Pump and Transportable Tablespaces. Then, we provide a brief overview of the other techniques.

Export/Import with Data Pump

Export/Import with Data Pump is the classic method that is used to migrate a database. The Export/Import Data Pump utility is more efficient than the standard Export/Import utility. When possible, we recommend the use of Data Pump. This section describes only the Oracle Data Pump utility.

Export and Import utilities transfer data objects between two databases, independently from hardware and software configurations. Objects can be tables, indexes, comments, grants, and so on. With Export, objects are extracted (tables first, then other objects, if any) and the extracted data is written into an Export dump file (Oracle binary-format). The Import utility takes the tables data and definitions from the dump file.

Migration technique

The migration technique includes the following overall steps:

1. Export the database by using the Export utility from the source (the dump file can be on a disk or on a tape).
2. Transfer the dump file to the target by using FTP, SFTP, RCP, or physically if there is no communication between the source and target servers (for example, dump files on tape).
3. Create the target database.
4. Import the data with Import utility in the new database.

5. Import the metadata to complete the full database structure.
6. Check the consistency.

Advantages

This technique offers the following advantages:

- ▶ Can be used across any platform
- ▶ No conversion is needed
- ▶ Parallelism helps speed up the operation
- ▶ Fine-grained object selection for inclusion or exclusion
- ▶ Ability to restart without loss of data
- ▶ Database can be migrated and upgraded in one operation
- ▶ Space estimation from a storage point of view

Limitations

This technique has the following limitations:

- ▶ Dump files that are generated by the Data Pump Export utility are not compatible with dump files that are generated by the original Export utility.
- ▶ Downtime can be significant for large databases.

Transportable Tablespace migration

The Transportable Tablespace migration works within the framework of Data Pump and RMAN. RMAN is a backup and recovery manager that is provided by Oracle and does not require any separate installation. RMAN stores metadata in the control files of the target database and, optionally, in a recovery catalog schema in an Oracle database. RMAN provides block-level corruption detection during backup and restore.

Main migration steps

This technique includes the following overall steps:

1. Convert tablespaces into read-only.
2. Use Data Pump to move the metadata of the objects.
3. Convert the tablespaces to the correct Endianness (if necessary) with RMAN Convert.
4. Create the database on the target system.
5. Transfer the data files to the target server.
6. Import the tablespaces.
7. Import the metadata.
8. Check the consistency.

Advantage

The main advantage in using this technique is that it can be used across different Endianness. If Endianness are the same, we can use the Transportable Database feature.

Limitations

This technique includes the following limitations:

- ▶ Requires a larger time investment to test the migration and to develop methods of validating the database and application. Consider whether the extra testing time, complexity, and risk that is involved are worth the potential to reduce migration downtime.
- ▶ Requires a higher level of skills for the database administrator and application administrator that is compared to the use of Data Pump full database Export and Import.

- ▶ Does not transport objects in the SYSTEM tablespace or objects that are owned by special Oracle users, such as, SYS or SYSTEM. Applications that store some objects in the SYSTEM tablespace or create objects as SYS or SYSTEM require more steps and increase the complexity of the platform migration.
- ▶ Self-contained Oracle TableSpaces can be moved between platforms only.
- ▶ If the destination database contains a tablespace with the same name, you must rename or drop it.
- ▶ Triggers, packages and procedures must be re-created on the target database.
- ▶ Only user tablespaces can be transported. System and SYSAUX objects must be created at the target.
- ▶ Tablespaces must be self-contained. (Materialized views or contained objects, such as, partitioned tables, are not transportable unless all of the underlying or contained objects are in the tablespace set.)
- ▶ The source and target databases must have the same character set.
- ▶ All system privileges are not imported into the upgraded database.
- ▶ Resetting sequences and recompiling invalid objects might be needed.
- ▶ The Transportable Tablespaces migration approach does not allow for the redesign of the database (logical and physical layout) as part of the migration.
- ▶ Fragmented data still exists.

Create Table As Select instruction

With this instruction, you can copy data from a source database to a target database by using the network because of a database link. You do not need extra space for dump files because you copy directly from the source to the target. However, as you use the network, the network traffic can be significant and slow down other operations, depending on the size of the tables.

This technique can be used to migrate one or several tables, but we cannot envisage to use it for an entire database. This technique also can be used with other techniques (Export/Import or Transportable Tablespaces, for example).

IBM Infosphere Data Replication platform

IBM InfoSphere® Data Replication is a data replication platform that is easy to use, highly scalable, and enterprise ready. It can provide trusted data synchronization (including log-based change data capture capabilities) to replicate information between heterogeneous data stores in near realtime. The software provides replication with transactional integrity to support big data integration, continuous availability, consolidation, warehousing, and business analytics initiatives. It also supports zero-downtime migrations and upgrades.

InfoSphere Data Replication provides the following benefits:

- ▶ Faster, simpler data replication that maintains transactional integrity and consistency for enterprise data volumes.
- ▶ Centralized, easier-to-use platform that helps simplify deployment and data integration processes.
- ▶ Support for heterogeneous data that moves information, at a lower cost, between a wide range of systems and data sources.

This product requires extra licences.

For more information, see the [IBM InfoSphere Data Replication website](#).

Oracle Streams

Oracle Streams use log data that is captured with LogMiner-based technology on the source system as its capture mechanism, from which logical change records (LCR) are generated. A stream allows transactions to be propagated to one or several databases. Oracle Streams can specify rules at multiple levels of granularity: database, schema, and table. Oracle Streams capture changes from the redo logs in a source database, the changes are staged, and then are propagated into the target database.

This product is used to propagate information among distributed databases, but the mechanism can be used for migration as well with other techniques (Export/Import, for example).

The main advantage of using Oracle Streams is that it provides minimal downtime. Minimal downtime (reconnecting the users only) and failback are possible because the source is untouched.

To use this technique, set up is significant, and some data types are not supported for capture processes. Therefore, an Export/Import of these object types also is required; for example, SecureFile CLOB, NCLOB, BLOB, BFILE, Rowid, User-defined types (including object types, REFs, arrays, and nested tables), XMLType are stored object relationally or as binary XML.

Oracle GoldenGate

By using Oracle GoldenGate, you can move data between like-to-like and heterogeneous systems, including different versions of Oracle Database, different hardware platforms, and between Oracle and non Oracle databases. The software performs real-time, log-based change data capture (CDC) and can move large volumes of transactional data between heterogeneous databases with low latency and minimal footprint.

This technique allows near zero downtime and works across platforms without conversion.

However, there are associated extra license costs. Also, you must consider memory and CPU overhead (3% - 5% CPU effect of Oracle GoldenGate Replication on the source system, depending on the number of redo logs that are generated).

Some data types, such as ANYDATA, BFILE, and TIMEZONE_REGION, are not supported.

Oracle Dataguard Heterogeneous Primary and Physical Standbys

Data Guard depends on the Log Writer process or the Archive process to capture and send redo data or logs to the standby site. This technique can be efficient for other platforms, but there are strong limitations for the IBM LinuxONE platform in terms of cross platform compatibility. For IBM LinuxONE, the supported migrations are with IBM LinuxONE and IBM Linux on Power platforms only.

For more information, see the My Oracle Support note *Dataguard support for Heterogeneous Primary and Physical Standbys in Same Dataguard Configuration* (ID 413484.1).

5.4.3 Considerations when migrating from File System to ASM or vice versa

The following organization types are available for your database files:

- ▶ File System
- ▶ Automatic Storage Management (ASM)

ASM is built into the Oracle kernel and provides the DBA with a way to manage many disk drives for single and clustered instances of Oracle. ASM is a file system/volume manager for all Oracle physical database files (such as, data files, online redo logs, control files, archived redo logs, RMAN backup sets, and SPFILEs). All of the database files (and directories) to be used for Oracle are contained in a disk group.

If you decide to change the way your Oracle database files are organized, you can use RMAN Backup/Restore capabilities. For more information about migrating databases to and from ASM by using recovery manager, see the [Oracle documentation website](#).

5.4.4 Best practices

For a successful migration operation, we strongly recommend that you review the best practices that are presented in this section. Most of these practices are based on common sense and some are based on experiences.

Test

Tests are the most critical part of any migration operation.

You should not underestimate the time that is needed for the tests in the migration planning. Even if the migration lasts a few hours, tests can last several weeks.

As part of your test plan, you must ensure that you have all the resources that are needed for the test phases and the migration in terms of staff.

Backup and monitoring procedures on the new server also must be checked before the final migration.

The validation phase after the migration is critical before switching to the new environment.

A migration also should be considered as a real and potentially complex project. Therefore, project management is key for a successful migration.

Sizing

For more information, see 5.2, “Sizing”.

Sizing is the only way to ensure that the correct quantity of resources in terms of CPU, memory, I/O, storage is available to run the database on the new platform.

The more accurate the inputs for the sizing are, the more accurate the resource estimation.

Performance measurement before and after migration

To compare the performance before and after migration, we advise that you take real performance measurements. You want to avoid the possible user complaint that the system was running faster before the migration to this new system without having real figures to compare against.

The following performance measurements can be used:

- ▶ Operating system level:
 - SAR
 - NMON
 - VMSTAT
 - IOSTAT
- ▶ Oracle database level: AWR reports
- ▶ Application level:
 - Duration of some batches
 - Response time for some complex user transactions

These performance measurements provide the baseline before any migration or operation. After the database is migrated to the target system, we recommend that you repeat the performance measurements on the target system to compare the results before and after the migration. If some performance issues appear, the new system might need some tuning. For more information, see Chapter 6, “Performance management for IBM z/VM, Linux, and Oracle on LinuxONE” on page 81.

Preparing the data before migration

Before any migration, we recommend that you compress the database as much as you can (including reorganization and data compression), which reduces the data to migrate and the possible downtime. The following tasks are part of preparing the data:

- ▶ Determine the invalid objects in the database

Before migration, you must determine the invalid objects so that you can compare this figure to after the migration. This list of invalid objects facilitates any diagnostic tasks.

Use the command that shown in Example 5-2 to determine the invalid objects.

Example 5-2 Determine invalid objects

```
SELECT object_name, owner, object_type
FROM dba_objects
WHERE status='INVALID'
ORDER BY object_type;
```

- ▶ Rebuild the indexes

To reduce the migration duration (especially if you choose Export/Import utilities), we recommend that you limit the quantity of data to migrate. For example, you can import in two operations: Import the data or import the metadata.

This limitation significantly reduces the migration time, mainly because the indexes and the data are not imported at the same time.

- ▶ Log during migration

If you plan to use Export/Import for your migration, we suggest you use one of the following methods to disable the logging for archiving:

- Disable the archive logs during the import, if possible (for this option, the database must be stopped to be placed in “mount” state).
- Disable the logging directly in to the tablespace by using the **alter tablespace...nologging** command.
- Set the parameters _disable_logging boolean TRUE in the init.ora.

- ▶ Redo logs

We also suggest the following tasks to simplify the Redo logs management:

- Minimize the number of Redo log members per thread (one member is sufficient).
- Increase the size of the Redo logs, if possible (to avoid useless switches).

Take advantage of IBM LinuxONE

To get the best level of performance, you need the same degree of parallelism for Export and for Import (number of processors parameter).

In most of the cases, if you run a consolidation project, you have more processors on your source system than on your Linux guest on IBM LinuxONE. You can then use the flexibility of your IBM LinuxONE infrastructure by allocating more CPU and memory during the import phase.

When the import is over, you can adjust the configuration according to your Linux guests needs.

5.5 Successful production

Migration is now successfully completed and your Oracle databases runs smoothly on your IBM LinuxONE System.

To ensure that your production carries on meeting the SLA, regular monitoring should be used at the following layers of the application stack (for more information, see Chapter 6., “Performance management for IBM z/VM, Linux, and Oracle on LinuxONE”):

- ▶ LinuxONE infrastructure
- ▶ z/VM hypervisor
- ▶ Linux
- ▶ Oracle database
- ▶ Application

Also, a workload analysis highlights growth variation and can help to prepare and anticipate capacity planning.



Performance management for IBM z/VM, Linux, and Oracle on LinuxONE

For the last 10 years, server consolidation, virtualization, and workloads as services in the cloud environment have been the driving factors for the IT infrastructure industry. IBM LinuxONE is a platform that is designed for the consolidation of discrete servers that are running workloads, such as Oracle.

It also is designed for virtualizing physical hardware, such as processing units, memory, and I/O devices, and facilitating Oracle databases as services in the cloud environment. Virtualized environments are created in LinuxONE by using hardware hypervisor and software hypervisor.

The physical machine can be divided into logical partitions (LPARs) and each LPAR can run one to thousands of Linux guests to host Oracle databases. Each Linux guest can also run one or many Oracle databases on them. Applications (such as Oracle) can be deployed to use simultaneous multithreading (SMT) to increase the processing efficiency and throughput. Single-instruction, multiple-data (SIMD) elevates performance enhancements in the analytics area.

These levels of sophistication and flexibility to run Oracle databases on LinuxONE, along with the ability of the systems administrator to overcommit certain resources, such as CPU and memory, which necessitates the need to monitor the LinuxONE vigorously for optimized usage and efficiency. Performance must be monitored on LinuxONE at various levels, such as the hypervisor, hosting operating system, and work load levels.

In this chapter, we describe some of the more widely used tools that are used to monitor Oracle performance on LinuxONE servers.

This chapter includes the following topics:

- ▶ 6.1, “Oracle performance monitoring overview” on page 83
- ▶ 6.2, “Oracle database tools” on page 84
- ▶ 6.3, “Hypervisor monitoring tools” on page 87
- ▶ 6.4, “zVPS” on page 87
- ▶ 6.5, “IBM Wave for z/VM” on page 88
- ▶ 6.6, “OMEGAMON XE on z/VM and Linux” on page 88
- ▶ 6.7, “Linux monitoring tools” on page 90
- ▶ 6.8, “Summary” on page 91

6.1 Oracle performance monitoring overview

One of the reasons for the popularity of Oracle workloads on LinuxONE servers is the portability and transparency of the Oracle database operations across various platforms. Monitoring and tuning methodology at the Oracle database level and Linux OS level remains basic across all platforms.

However, the LinuxONE servers are often used as large hosting servers with multiple partitions in a virtualized shared environment for Oracle workloads, which are migrated from discrete servers with non-shared resources. The shared resources in a virtualized environment for Oracle databases require that the associated hypervisor should also be monitored and tuned. Conducting regular health checks on the system provides utilization and historical performance information that can be used for capacity planning and aid in performance analysis.

The powerful processors, memory, and exceptional I/O capabilities of LinuxONE servers set great expectations for Oracle databases to perform at a higher level. Often, users complain about two common performance-related issues. The first complaint concerns degraded usage experiences in which users must spend more time on the day-to-day processes than usual. The other complaint concerns meeting scalability expectations. Before attempting to resolve any performance-related issues, it is vital to complete the following tasks:

1. Ensure that the Oracle workloads are sized accurately for CPU and memory requirements before they are migrated to LinuxONE servers.
2. Identify the users and functionality that was added to the current databases and any new workloads.
3. Document any changes that were made to the databases and processor allocations, memory changes for SGA or pGA, changes to I/O structures and ASM, and Oracle parameter change.
4. Make note of any environmental changes to any of the computing resources (including network resources).
5. Make note of any database architectural changes, such as standalone, to RAC, or multitenant.
6. Look for any bugs that are related to Oracle releases, and ensure that the relevant patches are applied.
7. Look for any changes in the pattern of Oracle workloads executions. Previous workloads might have been sized to be run at different times, and might now be competing for the same resources or running at the same time.
8. Before starting a tuning exercise, ensure that you have the tools available to monitor and collect data at the following levels:
 - Oracle database
 - Hypervisor
 - Linux OS
9. Ensure the processes to collect monitor data are established and repeatable.

Experience shows that Oracle databases performance issues are often the result of poorly written queries. The application developer's accessibility and willingness to accommodate the changes to the identified queries are mandatory requirements.

It is important to know what tuning tools are available for the LinuxONE server at various levels, what type of information they provide, and when to use them. In the following sections, we describe various sets of tools that are available at the Oracle database, hypervisor, and Linux levels.

6.2 Oracle database tools

Running the databases at optimal performance is a primary responsibility for an Oracle DBA. Achieving this goal requires familiarity with the available tools and the ability to choose the correct tool to use.

6.2.1 Oracle Diagnostics Pack

Oracle Diagnostics Pack is a comprehensive solution to manage the performance of Oracle database environments. This pack includes the following popular features:

- ▶ Automatic Workload Repository (AWR)

The AWR is a repository that collects statistics at predetermined intervals about the workloads within the database. The AWR provides a historical reference for performance changes over time, including establishment of performance baselines.

- ▶ Automatic Database Diagnostic Monitor (ADDM)

This feature is an automated tool that focuses on the Oracle database's most intensive operations. It drills down proactively to determine the root cause of any performance problems.

- ▶ Active Session History (ASH)

The ASH samples session activity every second and stores the information in views. Many DBAs use the v\$active_session_history view to isolate performance problems with individual database sessions.

6.2.2 Oracle Tuning Pack for Oracle Database

The Oracle Tuning Pack for Oracle Database automates the database tuning process and includes the following key features:

- ▶ SQL Tuning Advisor

One or more SQL statements can be entered that provide output in the form of specific advice or recommendations about how to optimize SQL for better performance, along with a rationale for each recommendation and its expected benefit. A recommendation might be to collect the statistics about objects, create indexes, restructure the SQL statements, or create SQL profiles.

An automatic mode facility is available that allows the database to automatically implement recommendations. The most frequent use of this feature often is real-time SQL monitoring. If a production environment experiences a performance issue, this method is the most efficient way for a DBA to determine what SQL statements are running while the problem is occurring.

- ▶ SQL Access Advisor

This feature enables you to optimize data access paths of SQL queries by recommending the proper set of materialized views and view logs, indexes, and partitions for a specific SQL workload. It gives advice about how to optimize schema design to maximize query performance.

This feature receives input from various sources, including AWR, to analyze a workload. It also provides recommendations about index creation and deletion, partition creation, and materialized views creation.

6.2.3 Oracle Enterprise Manager Cloud Control

Oracle Enterprise Manager Cloud Control is a popular tool for Oracle customers to manage databases and for provisioning. By using this tool, users can create or upgrade Oracle databases on Linux guests, and configure the network.

Cloud Control also provides an interface for performance advisors and for database utilities, such as SQL*Loader and Recovery Manager (RMAN). To manage Oracle databases, Database Control or Cloud Control can be used. These tools are intuitive, web-based tools from Oracle.

Database Control is included with every database installation and can be used to monitor a single database instance or a clustered database. Cloud Control can monitor many databases that are running on multiple environments from a single console. In addition to database monitoring, other applications, such as Siebel, PeopleSoft, and certain non-Oracle products (for example, IBM WebSphere Application Server) can also be monitored from a single Cloud Control console. Although the Cloud Control Server does not run on LinuxONE servers, the agents run on the Linux guests and communicate with the Cloud Control.

6.2.4 STATSPACK

If the users are not licensed to use Oracle Diagnostics Pack and Database Tuning Pack, they can use the STATSPACK utility, with less functionality. Data is stored in tables that are owned by PERFRSTAT.

6.2.5 Oracle Real Application Testing

In this section, we describe Oracle Real Application Testing, which includes the following features:

- ▶ Database Replay

This feature enables you to capture the database workload on a production system, and replay it on a test system with the same timing and concurrency as the production system on the same or later release of Oracle database.

- ▶ SQL Performance Analyzer

This feature enables you to assess the effect of system changes on SQL performance by identifying SQL statements that regressed, improved, or remained unchanged.

6.2.6 Oracle Orion tool

The Oracle Orion tool is used to predict the I/O performance of an Oracle database. The tool is part of Oracle database installation and simulates Oracle database I/O workloads by using the same I/O software stack as Oracle.

Orion can also simulate the effect of striping that is performed by Oracle Automatic Storage Management. It can run tests by using different I/O loads to measure performance metrics, such as MBPS, IOPS, and I/O latency.

If a database was created on a storage device, the storage alternatively can be calibrated by using the PL/SQL routine `dbms_resource_manager.calibrate_io()`.

6.2.7 Downloadable tools from Oracle

The following downloadable tools are provided by Oracle in their support website. These tools also help to monitor and tune Oracle databases that are running on LinuxONE:

- ▶ **SQLTExplain (SQLT)**

By using this powerful tool, users can diagnose SQL statements that are performing poorly. SQLT also helps to expedite SQL tuning process.

For more information, see MOS Note 215187.1.

- ▶ **Light Onboard Monitor (LTOM)**

LTOM provides real-time automatic problem detection and data collection. It proactively collects the necessary diagnostic traces in real time while the database or system performance problem is occurring.

For more information, see MOS Note 352363.1.

- ▶ **Remote Diagnostic Agent (RDA)**

The data that is captured provides Oracle Support with a comprehensive picture of the customer's environment, which aids in problem diagnosis.

For more information, see MOS Note 414966.1.

- ▶ **OSWatcher**

This tool gathers system profile data, including `vmstat`, `top`, and `iostat`, and works similar to a "sar".

For more information, see MOS Note 461053.1.

One of the deciding factors when considering the use of any of these tools is to ensure that the tool can collect usage metrics of the database. The tool also should identify specific database problems and solutions as they occur.

6.3 Hypervisor monitoring tools

In a LinuxONE server running Oracle databases, the certified hypervisor is IBM z/VM. Typically, a LinuxONE server is divided into LPARs, and each LPAR can run one or thousands of Linux guests to host Oracle databases. Each Linux guest can also run one or many Oracle databases on them.

Oracle databases and the Linux system run multiple processes, and the operating system allocates resources to each of these processes. These multiple layers of virtualization can make it difficult for an operating system to find the best way to use the allocated resources.

Also, the Linux operating system was not originally designed to run in a shared environment, which makes the tuning environment complicated.

6.3.1 IBM Performance Toolkit for VM feature

The Performance Toolkit for VM provides enhanced capabilities for a z/VM systems programmer, system operator, or performance analyst to monitor and report performance data. Offered as a priced optional feature of z/VM, the Performance Toolkit for VM is derived from the FCON/ESA program (5788-LGA), which was not available in all countries.

The Performance Toolkit for VM provides the following functions:

- ▶ Operation of the system operator console in full-screen mode.
- ▶ Management of multiple z/VM systems (local or remote).
- ▶ Post-processing of Performance Toolkit for VM history files and VM monitor data that is captured by the MONWRITE utility.
- ▶ Viewing of performance monitor data that uses web browsers or PC-based 3270 emulator graphics.
- ▶ TCP/IP performance reporting.

For more information about Performance Toolkit for z/VM and performance on Linux, see the following resources:

- ▶ [Performance Toolkit for VM Feature](#)
- ▶ [Linux on IBM System z: Performance Measurement and Tuning, SG24-6926](#)

6.4 zVPS

zVPS, the Velocity Performance Suite from Velocity Software, provides the function for full performance management. The requirements for continuous performance management include having low overhead, a full performance database, and complete metrics. For data collection from z/VM, zVPS uses the CP Monitor. For data collection for Linux, network, and Oracle, zVPS is “agentless” and uses the snmp daemon for data collection.

For more information about the zVPS tool, see the following resources:

- ▶ [Experiences with Oracle Database 12c Release 1 on Linux on System z, SG24-8159](#)
- ▶ [Velocity Software](#)

6.5 IBM Wave for z/VM

In certain cases where the users are new to z/VM hypervisor environment, the IBM Wave for z/VM tool can be used to improve the productivity for the day-to-day operations of the z/VM systems. IBM Wave for z/VM simplifies virtualization management of z/VM and virtual Linux servers on IBM LinuxONE. It integrates seamlessly with z/VM and Linux environments to make VM auditing, storage, and reporting processes simple and cost-effective. The IBM Wave for z/VM tool includes the following features:

- ▶ Manage virtual Linux servers and processes from a single user interface.
- ▶ Support advanced z/VM capabilities by using only a few keystrokes.
- ▶ Simplify and automate common virtualization and administration tasks.
- ▶ Gain flexible reporting.
- ▶ Rapidly provision virtual Linux servers and z/VM resources.
- ▶ Reduce the learning curve to manage z/VM and virtual Linux servers.
- ▶ Easily identify, make available, and aggregate resources, which allows for fast deployments to a private cloud infrastructure.

For more information about IBM Wave for z/VM, see the [IBM Wave for z/VM website](#).

6.6 OMEGAMON XE on z/VM and Linux

Bottleneck analysis and problem determination are facilitated by using sophisticated tools, such as IBM Tivoli® OMEGAMON® on z/VM and Linux. OMEGAMON detects performance problems and alerts before degraded response time becomes evident.

Information about your Linux instances that are running as z/VM guests and the Linux workloads reveals how they are performing and affecting z/VM and each other. OMEGAMON can perform the following functions:

- ▶ Compare Linux operations side by side with detailed performance metrics from other important systems.
- ▶ Data collection from the Performance Toolkit for VM (PTK is a prerequisite) complements data collection by the IBM Tivoli Monitoring for Linux on System z agent.
- ▶ View and monitor workloads for virtual machines, groups, response times, and LPAR reporting and view reports on z/VM and Linux usage of resources such as CPU utilization, storage, mini-disks, and TCP/IP.
- ▶ High-level views help executives understand how systems performance influences business and the bottom line.
- ▶ With granular views, IT staff can more easily track complex problems that span multiple systems and platforms and share related information.

OMEGAMON includes a powerful agent to monitor the Oracle databases that are running on LinuxONE servers. With its rich charting capabilities, it can provide the accurate information about how Oracle databases are performing by remote monitoring of the Linux servers with low overhead.

By monitoring Oracle system tables, it provides a wealth of information in a graphical way about the following key resources:

- ▶ Table spaces
- ▶ Cache Hit %
- ▶ Contention
- ▶ Archive Logs
- ▶ Rollback Segments
- ▶ Databases
- ▶ Instances
- ▶ Top SQL Statements
- ▶ Oracle Monitoring
- ▶ SGA
- ▶ pGA
- ▶ RAC
- ▶ ASM

Figure 6-1 shows an example of one of the OMEGAMON XE on z/VM and Linux windows.

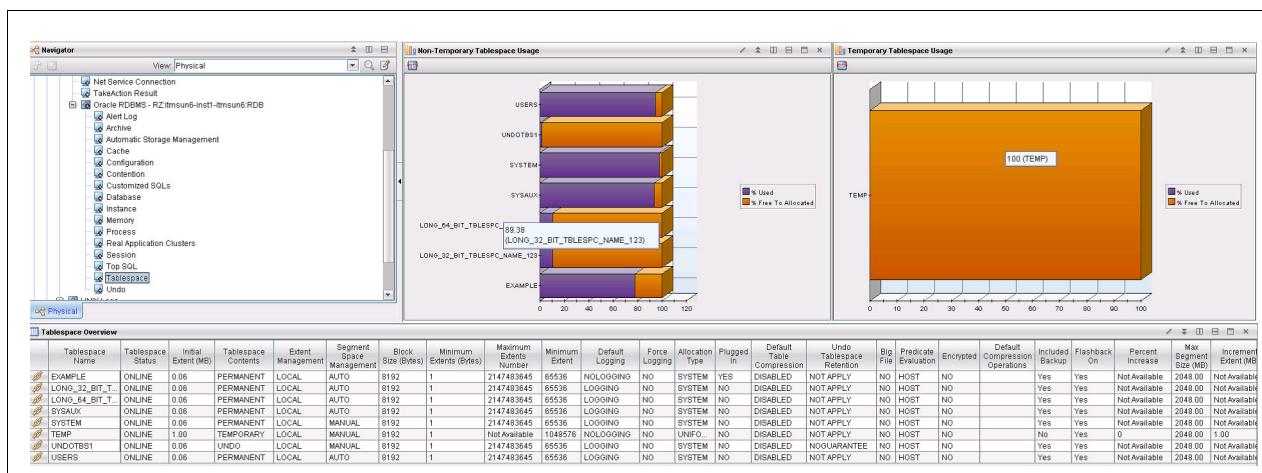


Figure 6-1 Example window of OMEGAMON XE on z/VM and Linux

The OMEGAMON agents are RAC aware. As an end-to-end solution for monitoring the LinuxONE servers, it provides the following benefits:

- ▶ Provides visibility into OS and Oracle metrics in-context. In Oracle environments, disk latency and other OS metrics, such as swap, memory, and CPU availability frequently affect Oracle.
- ▶ Easily navigates from the Oracle metrics to the underlying OS or hypervisor.
- ▶ Provides consistent metrics across personas (Dev, IT Ops, Lines of Business, Oracle SME, and so on).
- ▶ Has access to real-time and historical data.
- ▶ Use Analytics.

Figure 6-2 shows a sample window which displays the memory that is used by an Oracle relational database management system (RDBMS).

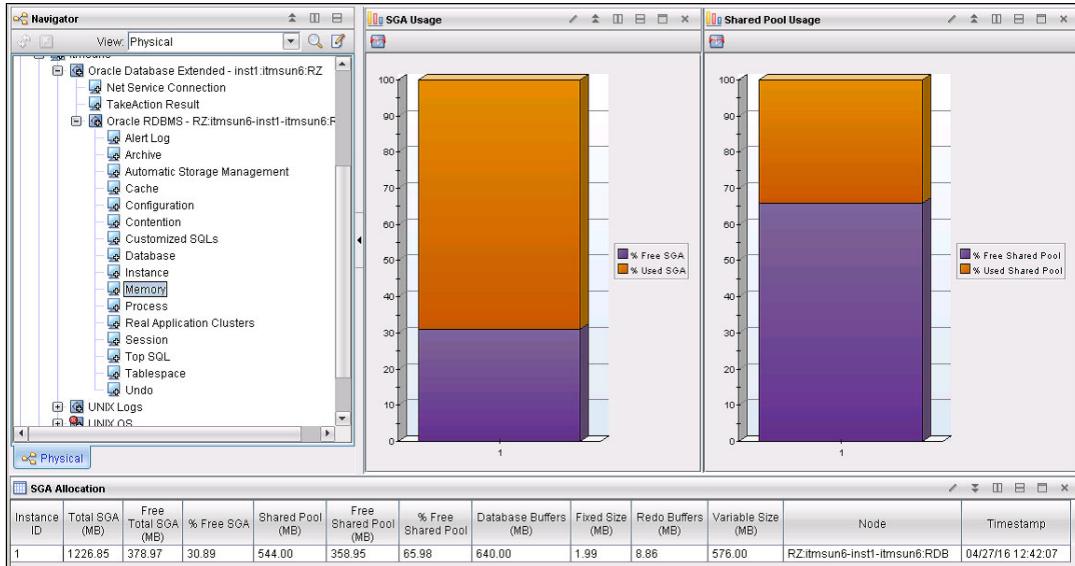


Figure 6-2 Memory used by an Oracle RDBMS

6.7 Linux monitoring tools

In a LinuxONE server that is running Oracle databases, many tools are available from the Linux community.

The following standard tools are available for monitoring Linux guests that are running on LinuxONE server (this list is ever growing):

- ▶ **Pidstat:** Per process statistics
- ▶ **Slabtop:** Kernel memory pool consumption
- ▶ **Lsof:** Check file flags of open files
- ▶ **Blktrace:** Low-level disk I/O analysis
- ▶ **Hypotop:** Cross guest CPU consumption monitor
- ▶ **Iptraf:** Network traffic monitor
- ▶ **Dstat:** Configurable live system overview
- ▶ **Irqstats:** Check IRQ amount and CPU distribution
- ▶ **Smem:** Per process/per mapping memory overview
- ▶ **Jinsight:** Java method call stack analysis
- ▶ **Htop:** Top on steroids
- ▶ **Strace:** System call statistics
- ▶ **Ltrace:** Library call statistics
- ▶ **Kernel tracepoints:** Get in-depth timing inside the kernel
- ▶ **Vmstat:** Virtual memory statistics
- ▶ **Sysstat:** Full system overview
- ▶ **Iostat:** I/O-related statistics
- ▶ **Dasdstat:** Disk statistics
- ▶ **scsi statistics:** Disk statistics
- ▶ **Perf:** Hardware counters, tracepoint-based evaluations, profiling to find hotspots
- ▶ **Valgrind:** In-depth memory/cache analysis and leak detection
- ▶ **Java Health Center:** High-level java overview and monitoring
- ▶ **Java Garbage Collection and Memory visualizer:** In-depth GC analysis

- ▶ Netstat: Network statistics and overview
- ▶ Socket Statistics: Extended socket statistics
- ▶ top/ps: Process overview
- ▶ Icastats/lscrypt: Check usage of crypto-hardware support
- ▶ Lsluns/multipath: Check multipath setup
- ▶ Lsqeth: Check hardware, checksumming and buffer count
- ▶ Ethtool: Check offloading functions
- ▶ Collectl: Full system monitoring
- ▶ Ftrace: Kernel function tracing
- ▶ Lttng: Complex latency tracing infrastructure
- ▶ Ziomon: Analyze FCP setup and I/O
- ▶ Systemtap: Another kernel tracing infrastructure
- ▶ Wireshark/Tcpdump: Analyze network traffic in depth
- ▶ Iotop: Order processes by disk I/O
- ▶ Iftop: Per connection traffic overview

For more information about these tools, see the [IBM developerWorks Resource Management and Analysis website](#).

For more information about how to use other tools and commands, see the following resources:

- ▶ [*Linux on IBM System z: Performance Measurement and Tuning*, SG24-6926](#)
- ▶ [Linux system monitoring](#)

6.8 Summary

Oracle databases are one of the popular workloads on LinuxONE server. Steps should be taken proactively to optimize the performance of Oracle databases on LinuxONE by using the appropriate tools described in this chapter.



High Availability

The IBM LinuxONE is a perfect choice for installations where High Availability (HA) is key customer criteria in their hardware infrastructure selection for Oracle database deployments. In this chapter, we show how the highly regarded HA characteristics of IBM LinuxONE are an ideal complement for installations that are running the Oracle Database and use Oracle's specific HA features.

In this chapter, we describe many of the Oracle Database options that are available to customers as a standard feature or a chargeable License Fee Based option. For more information about all the Oracle options, see Oracle's documentation or online resources.

We also describe the LinuxONE features that play an important role for clients who are looking for a hardware server with a robust set of characteristics in this domain.

Finally, we provide examples of many scenarios where the Oracle Database that is running Linux on LinuxONE can provide an excellent HA enterprise-ready environment solution.

This chapter includes the following topics:

- ▶ 7.1, “Oracle HA options”
- ▶ 7.2, “HA building blocks for Oracle Database on Linux running on LinuxONE” on page 101
- ▶ 7.3, “IBM LinuxONE with Oracle Database: Considerations and examples for HA” on page 103
- ▶ 7.4, “Summary and recommendations” on page 109

7.1 Oracle HA options

The topic of HA is important for installations that are running databases that process critical applications for the organizations that they serve. This subject affects almost every data center or Cloud environment in the IT Industry. HA is top of mind for those customers who choose Oracle Database as their software in this area and have (or are considering) LinuxONE as their infrastructure hardware.

The available Oracle HA options are listed in Table 7-1.

Table 7-1 Oracle HA options

HA/DR feature	Benefit	Cost	Level of availability
Real Application Clusters One Node	Automated fail over (active/passive) for planned and unplanned outages	Oracle optional licensed feature	High Availability with minimal disruption to database (seconds)
Real Application Clusters	Continuous availability of cluster	Oracle optional licensed feature	High Availability with no disruption to database
Application Continuity	Protects applications from database session failures by replaying in-flight transactions	Licensed (included with RAC, RAC One Node, and Active Data Guard)	Disaster Recovery
Edition-Based Redefinition (EBR)	Enables online applications upgrades with uninterrupted availability of the applications	Included with Enterprise Edition	High Availability for maintenance
Oracle Flashback	Flashback provides efficient and quick error correction (logical and table rows)	Included with Enterprise Edition	High Availability for error correction
Online Reorganization and Redefinition	Provides flexibility to modify a table's physical attributes, while allowing users full access to the database	Included with Enterprise Edition	High Availability for database maintenance
Data Guard/Active Data Guard	Disaster-recovery solution that provides fast automatic failover, in the case of database or node failures.	Data Guard included with Enterprise Edition. Active Data Guard requires a license	Disaster Recovery
GoldenGate	Provides real-time data replication for heterogeneous environments	Licensed product	Disaster Recovery and Migration/Replication Services

HA/DR feature	Benefit	Cost	Level of availability
Global Data Services (GDS)	GDS provides region-based workload routing, and replication lag based workload routing for Active Data Guard	Included with GoldenGate or Active Data Guard	Disaster Recovery
Storage Base Replication Solutions	Storage replication uses disk consistency groups for databases and applications, to provide a simplified DR approach.	Replication software cost (non-Oracle products)	Disaster Recovery
Oracle Site Guard	Oracle Site Guard enables administrators to automate complete site switchover or failover.	Included with Data Guard	Disaster Recovery
Recovery Manager (RMAN)	RMAN provides a foundation for backing up and recovering the Oracle database.	Included with Oracle Database Enterprise Edition.	High Availability for back ups
Secure Backup (OSB)	OSB delivers centralized disk or backup	Included with Oracle Database Enterprise Edition	High Availability for back ups
Oracle Multitenant	Consists of a container database (CDB) for the metadata and the pluggable databases (PDB) which contain the application database (can be unplugged and plugged into different CDB databases)	Optional licensed feature of the Database	High Availability

Table 7-1 on page 94 is meant to be a general overview of possible Oracle Database HA features that might be used with Linux that is running on IBM LinuxONE.

Oracle Maximum Availability Architecture (MAA) is a set of best practices that are developed by Oracle that help with a HA architecture. It uses the Oracle HA and disaster technologies. MAA is designed to provide data protection and availability by minimizing or eliminating planned and unplanned downtime at the technology layers, including hardware and software components. It delivers protection and HA for many types of failure, whether from hardware failures that cause data corruption, or anything ranging from human error to software malfunctions. It also includes HA features to help an installation when nature-related disasters affect a broad geographic area.

Oracle provides a comprehensive and integrated set of HA technologies that enable rapid recovery from failures and minimize planned downtimes. The focus of this section is on HA options that are related to the Oracle Database.

7.1.1 Oracle Real Application Clusters

Oracle Real Application Clusters (RAC) is a cluster database with a shared cache architecture that provides highly scalable and available database solutions for all business applications.

It was introduced on Oracle Database Version 9 (9i) several years ago. It is one of the key options for HA that installations can choose when running Oracle Database. RAC is a chargeable License Fee Based option of the Oracle Database Enterprise Edition.

Oracle RAC on Linux running on LinuxONE provides multiple Linux guest operating systems for a database application to help limit application downtime. Every quarter, Oracle provides patch set updates with important security and database fixes. Oracle RAC/RAC One Node application downtime can be minimized by patching one system at a time (rolling), while the other Linux guest systems continue to run applications.

7.1.2 Oracle RAC One Node

Oracle RAC One Node is a single instance of an Oracle RAC-enabled database that is running on one node in a cluster. It is an HA solution that runs a database on one node of a cluster at a lower price point than full Oracle RAC.

RAC One Node provides automatic failover capability to other nodes in the cluster. Typically, planned maintenance operations can be completed with minimal disruption to application users by relocating the application Virtual IP addresses (VIPs) while the instances are active and running.

Technologies, such as transparent application failover (TAF), can be used to seamlessly move an active running SQL statement from one Oracle RAC ONE node to another node without disruption to the user.

One of the benefits of Oracle RAC One Node is that applications that might not be RAC friendly are not as affected as a full multi-node RAC Cluster solution. For example, an insert intensive application that uses many non-cached sequences might perform better running on only one node versus being distributed across multiple nodes in a RAC cluster.

Another benefit of Oracle RAC One Node is that a RAC One Node mode can be converted to full RAC (and vice versa) by using the `srvctl convert` command.

7.1.3 Oracle Clusterware

Oracle Clusterware provides the foundation for running Oracle RAC/RAC One Node.

A cluster consists of multiple interconnected computers or servers that appear as though they are one server to users and applications. Oracle RAC uses Oracle Clusterware for the infrastructure to bind multiple servers so that they operate as a single system.

In the addition to being the integrated foundation for Oracle RAC, Oracle Clusterware 12c Release 1 can be used to deliver HA for other applications. Also, Oracle Clusterware depends on interconnect technology for rapid dissemination of information between different Oracle instances.

7.1.4 Edition-based redefinition

Edition-based redefinition (EBR) is an Oracle Database feature that was originally introduced with Oracle Database 11gR2. It enables online applications upgrades with uninterrupted availability of the applications. When the installation of an upgrade is complete, the pre-upgrade application and the post-upgrade application can be used simultaneously. Therefore, a session can continue to use the pre-upgrade application until its user decides to end it and all new sessions can use the post-upgrade application. When there are no longer any sessions using the pre-upgrade application, it can be closed. EBR enables hot rollover from a pre-upgrade version to a post-upgrade version, with no downtime.

7.1.5 Online Reorganization and Redefinition

The Online Reorganization and Redefinition feature that is available in Oracle Database 12c provides the flexibility to modify a table's physical attributes and transform data and table structure, while allowing users full access to the database. This capability improves data availability, query performance, response time, and disk space utilization.

During an application upgrade, administrators can start the redefinition process and at intervals synchronize the interim table so that it includes the latest changes to the original table. The advantage is that the amount of time to complete the final redefinition step is reduced. Also, administrators can validate and use the data in the interim table before completing the redefinition process.

7.1.6 Oracle Flashback

Oracle Database Flashback Technologies are a set of data recovery solutions that allow reversing mistakes by selectively undoing the effects of a previous error. Flashback provides efficient and quick error correction. Flashback supports recovery at all levels, including the row, transaction, table, and entire database. It also provides a growing set of features to view and rewind data back and forth in time.

Oracle Flashback Database is related to data protection features that enable a user to rewind data back in time to correct any problems that are caused by logical data corruption or user errors within a designated time window. Oracle Flashback provides an efficient alternative to point-in-time recovery and does not require a backup of the database to be restored first.

Flashback Database and restore points are effective in traditional database recovery situations and can also be useful during database upgrades, application deployments, and testing scenarios when test databases must be quickly created and re-created. Flashback Database also provides an efficient alternative to rebuilding a failed primary database after a Data Guard failover.

Note: Oracle Flashback does not replace a regular database backup and recovery procedure.

7.1.7 Oracle Data Guard/Active Data Guard

Data Guard is an Oracle database offering that provides the management, monitoring, and automation software to create and maintain one or more synchronized copies of a production database. It also provides HA for mission critical applications.

It is a HA and disaster-recovery solution that provides fast automatic failover, in the case of database failures, node failures, corruption, and media failures. The standby databases can be used for read-only access, reporting purposes, and testing and development purposes. Data Guard is included with Oracle Database Enterprise Edition.

Active Data Guard is an optional license component for Oracle Database Enterprise Edition. Active Data Guard adds advanced capabilities to extend basic Data Guard functionality by allowing for databases to be opened at the disaster recovery (DR) site for read-only access use by applications.

Oracle Far Sync allows for zero data loss at any distance by first copying transaction logs initially to another DR site that is geographically close by using synchronous replication of the database to the nearby site. Then, this nearby site's data is replicated asynchronously to another standby DR site at a great distance.

7.1.8 Application Continuity

Application Continuity (AC) protects applications from database session failures because of instance, server, storage, network, or any other related component, and even complete database failure. Application Continuity replays affected "in-flight" requests so that the failure appears to the application as a slightly delayed execution, masking the outage to the user.

AC is a feature that is available with the Oracle RAC, Oracle RAC One Node, and Oracle Active Data Guard.

If an entire Oracle RAC cluster fails, which makes the database unavailable, Application Continuity replays the session, including the transaction, following an Oracle Active Data Guard failover. Use of Application Continuity with a standby database requires Data Guard Maximum Availability mode (zero data loss) and Data Guard Fast Start Failover (automatic database failover).

7.1.9 Oracle GoldenGate

Oracle GoldenGate is a software package for real-time data integration and replication in heterogeneous IT environments. It also enables HA solutions, real-time data integration, transactional change data capture, data replication, transformations, and verification between operational and analytical enterprise systems.

Oracle GoldenGate 12c is the newest version of this Oracle product. While maintaining excellent performance with simplified configuration and management, it features a tighter integration with Oracle Database, support for cloud environments, expanded heterogeneity, and enhanced security.

7.1.10 Global Data Services

Oracle GoldenGate and Oracle Active Data Guard allow for distribution of application workloads with replicated databases. When applications are spread across multiple databases in multiple data centers, it can sometimes be challenging to efficiently optimize databases with the best performance based on server loads and network latencies.

Oracle Global Data Services (GDS) provides the following key capabilities for a set of replicated databases that are globally distributed or located within the same data center:

- ▶ Region-based workload routing
- ▶ Connect-time Load balancing
- ▶ Runtime load balancing advisory for Oracle-integrated clients
- ▶ Inter-database Service failover
- ▶ Replication lag based workload routing for Active Data Guard
- ▶ Role-based global Services for Active Data Guard
- ▶ Centralized workload management framework

Oracle GDS/Global Service Manager (GSM) is available for Linux running on LinuxONE, starting with Oracle 12c R1. Oracle GDS is included with an Active Data Guard or a Golden Gate replication license.

7.1.11 Storage-based replication

Oracle databases typically use a principle of dependent-write I/O through a series of synchronized writes to the data files and logs. No dependent-write is issued until the predecessor write completes. This feature allows those systems to be restorable if there is a power failure.

For an Oracle consistent backup/DR solution in “hot” or online mode, the database must be synchronized with the online database logs. This configuration is typically done by using the **alter database begin** and **alter database end backup** commands. When these Oracle backup commands are issued, data blocks for the table spaces are flushed to disk and the data file headers are updated with the last checkpoint System Change Number (SCN).

Updates of the checkpoint SCN to the data file headers are not performed while in this backup mode. Oracle does not update these data file headers while in backup mode. When these files are copied, the non-updated SCN signifies to the database that recovery is required.

For a storage-based “consistent” Oracle replication strategy, co-ordination between the storage-based disk backup solution and the Oracle database is needed.

Oracle Data Guard is the only certified and supported way (by Oracle) to protect Oracle database files from Database block corruption. If an Oracle data block is corrupted on disk, a storage-based replication can replicate the corrupted blocks to the DR site.

Disk-based storage replication is supported for replicating Linux OS file systems, Oracle binaries, and any non-database file systems that are required to be kept in sync with the database.

7.1.12 Oracle Site Guard

Oracle Site Guard is a disaster-recovery solution that enables administrators to automate complete site switchover or failover. It orchestrates the coordinated failover of Oracle Data Guard with Application Servers, which can be integrated with a storage-based replication technique.

Oracle Site Guard integrates with underlying replication mechanisms that synchronize primary and standby environments and protect mission critical data. It includes built-in support for Oracle Data Guard for Oracle database. Oracle Site Guard supports typical storage replication solutions that are used with LinuxONE but requires some scripting to integrate with a SAN storage-based replication solution.

7.1.13 Oracle Recovery Manager

Oracle Recovery Manager (RMAN) provides a comprehensive foundation for efficiently backing up and recovering the Oracle database. Providing block-level corruption detection during backup and restore, RMAN optimizes performance and space consumption during backup with file multiplexing and backup set compression. It works with Oracle Secure Backup and third-party media management products for tape backup.

RMAN handles all underlying database procedures before and after backup or restore, which frees dependency on OS and SQL*Plus scripts.

7.1.14 Oracle Secure Backup

Oracle Secure Backup (OSB) delivers centralized disk and backup management for the entire IT environment. It consists of the following offerings, both of which are integrated with Oracle RMAN:

- ▶ Centralized backup management and high-performance and heterogeneous data protection in distributed UNIX, Linux, Windows, and network-attached storage (NAS) environments.
- ▶ An OSB load Module that provides integrated Oracle Database backup to Amazon S3 cloud (Internet) storage.

7.1.15 Oracle Multitenant

Oracle Database 12c offers a new chargeable License Fee Based option that is named Oracle Multitenant. This option provides simplified consolidation that requires no changes to your applications. In this new architecture, a multitenant container database can hold many pluggable databases. An administrator manages the multitenant container database, but application code connects to one pluggable database.

Oracle Multitenant consists of a container database (CDB) which features all of the metadata for the Oracle system. No application data schemas are configured in the CDB.

The pluggable database (PDB) contains the application-specific database, and can be unplugged and plugged into different CDB databases.

Oracle Multitenant allows the management of many databases as one and can provide several benefits from an administration perspective. For example, in a multiple PDB configuration, many pluggable databases can be backed up with one Oracle RMAN configuration, which reduces the time that is necessary to back up many databases if they were independent.

Oracle Multitenant in a HA scenario allows for the ability to unplug and move a pluggable database from one container database to another. This feature is useful in a HA environment where a pluggable database is moved to another CDB running on another guest.

In the case of an Oracle upgrade, the database upgrade downtime is reduced as the schema updates are mostly made to the system table spaces, which are a component of the CDB and not to the metadata components of the PDB. For example, unplugging a PDB from a 12.1.0.1 CDB into a 12.1.0.2 CDB reduces downtime for an application instead of directly upgrading a database from 12.1.0.1 to 12.1.0.2.

7.2 HA building blocks for Oracle Database on Linux running on LinuxONE

This section describes how Oracle HA options interact with Linux running on LinuxONE. The many Oracle HA options are an ideal complement for the superb HA features that are provided by LinuxONE.

Figure 7-1 shows a complete HA scenario that consists of the following major components:

- ▶ The Oracle Database and its HA options
- ▶ The Linux operating system
- ▶ LinuxONE

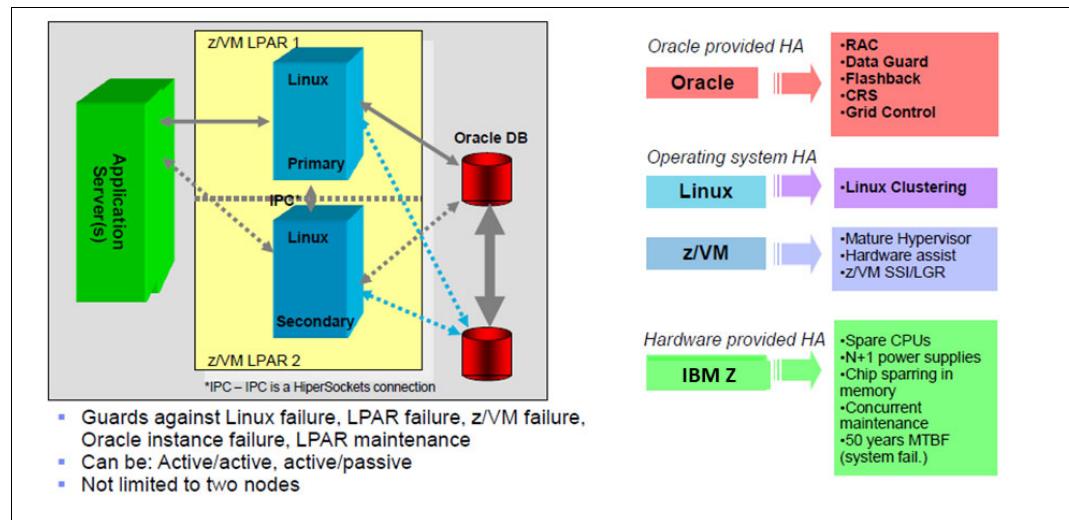


Figure 7-1 Building blocks of an HA environment for Oracle Database on Linux running on LinuxONE

7.2.1 Hardware provided HA

IBM LinuxONE has high reliability through various redundant hardware options. IBM LinuxONE has transparent CPU sparing; that is, if a CPU encounters a hardware failure, a spare CPU can be quickly brought online without affecting the applications or database.

IBM LinuxONE has Redundant Array of Independent Memory (RAIM) memory modules. RAIM memory prevents a memory module from affecting the availability of a system or application.

In addition, IBM LinuxONE includes numerous other RAS features. The following hardware-provided HA features are a standard offering:

- ▶ Concurrent maintenance: At a minimum of two processor books, hardware parts of the processor can be replaced without a Power on Reset with the normal continuation of operations.
- ▶ Chip Sparring in Memory: An error detection and correction mechanism on IBM LinuxONE that allows error detection during instruction execution and transparent error correction of spare processors, which were configured to the system.
- ▶ N+1 Processor supplies: The ability to have two sets of redundant power supplies. Each set of the power supplies has its individual power cords or pair of power cords, depending on the number of Bulk Power Regulator (BPR) pairs installed.

7.2.2 Operating system HA

Several operating systems can run on an IBM LinuxONE. In this section, we describe those systems that are relevant for an environment in which Oracle Databases are deployed.

Linux clustering

Red Hat Enterprise Linux Servers and SUSE Enterprise Linux Server are the only two Linux distributions that are certified by Oracle to run its database on IBM LinuxONE. Oracle Clusterware works by moving application VIPs between Linux guest nodes. Several Linux operating system-based solutions can also be used to provide HA for applications.

SUSE's Linux Enterprise High Availability Extension, Sine Nomine's High Availability Option (HAO) for Red Hat, Red Hat High Availability and Resilient® Storage Add Ons (as of RHEL 7 Update 2), and IBM Tivoli System Automation all can be configured to provide HA for business-critical applications.

These Linux based HA solutions work in a similar manner, whereby Oracle Database is configured to run on a Linux guest. If planned maintenance or an unplanned system event occurs, the HA solution fails over to another idle Linux guest.

The Linux HA solution shuts down the source system database and unmounts the Oracle file systems on the source system. Next, the Linux HA solution mounts the necessary Oracle file systems, starts the application VIPs on the server, and then, starts the Oracle listener on this VIP IP address on the failover server. The last step is to start the database on the failover Linux guest such that the applications can now connect to the failover server.

Depending on activity load in the database and other factors, a failover can be as quick as a few seconds to the new server, or up to several minutes if many insert or update operations require rolling back.

Hypervisor (z/VM)

IBM z/VM offers a base to use IBM virtualization technology on IBM LinuxONE. Fully tested and deployed in Oracle development environments, Oracle products are certified with z/VM.

IBM z/VM provides a highly secure and scalable enterprise cloud infrastructure. It also provides an environment for efficiently running multiple diverse critical applications on IBM LinuxONE with support for more virtual servers than any other platform in a single footprint.

IBM z/VM virtualization technology is designed to allow for the capability to run hundreds to thousands of Linux servers on a single LinuxONE footprint.

IBM Wave for z/VM helps dramatically simplify the administration and management of z/VM and virtual Linux servers that are running on IBM LinuxONE. IBM Wave integrates seamlessly with z/VM and enterprise Linux environments to help administrators view, organize, and manage resources in an optimized and standardized manner.

7.2.3 Oracle provided HA

Oracle Clusterware provides HA for an application by transferring the applications connectivity information with VIP. When a node goes down (planned or unplanned), the application connectivity information or VIP is moved to another node in the cluster to accept the application's connections.

When the VIP is moved from one Linux host to another, Oracle broadcasts the new IP to the network and the newly associated network media access control or MAC address with an ARP command.

Applications can seamlessly route database connections to the new server without incurring downtime by connecting to the highly available VIP, as opposed to a hardcoded Linux server IP address.

Oracle Clusterware interfaces directly with the hardware technology that is available on LinuxONE.

7.3 IBM LinuxONE with Oracle Database: Considerations and examples for HA

In this section, we provide some considerations and examples of HA.

7.3.1 Networking

IBM LinuxONE has several options for designing the network between Logical Partitions (LPARs) and other servers on a network.

An Open System Adapter (OSA) is a physical network card in LinuxONE. An OSA can be dedicated to an Oracle Linux guest, which is shared among multiple Linux guests, or configured in a virtual switch (VSwitch) configuration. VSwitches are beneficial for systems with many Linux guests sharing network infrastructure.

A HiperSocket is a high-speed, low-latency memory to memory network that traverses internally within LinuxONE only. IBM HiperSockets™ are advantageous for workloads in which a large amount of data must be transferred quickly between LPARs provided there is enough CPU capacity to support the network transfers.

Figure 7-2 shows some of these network concepts to consider in a network design for development, test, and production environments.

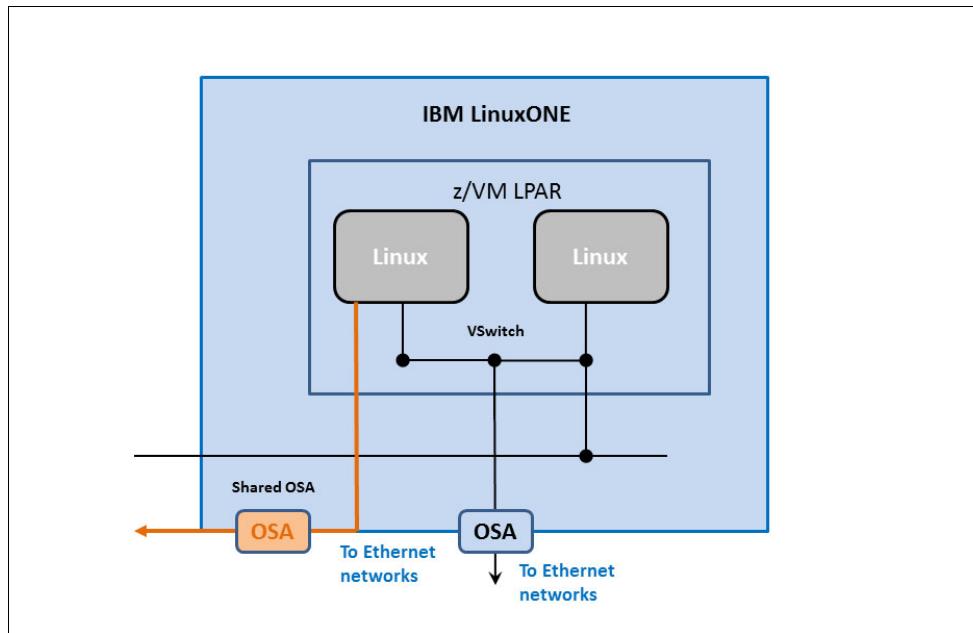


Figure 7-2 IBM LinuxONE servers network considerations

7.3.2 Oracle HA networking options

IBM LinuxONE includes the following options for supporting the Oracle Interconnect® between Oracle nodes in an Oracle RAC or Oracle RAC OneNode cluster configuration:

- ▶ Link Aggregation: (active/active) Allows up to eight OSA-Express adapters to be aggregated using a VSwitch type of configuration.
- ▶ Linux Bonding: Whereby Linux provides the HA by using two or more Linux interfaces in an active/backup or active/active configuration; for example, Linux interfaces “eth1 and eth2” are configured to create a highly available bonded interface that is named “bond0”.
- ▶ Oracle HA IP (HAIP): With Oracle 11gR2+, you can now have up to four Private interconnect interfaces in which Oracle’s HAIP functionality is used to balance network load across the Oracle RAC/RACONE interconnect network interfaces.

7.3.3 Oracle RAC Interconnect z/VM Link Aggregation

Figure 7-3 shows a typical IBM z Server topology, with multiple Oracle RAC clusters, that uses multiple VSwitches with Link Aggregation.

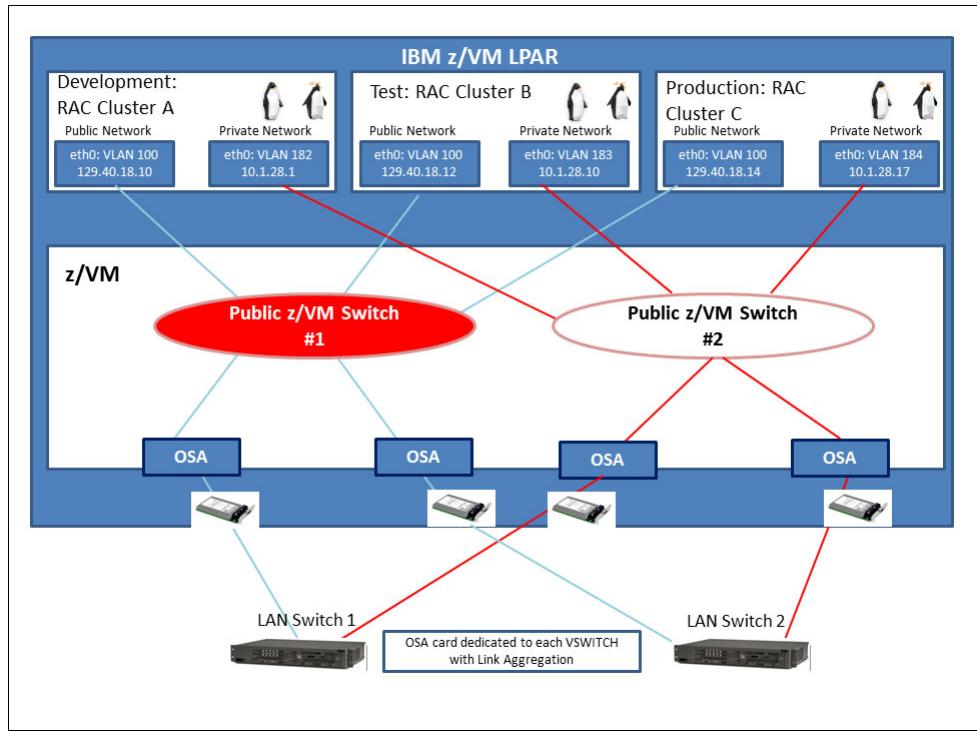


Figure 7-3 Oracle Interconnect with z/VM Link Aggregation

7.3.4 Oracle RAC interconnect considerations

The design of the Oracle RAC/OneNode interconnect is vital to the availability of an Oracle cluster. Oracle requires that the Cluster interconnect is configured on a private, dedicated LAN or VLAN (tagged or untagged), which is non-routable and isolated from other non-interconnect traffic.

The benefit of a VSwitch approach is that multiple RAC clusters can share the network infrastructure. It is strongly recommended to have a minimum of two VSwitches for an Oracle RAC configuration (public and private).

IBM LinuxONE environments that run in two separate LPARs can use multiple HiperSocket interfaces, which can be load balanced with Oracle's HAIP load balancing capability.

For environments that are CPU bound or cluster interconnect sensitive, the dedicated OSA cards approach (outside of a VSwitch) works well to ensure that the cluster interconnect continues to operate efficiently during periods of high CPU load.

7.3.5 I/O channel failover considerations

IBM LinuxONE supports multiple logical channel subsystem (LCSS) images that are mapped onto a physical channel subsystem (see Figure 7-4).

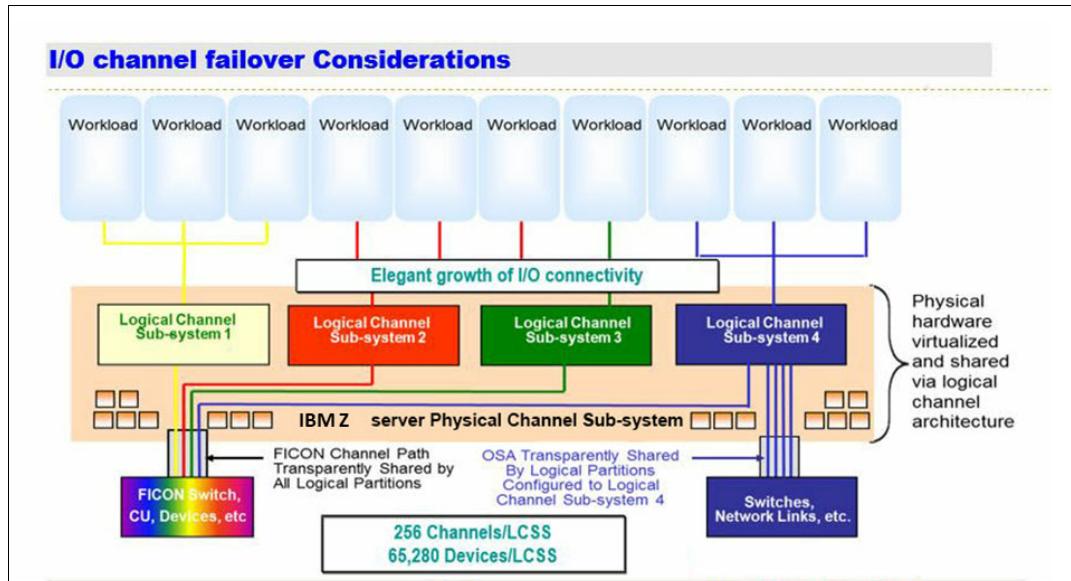


Figure 7-4 Example of an I/O channel failover architecture

Consider the following points:

- ▶ The physical hardware, such as the FICON channels and OSA (Open System Adapter) cards, are shared by using the logical channel architecture.
 - ▶ The processors that run the channel subsystem are called the System Assist Processors (SAP). More than one SAP can be running the channel subsystem. The SAP drives the z Server's I/O channel subsystem, which serves a collection of more than 1,000 high-speed buses.
 - ▶ The SAP relieves the operating system and the general-purpose CPs of much of the work that is required to run I/O operations.
- Specifically, the SAP schedules and starts each I/O request; that is, it locates an available channel path to the requested I/O device and starts the I/O operation. The SAP does not handle the actual data movement between central storage (CS) and the channel.
- ▶ Channels are the communication path from the CS to the control units and I/O devices. They are represented by the black rectangles in Figure 7-4; one for FICON, one for OSA.
 - ▶ Channel Path Identifier (CHPID) is a value that is assigned to each channel path that uniquely identifies that path. You can have a maximum of 256 in an LCSS.
 - ▶ Within the physical I/O subsystem, channel paths and I/O adapters (for example, Open System Adapter Ethernet cards) can be shared, potentially by all logical subsystems. As shown in Figure 7-4, the FICON Channel is being shared by all 4 LCSS, and thus all partitions.
 - ▶ If necessary (for example, for performance reasons), a path or an adapter can be restricted to one LCSS or a subset of the LCSSs. A path or an adapter can be restricted to one LCSS or a subset of the LCSSs (as shown with the OSA on the right in Figure 7-4), which can be used only by LCSS 4 or the blue LPARs that are shown at the top of Figure 7-4.

More information

In this section, we provide more information about the LCSS, its maximums in terms of LPARs, Channels, and Physical devices it supports, and why we must create the LCSS concept to map to the physical channel subsystem.

IBM LinuxONE has a unique channel architecture that is designed to provide powerful and flexible support for the most demanding I/O performance and high-volume workloads. This channel architecture is managed through the foundation technology called LCSS. Each IBM LinuxONE server can have up to four of these LCSSs and each can support 15 LPARs. Each LPAR can address 256 data channels and 64,000 I/O devices. Therefore, a single LinuxONE system can handle over 1,000 data channels and a quarter of million I/O devices.

The LCSS's architecture is further enhanced by the LinuxONE Multiple Image Facility. This Multiple Image Facility allows all 15 LPARs that share a common LCSS to directly access each I/O device without having to forward the request through an intermediate partition, as is the case with UNIX architectures. The direct result of a shorter path length is improved I/O performance.

IBM Z channel spanning allows each device (disk or tape units that are attached by using Fibre Channel technology), to appear to be on any LCSS. The result is that it can be accessed by any partition on IBM LinuxONE. Therefore, greater I/O flexibility and simplified operational management can be achieved.

LinuxONE Supports the following components:

- ▶ Up to four LCSSs
- ▶ Up to 15 LPARs per LCSS
- ▶ Up to 256 Channels per LCSS
- ▶ Up to 256 CHPD per LCSS
- ▶ Up to 64 K Physical Devices per LCSS
- ▶ Up to 1024 CHPIDs made available for entire system
- ▶ Up to 1024 Physical Channels for entire system

Usually in computing, virtualization allows more logical resources than physical. In this case, the physical resources exceed the logical channel addressing ability of the LCSS.

The architecturally defined channel-path identification number, called the *channel-path identifier* (CHPID) must be maintained without change. The CHPID value is defined as an 8-bit binary number that results in a range of unique CHPID values 0 - 255.

Since the inception of the precursor S/370 XA channel-subsystem architecture in the late 1970s, this 8-bit CHPID was maintained without change because of its pervasive use in the IBM z/VM operating system.

For example, the CHPID value is maintained in many internal programming control blocks, is displayed in various operator messages, and is the object of various system commands, programming interfaces, and so on, all of which must be redesigned if the CHPID value was increased to more than an 8-bit number to accommodate more than 256 channel paths.

Therefore, another level of channel-path-addressing indirection was created because of the I/O subsystem that allows more than 256 physical channel paths to be installed and uniquely identified without changing the former 8-bit CHPID value and the corresponding programming dependencies on the CHPID.

The new channel-path-identification value, called the *physical-channel identifier* (PCHID), is a 16-bit binary number 0 - 65,279, which uniquely identifies each physically installed channel path.

With the current IBM LinuxONE servers, a maximum of 1024 external channel paths out of the 65 K (for example, ESCON, FICON, OSA) and 48 internal channel paths (for example, Internal Coupling and IQDIO hyperlink) are each assigned a unique PCHID value of 0 - 2,047.

7.3.6 Government Client Example: Oracle MAA RAC and Data Guard

Oracle's MAA best practices can also be used to provide an Oracle-centric HA and DR solution with IBM LinuxONE. In this scenario, the Oracle Database, RAC, and Data Guard are deployed on IBM LinuxONE (for the database portion of the deployment).

Oracle MAA (see Figure 7-5) provides a comprehensive architecture for reducing time for scheduled and unscheduled outages. An Oracle MAA solution consists of two identical sites: The primary site contains the RAC database, the secondary site contains Physical Standby database or both Physical and Logical one on RAC. Active Data Guard is used for online reporting (extra licensed) at the DR site. Data Guard switchover and failover functions allow the roles to be traded between sites.

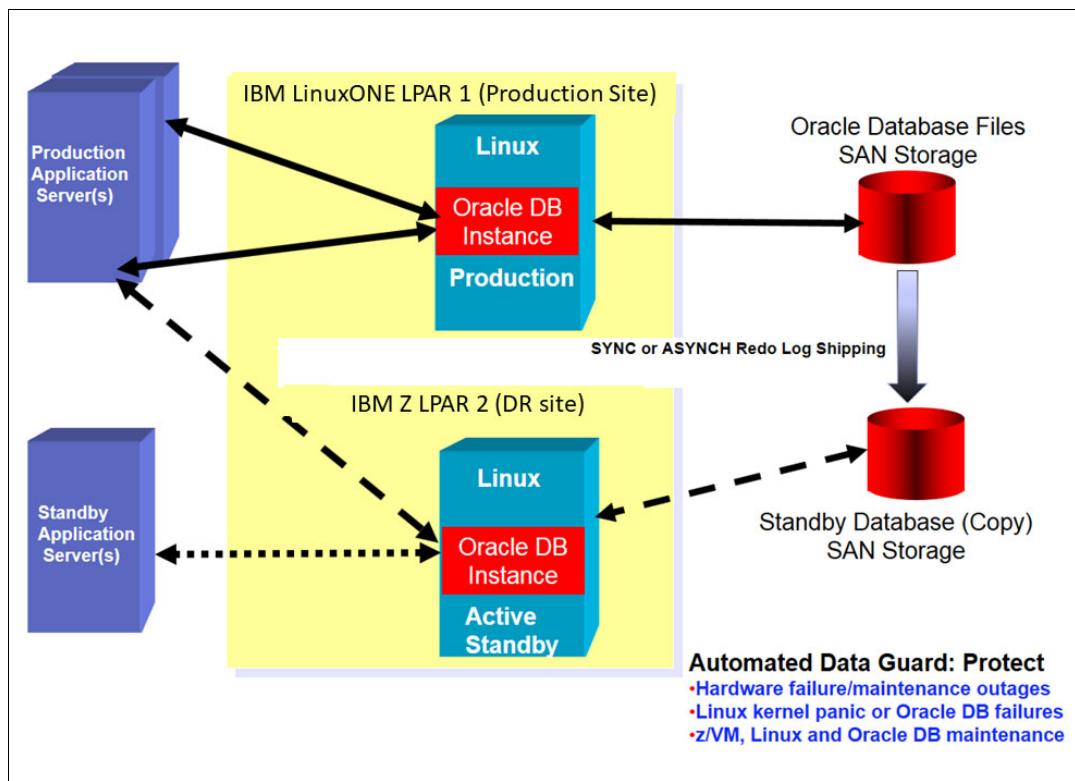


Figure 7-5 Public Sector client MAA example

7.4 Summary and recommendations

In this chapter, we described the main Oracle Database options for HA. We also described a few of the HA options for IBM LinuxONE.

IBM LinuxONE servers have the highest availability rating of any commercially available server on the market today. LinuxONE's ability to perform dynamic reconfigurations to spare out another processor, use RAIM Memory, and use another I/O/network path is true HA.

LinuxOne's virtualization technology has been in the marketplace for over 50 years. It was deployed in Oracle development since the first 10gR2 database was certified on the platform. Decades of experience in error correction and testing methodologies help ensure that the hardware is highly available and that the software also is designed that way.

Combining the highest available server infrastructure of LinuxONE with Oracle's maximum availability architecture creates the most powerful and robust combination available for running business critical Oracle workloads.

IBM LinuxONE is also a good fit for Oracle HA because of LinuxONE's Dynamic capabilities. More memory, CPU, I/O, or network resources can be dynamically turned on and added to a system microcode without the need to shut down an application.

LinuxONE with Oracle's HA can protect an application and a business from unplanned events, which provides excellent business value through Reliability, Availability, Scalability, and Oracle's HA options.

The LinuxONE's hardware stack is fully certified with Oracle for the entire Oracle database Enterprise Edition solution stack to run virtualized under z/VM.

Oracle continues to demonstrate the delivery of their quarterly patch set updates and critical patches for Red Hat and SUSE Linux distributions for LinuxONE currently with other platforms. This commitment to the currency of critical patches helps add to HA capabilities of running the Oracle database with IBM LinuxONE.

Customers who deployed Oracle solutions with IBM LinuxONE enjoy lower total cost of ownership over the entire product lifecycle. They also achieve near immediate return on investment by using its crucial embedded security, near flawless system availability, and reliability.

When it comes to HA, LinuxONE deliver unmatched performance, scalability, and mission-critical reliability in physical, virtual, and cloud environments.



Surrounding Oracle with open source software

In addition to Oracle workloads, LinuxONE offers you the opportunity to combine the best of your enterprise databases and open source technology, including several open source database managers that are supported on LinuxONE.

This chapter includes the following topics:

- ▶ 8.1, “Open source database managers on LinuxONE” on page 112
- ▶ 8.2, “Summary” on page 119

8.1 Open source database managers on LinuxONE

The question is often asked, “Why are Open Source Database Managers gaining popularity?”

Through mergers and acquisitions, experimentation and IT trends, most IT shops seem to have more than one database model and more than one database manager. The innovation in the open source software community created experimentation and even production use of open source database managers (OSDBMs) to handle new data formats that are driven by cloud computing, mobile computing, and the API economy.

Most information technology organizations run their OSDBMs on a separate system from their production Oracle databases. However, this configuration is not necessary with IBM LinuxONE. LinuxONE allows organizations to securely run multiple database managers and database models on the same system, without compromising service level agreements for response time and availability.

This section introduces several popular OSDBMs that can be deployed on LinuxONE alongside your production Oracle databases. Each OSDBM that is described in this chapter was selected based on the significant potential business value from being deployed on LinuxONE.

8.1.1 Database types

The reason for having multiple database managers is not only to serve different applications with different sets of data and different requirements for qualities of service. Some types of data are not well-managed by traditional data models.

Several models are available for storing and managing data. Today, database models and how programs access the data are in two broad categories: SQL (or relational) and NoSQL (Not only SQL or non-relational).

SQL

This section describes relational database managers, which use structured query language (SQL) as the programming language for inserting, deleting, updating, and querying data. This relational model for database management was invented by E.F. Codd of IBM's San Jose Research Laboratory and is the theoretical basis for relational databases, such as the Oracle Database and other proprietary or closed-source SQL relational database managers that run on LinuxONE, such as IBM DB2, IBM Informix®, Microsoft SQL Server, and SAP Adaptive Server Enterprise (formerly Sybase Adaptive Server Enterprise and Sybase SQL Server).

Many other open source relational database managers run on LinuxONE, including the following examples:

- ▶ Apache Derby (previously IBM Cloudscape)
- ▶ MariaDB
- ▶ MySQL (owned by Oracle)
- ▶ PostgreSQL (object-relational)

NoSQL

This category describes databases that were created to meet a new demand as a result of the Web 2.0 era, including modern applications that generate large volumes of data (structured, semi-structured, or unstructured data). NoSQL databases help companies to better manage this type of data through a flexible schema that is boosting the agility and increasing the performance, while sacrificing the Atomicity, Consistency, Isolation, and Durability (ACID) properties of transactions in relational databases.

Therefore, depending on your business needs, data can be stored in document-oriented databases, graph databases, wide column databases, or key-value databases, which are the main types of NoSQL databases.

Several closed-source non-relational databases run on LinuxONE, including IBM Domino® server, IBM FileNet® ECM, and Software AG Adabas. The following open source NoSQL database managers run on LinuxONE:

- ▶ Alfresco
- ▶ Apache Cassandra
- ▶ Apache Geode
- ▶ Apache ZooKeeper (key-value store)
- ▶ Apache CouchDB (document-oriented DB)
- ▶ Couchbase (formerly Membase, a document-oriented DB)
- ▶ MongoDB
- ▶ Neo4j (graph database)
- ▶ PostgreSQL (PostgreSQL also supports several NoSQL models)
- ▶ Redis (key-value store)

Two of these OSDBMs (PostgreSQL and MongoDB) are described next.

8.1.2 PostgreSQL

PostgreSQL is an open source object-relational database management system (ORDBMS). It conforms to current SQL standards and ensures the ACID database properties. This ORDBMS features a large and active community of contributors and is used by companies of all sizes from all around the world.

Many companies choose to use PostgreSQL to build their IT solutions for the following reasons:

- ▶ Performance, robustness, and stability
- ▶ Standards-compliance to ensure data reliability and integrity
- ▶ Optimizing database cost savings
- ▶ Compatibility with the main open source and proprietary databases
- ▶ Reuse skills from other databases

Added value of running PostgreSQL on LinuxONE

LinuxONE capabilities improve PostgreSQL performance and maximize the security of sensitive data, including advanced encryption features. In addition to a significant improvement of the throughput compared to alternative platforms, LinuxONE vertical scaling reduces the complexity of managing different boxes and enhances business agility.

IBM works closely with the PostgreSQL open source community to port and validate PostgreSQL packages on LinuxONE while ensuring that PostgreSQL uses all IBM LinuxONE capabilities. In addition, a partnership between IBM and 2ndQuadrant provides PostgreSQL enterprise support for clients (PostgreSQL running on RHEL or SUSE). For more information about this support, see the [2ndQuadrant website](#).

Where to obtain PostgreSQL open source packages

On LinuxONE, PostgreSQL can be run on RHEL, SUSE, or Ubuntu. Table 8-1 lists the PostgreSQL open source packages that are available from IBM LinuxONE. By using this information, you can get the PostgreSQL open source packages tested and validated by IBM Open-source team or instructions to build the binaries.

Table 8-1 PostgreSQL open source packages

Linux distribution	PostgreSQL 9.6 packages	PostgreSQL 9.4 packages
RHEL	Building PostgreSQL for the three distributions	RHEL6
		RHEL7
SUSE	Building PostgreSQL for the three distributions	SUSE Linux Enterprise Server 12
		SUSE Linux Enterprise Server 11
Ubuntu	Building PostgreSQL for the three distributions	Ubuntu 16: Available by default in Ubuntu distribution

8.1.3 MongoDB overview

MongoDB is a NoSQL Document Oriented Database. It stands out from relational databases by using dynamic schemas, with which records can be inserted without creating an initial schema to define data structure. Therefore, fields and their values can be easily modified to map application changes without interruption. This OSDBM is widely used for mobile apps, real-time analytics, product catalogs, and content management systems. It can also be used for many other use cases, such as storing stream of data from IOT.

MongoDB stores data in a JSON-like documents (aligning data storage formats with modern programming languages that are used by developers) instead of columns and rows in relational databases. Each document can contain different fields, which contain a value that belongs to the same data type including subdocuments and arrays.

Instead of having related data that is represented by different tables, the cost of joining separate tables is eliminated by storing the linked objects in the same document. This configuration reduces complexity and simplifies data access. Similar Documents are organized into collections in the database.

Figure 8-1 shows the difference between MongoDB document and relational data models.

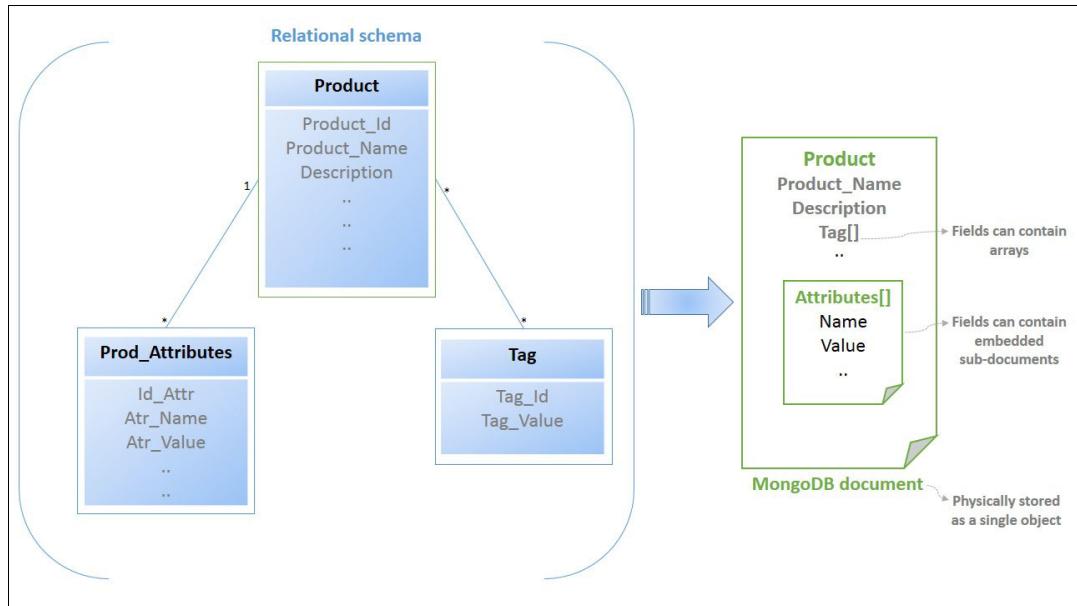


Figure 8-1 MongoDB document model versus relational data model

Table 8-2 lists some of the concepts that are used in Mongodbs and their counterparts in relational databases; for example, Oracle database.

Table 8-2 MongoDB versus. Oracle database

MongoDB	Oracle database
JSON-like document	Row
Collection	Table
Embedded subdocuments and Linking	Join
Index	Index

Added value of running MongoDB on LinuxONE

With LinuxONE, many cores can be used (scale 1 - 141 cores), which allows MongoDB to scale vertically (up to 2 TB single-node MongoDB) without the use of the sharding technique, which is also known as horizontal scaling, to split a large amount of data across different servers. This method is the best way to remove the overhead of sharding and reduce the complexity of managing different Mongodbs shards.

The network communication between the MongoDB shards is also drastically reduced (near zero latency) by using IBM HiperSockets technology that is used in LinuxONE to accelerate the communication between the different LPARs.

The enterprise edition of MongoDB (called MongoDB Enterprise Advanced) is supported on LinuxONE.

In addition, the capabilities of LinuxONE in terms of security (EAL5+ certification,) high availability, and resiliency can be used to efficiently enhance MongoDB's capabilities and achieve the most stringent market requirements.

Obtaining MongoDB packages

For more information, see the following resources:

- ▶ The enterprise edition of MongoDB can be downloaded from the [MongoDB Download Center website](#).

At the website, click the **Enterprise Server** tab. Then, find your Linux distribution (s390x) by using the Platforms drop-down list and click **Download**.

- ▶ The open source packages that are tested and validated by IBM Open-source team and instructions to build the binaries can be found at the [Building MongoDB website](#).

8.1.4 Integrating Apache Spark with Oracle and MongoDB databases that are running on LinuxONE

LinuxONE can be used to run open source databases on the same server as enterprise databases. At the same time, the same quality of service can be maintained and client requirements can be met in terms of performance, high availability, data security, and cost optimization.

One important need that all companies must meet consists of extending the value of internal and external data by extracting useful information at the correct time by using Analytics, which is a data analysis process that is necessary for the survival of any business.

As a result of this critical business need, IBM LinuxONE provides support for one of the most popular open source analytics frameworks, which is called Apache Spark. With Apache Spark on LinuxONE, organizations can implement their critical analytics use cases without moving sensitive data, while taking advantage of LinuxONE capabilities that are coupled with a powerful analytics framework. This combination simplifies and accelerates data analysis, especially when the different databases are running on the same server.

In this section, we consider a sample analytics use case that involves a data integration challenge for a client that is running Oracle and MongoDB databases. This use case shows how Apache Spark on LinuxONE can help you perform analytics by easily combining and integrating SQL and NoSQL without any external data movement.

This example is intended for illustrative purposes only. You might be able to build your own use cases by mixing different sources of data from other databases that have not been mentioned here, with the possibility of including external data; for example, social data, depending on your business needs.

The following topics are described next:

- ▶ An overview of Apache Spark
- ▶ Use case overview: Integration between Oracle data and MongoDB data
- ▶ Use case implementation by using Apache Spark

Apache Spark overview

Apache Spark is an open source cluster in-memory computing engine for fast and efficient execution of complex analytics. It includes the following features:

- ▶ Parallel processing compute engine: Spark core engine with High-level APIs in Scala, Java, Python, and R.
- ▶ Integrated tools (see Figure 8-2 on page 117) for SQL and structured data processing, machine learning (MLib), graph processing, and streaming.

Figure 8-2 shows a high-level overview of Apache Spark.

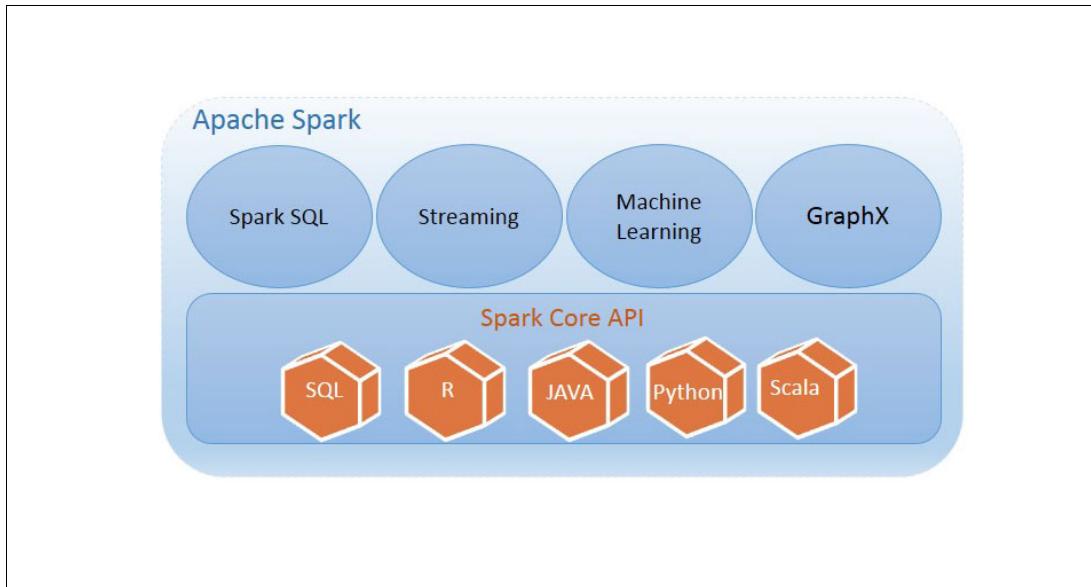


Figure 8-2 High-level overview of Apache Spark

Use case overview: Integration between Oracle data and MongoDB data

As a solution to new business needs, some clients choose to combine SQL and NoSQL data stores. In this example, the client chose to use MongoDB for its new web application to store the multiple types of products that are presented on the website. This configuration benefits from MongoDB's simple and flexible data model because MongoDB is a good fit to build e-commerce product catalogs compared to RDBMS. Oracle database is still used as a system of record to perform complex transactions, such as order management.

For example, to obtain details about the most purchased type of product by women during a certain period for each region, the data analysis must go through Oracle tables, such as orders table, and MongoDB documents, such as products collection. This process appears to be simple, but it is a meticulous process that can take a long time to receive analytics results. Therefore, the issue becomes how Apache Spark can help to perform this data analytics with the best performance and the least effort possible.

Use case implementation that uses Apache Spark

Among the concepts that are used by Apache Spark to catch data from different sources, we can find DataFrames. A DataFrame is an immutable distributed collection of data that is organized into named columns and partitioned across nodes in a cluster. It features the benefits of Spark SQL's optimized execution engine and provides relational view of data to perform data analysis, especially for people with SQL skills.

In this case, you can use these DataFrames to load the different data from Oracle and MongoDB databases that are running on LinuxONE. After the DataFrames are instanced, you can easily use the different algorithms and high-level operators that are provided by Apache Spark to transform these DataFrames (map, filter, join, and so on).

Operations, such as complex joins between data from MongoDB and Oracle, can be performed in a short time without taking into account all complications and costs that are related to data movement between different physical servers. It also benefits from the memory processing capacity of Apache Spark.

Figure 8-3 shows an overview of the use of Apache Spark (Spark SQL module) with Oracle and MongoDB databases to perform analytics operations on two types of data (SQL: Table “Order” and NoSQL: Document “Product”).

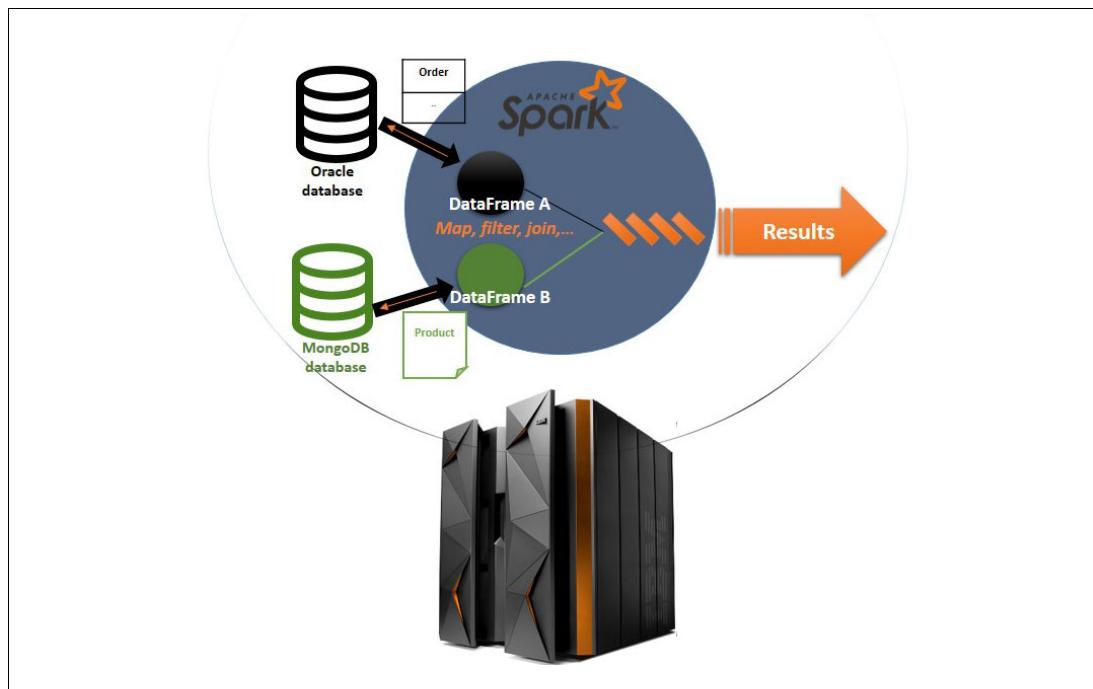


Figure 8-3 Data integration use case that uses Apache Spark

Loading data from Oracle database into Apache Spark

In this section, we describe an example of how to load data from Oracle database 12c into Apache Spark 2.1.0 (both running on LinuxONE) by using JDBC driver and DataFrame API.

Note: The IBM packages for Apache Spark can be found at the [IBM developerWorks website](#).

Complete the following steps:

1. Download Oracle Database 12c JDBC driver from the [Oracle Database 12.1.0.1 JDBC Driver and UCP Downloads website](#).
2. Go to the directory where you installed Spark. Then, start the Spark shell and specify the path location of the JDBC driver to use, by running the command as shown in the Example 8-1.

Example 8-1 Command to specify JDBC driver path location

```
sparkdvl:/opt/spark/spark-dk-2.0.0.0/spark/bin # ./spark-shell --driver-class-path  
ojdbc6.jar --jar s ojdbc6.jar
```

After running the command shown in Example 8-1, you should see the Spark interactive shell in Scala, as shown in Figure 8-4 on page 119.

```

Spark context Web UI available at http://10.3.58.38:4040
Spark context available as 'sc' (master = local[*], app id = local-1485110748668).
Spark session available as 'spark'.
Welcome to

    / \ \ / \
   / \ \ / \
  / \ \ / \
 / \ \ / \
/ \ \ / \
version 2.0.0

Using Scala version 2.11.8 (IBM J9 VM, Java 1.8.0)
Type in expressions to have them evaluated.
Type :help for more information.

scala> ■

```

Figure 8-4 Spark shell

Note: In this example, the Oracle database JDBC driver is in the same directory where we can run the Spark Shell.

3. Before establishing the connection with the database, ensure that the Oracle listener is running by using the `lsnrctl status` command.

Example 8-2 shows the command that is used to connect an Oracle database to Apache Spark by using the JDBC driver and read data from the Oracle table “Users” into a “jdbcDF” DataFrame.

Example 8-2 Command to connect to an Oracle database and its resulting output

```

scala> var jdbcDF =
spark.read.format("jdbc").option("url","jdbc:oracle:thin:@10.3.58.125:1521/orcl.mo
p.fr.ibm.com").option("dbtable","narjisso.ORDERS").option("user",
"narjisso").option("password","*****").load()
jdbcDF: org.apache.spark.sql.DataFrame = [ORDER_ID: decimal(6,0), STATUS: string
... 7 more fields]

```

A successful execution of this command returns a DataFrame that contains all data read from table “Users” that can be used for future SQL queries with other DataFrames.

A similar process can be used to read MongoDB documents into DataFrames by using the MongoDB Spark connector, which simplifies the integration between different data sources and rapid and efficient implementation of analytics use cases.

8.2 Summary

In a data-driven world, IBM offers a server that provides the flexibility and stability to run a wide range of open source products next to Oracle databases. This ability enables companies to benefit from the constant innovation that stems from Open source world.

By choosing LinuxONE, organizations can benefit from a secure, highly available, and easy-to-manage environment to consolidate their data while optimizing overall costs.



A

Oracle Database 11gR2 on Red Hat Enterprise Linux Servers 7

In this appendix, we describe the steps for installing Oracle 11g Release 2 (11gR2) with Red Hat 7 and Oracle workloads on LinuxONE.

This appendix includes the following topics:

- ▶ A.1, “Pre-Installation requirements documentation” on page 122
- ▶ A.2, “Obtaining the code for Oracle 11gR2 on IBM LinuxONE” on page 122
- ▶ A.3, “Java version for 11gR2 installations on Red Hat 7” on page 122
- ▶ A.4, “Oracle RAC installation issue when version 11.2.0.4 is used with Red Hat Enterprise Linux server 7” on page 123
- ▶ A.5, “Security-Enhanced Linux” on page 124
- ▶ A.6, “Pre-installation checking” on page 124
- ▶ A.7, “Oracle Universal Installer” on page 125

A.1 Pre-Installation requirements documentation

For more information about the pre-installation phase, see *Oracle Database (RDBMS) on UNIX, IBM AIX, HP-UX, Linux, Mac OS X, Solaris, Tru64 UNIX Operating Systems Installation and Configuration* guide.

For more information about Oracle Database 11.2.0.4 on Red Hat 7 requirements, see Doc ID 1967531.1

A.2 Obtaining the code for Oracle 11gR2 on IBM LinuxONE

Complete the following steps:

1. Obtain the most recent Oracle Database code.
2. Download Oracle 11.2.0.4 for [My Oracle Support Patch: 13390677](#).

A.3 Java version for 11gR2 installations on Red Hat 7

The Oracle distribution code (11.2 and 12c) includes the IBM Java Runtime Environment (JRE) as part of the distribution. Oracle Database 11gR2 Version 11.2.0.4 is included with JRE 1.5, which does not run on Red Hat 7 systems.

The Oracle Universal Installer (OUI) fails for 11.2.0.4 installations (bug 20370383) on Red Hat 7 installations with the base JRE (1.5) installed with Oracle. The JRE that is supplied with Oracle 12c installations is 1.6 and requires no update.

Download and install the IBM Java 1.6 (IBM JDK 1.6 is recommended for Oracle 11.2.0.4 installations only) from the [IBM Java SDK Downloads website](#).

You might need to login with your IBM ID or create an ID if you do not yet have an ID. Complete the following steps:

1. Select the **Java 6** tab.
2. For IBM LinuxONE, scroll down to the Linux on z Systems® 64-bit download area.
Download the SDK or the JRE, depending on your needs (see Figure A-1 on page 123).
The SDK and JRE are indicated in the file name.

Linux on z Systems 64-bit

 [Installable package \(InstallAnywhere as root\)](#)
(File name: ibm-java-s390x-sdk-6.0-16.35.bin, Size: 96MB)

 [Simple unzip with license \(InstallAnywhere root not required\)](#)
(File name: ibm-java-sdk-6.0-16.35-linux-s390x.bin, Size: 96MB)

 [Installable package \(InstallAnywhere as root\)](#)
(File name: ibm-java-s390x-jre-6.0-16.35.bin, Size: 75MB)

 [Simple unzip with license \(InstallAnywhere root not required\)](#)
(File name: ibm-java-jre-6.0-16.35-linux-s390x.bin, Size: 75MB)

Figure A-1 IBM LinuxONE 64-bit download - select Linux on z Systems 64-bit

3. Copy the .bin file to each node in your cluster.
4. Install the IBM JDK (the specific bundle to develop software in the Java language) by running the .bin file as the root user, as shown in the following example:


```
# chmod u+x ibm-java-s390x-jre-6.0-16.35.bin
# ./ibm-java-s390x-jre-6.0-16.35.bin
```
5. Follow the prompts to install the JDK to the wanted location (selecting the appropriate language and installation directories).

A.4 Oracle RAC installation issue when version 11.2.0.4 is used with Red Hat Enterprise Linux server 7

Acquire Oracle patch 18370031 before starting an Oracle RAC/Grid Infrastructure installation for Oracle 11.2.0.4 installations on Red Hat 7 systems. Starting in Red Hat 7, the new systemd services require an Oracle patch for the root.sh script to successfully complete in each Oracle RAC node.

Complete the following steps:

1. Download Patch 18370031 from My Oracle Support.
2. Start an interactive Oracle Grid Infrastructure installation by using the Oracle Universal Installer (OUI), as described as in “Interactive installation” on page 20. However, do *not* run root.sh on any node until *after* Patch 18370031 is applied.
3. When the OUI prompts the user to run the root.sh scripts, Patch 18370031 should be applied by following the instructions in the README file that is supplied with the patch.

If a software-only grid installation is run, the patch should be applied after the installation concludes but before any configuration is attempted.

A.5 Security-Enhanced Linux

In this section, we describe some information important to note before the installation process is started.

Security-Enhanced Linux (SELinux) is a Linux kernel security module that provides a mechanism for supporting access control security policies for security of Red Hat operating systems.

By default, Red Hat Enterprise Linux server versions 6 and 7 are installed with SELinux in enforcing mode (meaning that the SELinux policy is in effect and it enforces access denials and audits them). This configuration is acceptable for the 11gR2 installation process for Red Hat Enterprise Linux server version 6 and 7.

Previously on Red Hat Enterprise Linux server version 5, to run sqlplus, operators switched SELinux to the Permissive mode or sqlplus failed to start and produced the error that is shown in Example 8-3.

Example 8-3 The sqlplus error message

```
./sqlplus: error on libnnz11.so: cannot restore segment prot after relo...
```

For more information, see Oracle support NOTE 454196.1.

A.6 Pre-installation checking

This section describes more pre-installation checking that you must perform after ensuring a GUI is established.

A.6.1 Operating system requirements for IBM LinuxONE

When IBM is used, RHEL 7 servers must be running Red Hat kernel 3.10.0-229.el7 (s390x) or higher (Red Hat 7.2+ is recommended).

A.6.2 Oracle RPM Checker with Red Hat Enterprise Linux Server on LinuxONE

In this section, we describe how to use the Oracle RPM Checker utility to verify that the required Linux packages are installed on the operating system before starting the Oracle Database or Oracle Grid Infrastructure installation.

Download the Oracle 11.2.0.4 for Red Hat 7 rpm checker from the following [My Oracle Support website](#).

After you download the appropriate RPM checker on LinuxONE, decompress the rpm and install it as root to assess the missing rpms by using the following command (depending on your distribution level):

```
yum install ora-val-rpm-RH7-DB-11.2.0.4-1.s390x.rpm
```

By using this command, the required rpms are automatically pulled in if your yum repository is set up. If your yum repository is not set up, you can install the rpm dependency checker as shown in the following example:

```
rpm -ivh ora-val-rpm-RH7-DB-11.2.0.4-1.s390x.rpm
```

A.7 Oracle Universal Installer

By using Oracle Universal Installer (OUI), you can install software only, or install and create a database. Other installers are available, such as netca, to create the Oracle listener services, or dbca, to create databases. The Oracle Installers can be run interactively or in silent mode.

Note: If an Oracle 11.2.0.4 Installation on Red Hat 7 is running, you must start the OUI as shown in the following example:

```
./runInstaller -ignorePrereq -ignoreSysPrereqs -jreloc <IBM jre_1.6_home>
```

An example of the output from this command is shown in Figure A-2.

```
[oracle@orallab01 database]$ ./runInstaller -ignorePrereq -ignoreSysPrereqs -jreloc /opt/ibm/java-s390x-60/jre
Starting Oracle Universal Installer...
Checking Temp space: must be greater than 500 MB.  Actual 13001 MB      Passed
Checking swap space: must be greater than 150 MB.  Actual 781 MB      Passed
Checking monitor: must be configured to display at least 256 colors.  Actual 16777216      Passed
Preparing to launch Oracle Universal Installer from /tmp/OraInstall2017-02-20 12-17-57PM. Please wait ...[oracle@orallab01 database]
```

Figure A-2 OUI Installation command for Red Hat 7 and Oracle 11.2.0.4

Related publications

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

IBM Redbooks

The following IBM Redbooks publications provide additional information about the topic in this document. Note that some publications referenced in this list might be available in softcopy only:

- ▶ *Experiences with Oracle Database 12c Release 1 on Linux on System z*, SG24-8159
- ▶ *Linux on IBM System z: Performance Measurement and Tuning*, SG24-6926
- ▶ *Experiences with Oracle 11gR2 on Linux on System z*, SG24-8104

You can search for, view, download or order these documents and other Redbooks, Redpapers, Web Docs, draft and additional materials, at the following website:

ibm.com/redbooks

Online resources

The following websites are also relevant as further information sources:

- ▶ My Oracle Support (registration required):
<https://support.oracle.com>
- ▶ Oracle Help Center
<http://docs.oracle.com>

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services



SG24-8384-01

ISBN 0738457957

Printed in U.S.A.

Get connected

