THE ORACLE 11G PERFORMANCE TUNING

THE ORACLE 11G PERFORMANCE TUNING

By

Rajeev A. Parida
MCA, OCP

UNIVERSITY SCIENCE PRESS

(An Imprint of Laxmi Publications (P) Ltd.)

BANGALORE • CHENNAI • COCHIN • GUWAHATI • HYDERABAD
JALANDHAR • KOLKATA • LUCKNOW • MUMBAI • RANCHI • NEW DELHI
INDIA • USA • GHANA • KENYA

THE ORACLE 11G PERFORMANCE TUNING

Printed and bound in India
Typeset at Excellent Graphics
First Edition: 2015
UOP-9735-395-ORACLE 11G PERFORM TUNE-PAR
ISBN 978-93-83828-31-9

Price: ₹ 395.00

Published in India by

UNIVERSITY SCIENCE PRESS
(An Imprint of Laxmi Publications Pvt.Ltd.)
113, GOLDEN HOUSE, DARYAGANJ,
NEW DELHI - 110002, INDIA
Telephone : 91-11-4353 2500, 4353 2501
Fax : 91-11-2325 2572, 4353 2528
www.laxmipublications.com info@laxmipublications.com

Branches

✆	Bangalore	080-26 75 69 30	
✆	Chennai	044-24 34 47 26,	24 35 95 07
✆	Cochin	0484-237 70 04,	405 13 03
✆	Guwahati	0361-254 36 69,	251 38 81
✆	Hyderabad	040-27 55 53 83,	27 55 53 93
✆	Jalandhar	0181-222 12 72	
✆	Kolkata	033-22 27 43 84	
✆	Lucknow	0522-220 99 16	
✆	Mumbai	022-24 91 54 15,	24 92 78 69
✆	Ranchi	0651-220 44 64	

C—
Printed at:

Contents

Chapters *Pages*

Chapters *Pages*

Chapters *Pages*

PREFACE

In today's Internet-centric economy, too many organizations respond to Oracle performance problems by throwing money at these problems by buying larger and more expensive computers or by hiring expert consultants. But there's a lot you can do on your own to increase dramatically the performance of your existing system. Whatever version of Oracle you're running from Version 10g to Oracle 11g, proper tuning can save your organization a huge amount of money in additional equipment, extra memory, and hardware upgrades. The first edition of Oracle 11g Performance Tuning became a classic for programmers, managers, database administrators, system administrators, and anyone who cares about improving the performance of an Oracle database system.

Oracle 11g Performance Tuning is a must-have reference guide for all Oracle professionals. It provides much-needed information on best practices, tips, and techniques in debugging and tuning of the Oracle 11g database. It will bring you the insider information that you can't get anywhere else. It is a book for people who administer the operation of an Oracle Database system. Referred to as database administrators (DBAs), they are responsible for creating Oracle Database, ensuring its smooth operation, and monitoring its use.

PURPOSE OF THIS BOOK

My main purpose is to give you a foundation for using, the Oracle database effectively and efficiently. Therefore, I wrote with these principles in mind:

- I have tried to concentrate on the most important Oracle 11g performance tuning issues. Every topic provides a comprehensive but concise discussion of how Oracle 11g handles a tuning issue and the repercussions of that action.
- One of the first decisions I made was to concentrate on principles rather than syntax. There simply isn't room for myriad syntax diagrams and examples in this book.
- I have tried to make this an ideal first oracle 11g tuning book for a wide spectrum of oracle users but not the last you will very likely have to refer to Oracle documentation or other, more specific books for more details about using Oracle 11g tuning. However, I hope this book will act as information from other sources and put it to the best use.

This book is the result of over 14 years of experience with oracle and other databases. I have tried to apply that experience as best I can here.

WHO THIS BOOK IS FOR

This book is an aid for people responsible for the operation, maintenance, and performance of Oracle database. This book describes detailed ways to enhance Oracle performance by writing and tuning SQL properly, using performance tools, and optimizing instance performance. It also explains how to create an initial database for good performance and includes performance-related reference information. This book could be useful for database administrators, application designers, and programmers.

Good Luck, be patient, and I look forward to work with you.

– AUTHOR

ACKNOWLEDGEMENT

"You must be the change you wish to see in the world"

Mahatma Gandhi

There have been many who have helped me become who I currently am, and who have directly or indirectly participated in this project. I must start with my gratitude to God, the source of all wisdom, knowledge, and ever-present help. On earth, I acknowledge the consistent support from my wife Anupama and my son Ayush. They patiently put up with my absences as I sat at the PC, typing away late in the night and on weekends, missing family time together. Without Anu, none of this would have been possible. She is not only a wonderful mother but also my true love and best friend.

I would like to express my special gratitude to my wife Anupama Parida for her eagle eye and gentle correction of my text and her expert technical review of the content and her encyclopedic knowledge of all things oracle.

I would also like to thank the following technical reviewers who have provided very useful and critical comments to improve the presentation and content of the book in its first edition: Mr. Kaushal Sawhney, NIIT Technologies Ltd, India, Mr. Nick Lukin, Oracle Corporation; Ms. Binodini Paital, IBM, India ; Mr. Paul Anderson, Accenture, Australia.

A book such as this would not have seen the light of day without the dedicated efforts of all the Firewall Media Publishing staff who helped me.

Many of my professional colleagues at NIIT Technologies Ltd were a source of both inspiration and guidance, especially Late Amitabh Gupta, Somnath Das Gupta, Biswarup Dey, Sameer Grover, Sanjay Karambelkar, Amiya Maharatha , Shashank Batra, Neeraj Jain, Naveen Kumar and Pravat Roy. Thanks to all of those above who made both big and small differences in my life and in this book.

An acknowledgement to all those whom I have worked with in the past is most in order. I dare not print a list of all those people for fear of leaving someone out. To all of you I owe more thanks than I can say. Thanks to previous employers who gave me wonderful opportunities.

Furthermore, my special thanks to Mr. Abhiram Behera and Mrs. Nirupama Behera for formulating the original concept and made many suggestions for the topic covered in this book via email from Florida, USA.

Last but not least, to my parents, thanks for your enduring level of patience, love, encouragement, support, and sacrifice during the past 24 months. Without them, none of this would have been possible.

– AUTHOR

1 Getting Started with Oracle Architecture

Learning Objectives

This chapter provides an introduction to performance tuning and contains the following sections:

- Oracle Database
- Connecting to a Server
- Oracle Database Server Architecture
- Instance: Database Configurations
- Connecting to the Database Instance
- Oracle Database Memory Structures
- Shared Pool
- Database Buffer Cache
- Program Global Area (PGA)
- Process Architecture
- Database Storage Architecture
- Logical and Physical Database Structures
- SYSTEM and SYSAUX Tablespaces
- Automatic Storage Management
- Introduction to Performance Tuning
- SQL Tuning
- Introduction to Performance Tuning Features and Tools
- Additional Oracle Database Tools

ORACLE DATABASE

A database is a collection of data treated as a unit. The purpose of a database is to store and retrieve related information.

The Oracle Relational Database Management System (RDBMS) reliably manages a large amount of data in a multiuser environment so that many users can concurrently access the same data. This is accomplished while delivering high performance. At the same time, it prevents unauthorized access and provides efficient solutions for failure recovery.

CONNECTING TO A SERVER

A database user can connect to an Oracle server in one of three ways:

The user logs on to the operating system running the Oracle instance and starts an application or tool that accesses the database on that system. The communication pathway is established using the interprocess communication mechanisms available on the host operating system.

The user starts the application or tool on a local computer and connects over a network to the computer running the Oracle database. In this configuration (called *client/server*), network software is used to communicate between the user and the back-end server. The client/server architecture database system has two parts: a front end (client) and a back end (server) connected through a network. Network software is used to communicate between the user and the Oracle server.

The client is a database application that initiates a request for an operation to be performed on the database server. It requests, processes, and presents data managed by the server. The client workstation can be optimized for its job. For example, the client might not need large disk capacity, or it might benefit from graphic capabilities. Often, the client runs on a different computer than the database server. Many clients can simultaneously run against one server.

The server runs Oracle Database software and handles the functions required for concurrent, shared data access. The server receives and processes requests that originate from client applications. The computer that manages the server can be optimized for its duties. For example, the server computer can have large disk capacity and fast processors.

The user accesses an application server through a tool (such as a Web browser) on the local computer (client). The application server then interacts with a back-end database server on behalf of the client.

A traditional multitier architecture has the following components:

- A client or initiator process that starts an operation.
- One or more application servers that perform parts of the operation. An application server contains a large part of the application logic, provides access to the data for the client, and performs some query processing, thus removing some of the load from the database server. The application server can serve as an interface between clients and multiple database servers and can provide an additional level of security.

An end server or database server that stores most of the data used in the operation.

This architecture enables use of an application server to do the following:

- Validate the credentials of a client (such as a Web browser).
- Connect to an Oracle Database server.
- Perform the requested operation on behalf of the client.

ORACLE DATABASE SERVER ARCHITECTURE

There are three major structures in Oracle database server architecture: memory structures, process structures, and storage structures. A basic Oracle database system consists of an Oracle database and a database instance.

The database consists of both physical structures and logical structures. Because the physical and logical structures are separate, the physical storage of data can be managed without affecting access to logical storage structures.

The instance consists of memory structures and background processes associated with that instance. Every time an instance is started, a shared memory area called the System Global Area (SGA) is allocated and the background processes are started. Processes are jobs that work in the memory of computers. A process is defined as a "thread of control" or a mechanism in an operating system that can run a series of steps. After starting a database instance, the Oracle software associates the instance with a specific database. This is called *mounting the database*. The database is then ready to be opened, which makes it accessible to authorized users.

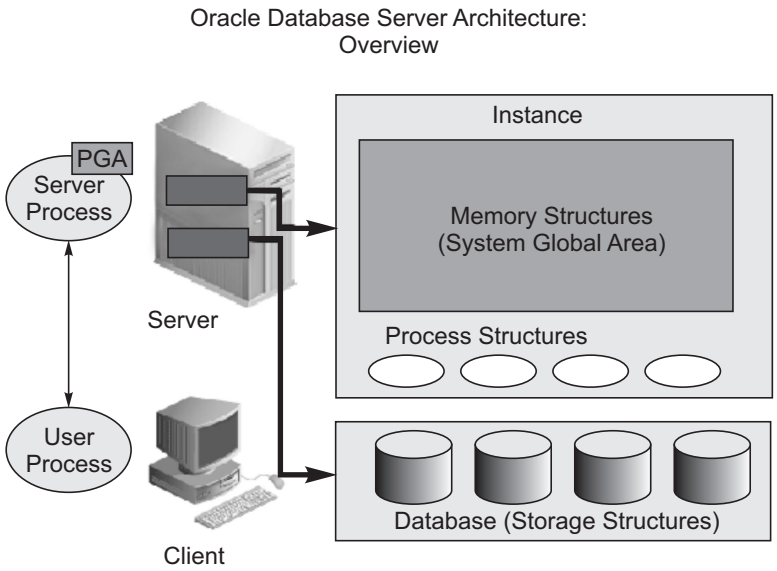

Figure 1.1. Oracle Architecture

The Oracle Automatic Storage Management (ASM) uses the concept of an instance for the memory and process components, but is not associated with a specific database.

INSTANCE: DATABASE CONFIGURATIONS

Each database instance is associated with one and only one database. If there are multiple databases on the same server, then there is a separate and distinct database instance for each database. A database instance cannot be shared. A Real Applications Cluster (RAC) database usually has multiple instances on separate servers for the same shared database. In this model, the same database is associated with each RAC instance, which meets the requirement that at most only one database is associated with an instance.

CONNECTING TO THE DATABASE INSTANCE

Connections and sessions are closely related to user processes but are very different in meaning.

A *connection* is a communication pathway between a user process and an Oracle Database instance. A communication pathway is established using available interprocess communication mechanisms (on a computer that runs both the user process and Oracle Database) or network software (when different computers run the database application and Oracle Database, and communicate through a network).

A *session* represents the state of a current user login to the database instance. For example, when a user starts SQL*Plus, the user must provide a valid username and password, and then a session is established for that user. A session lasts from the time a user connects until the user disconnects or exits the database application.

Multiple sessions can be created and exist concurrently for a single Oracle database user using the same username. For example, a user with the username/password of HR/HR can connect to the same Oracle Database instance several times.

ORACLE DATABASE MEMORY STRUCTURES

Oracle Database creates and uses memory structures for various purposes. For example, memory stores program code being run, data that is shared among users, and private data areas for each connected user. Two basic memory structures are associated with an instance:

- **System Global Area (SGA):** Group of shared memory structures, known as SGA components, that contain data and control information for one Oracle database instance. The SGA is shared by all server and background processes. Examples of data stored in the SGA include cached data blocks and shared SQL areas.
- **Program Global Areas (PGA):** Memory regions that contain data and control information for a server or background process. A PGA is nonshared memory created by Oracle Database when a server or background process is started. Access to the PGA is exclusive to the server process. Each server process and background process has its own PGA.

The SGA is the memory area that contains data and control information for the instance. The SGA includes the following data structures:

- **Shared pool:** Caches various constructs that can be shared among users
- **Database buffer cache:** Caches blocks of data retrieved from the database
- **KEEP buffer pool:** A specialized type of database buffer cache that is tuned to retain blocks of data in memory for long periods of time
- **RECYCLE buffer pool:** A specialized type of database buffer cache that is tuned to recycle or remove block from memory quickly
- **nK buffer cache:** One of several specialized database buffer caches designed to hold block sizes different than the default database block size
- **Redo log buffer:** Caches redo information (used for instance recovery) until it can be written to the physical redo log files stored on the disk
- **Large pool:** Optional area that provides large memory allocations for certain large processes, such as Oracle backup and recovery operations, and I/O server processes
- **Java pool:** Used for all session-specific Java code and data in the Java Virtual Machine (JVM)
- **Streams pool:** Used by Oracle Streams to store information required by capture and apply.

When you start the instance by using Enterprise Manager or SQL*Plus, the amount of memory allocated for the SGA is displayed.

A Program Global Area (PGA) is a memory region that contains data and control information for each server process. An Oracle server process services a client's requests. Each server process has its own private PGA that is allocated when the server process is started. Access to the PGA is exclusive to that server process, and the PGA is read and written only by the Oracle code acting on its behalf. The PGA is divided into two major areas: stack space and the User Global Area (UGA).

With the dynamic SGA infrastructure, the sizes of the database buffer cache, the shared pool, the large pool, the Java pool, and the Streams pool can change without shutting down the instance.

The Oracle database uses initialization parameters to create and manage memory structures. The simplest way to manage memory is to allow the database to automatically manage and tune it for you. To do so (on most platforms), you only have to set a target memory size initialization parameter (MEMORY_TARGET) and a maximum memory size initialization parameter (MEMORY_MAX_TARGET).

SHARED POOL

The shared pool portion of the SGA contains the library cache, the data dictionary cache, the SQL query result cache, the PL/SQL function result cache, buffers for parallel execution messages, and control structures.

The *data dictionary* is a collection of database tables and views containing reference information about the database, its structures, and its users. Oracle Database accesses the data dictionary frequently during SQL statement parsing. This access is essential to the continuing operation of Oracle Database.

The data dictionary is accessed so often by Oracle Database that two special locations in memory are designated to hold dictionary data. One area is called the *data dictionary cache*, also known as the row cache because it holds data as rows instead of buffers (which hold entire blocks of data). The other area in memory to hold dictionary data is the *library cache*. All Oracle Database user processes share these two caches for access to data dictionary information.

Oracle Database represents each SQL statement that it runs with a shared SQL area (as well as a private SQL area kept in the PGA). Oracle Database recognizes when two users are executing the same SQL statement and reuses the shared SQL area for those users.

A shared SQL area contains the parse tree and execution plan for a given SQL statement. Oracle Database saves memory by using one shared SQL area for SQL statements run multiple times, which often happens when many users run the same application.

When a new SQL statement is parsed, Oracle Database allocates memory from the shared pool to store in the shared SQL area. The size of this memory depends on the complexity of the statement.

Oracle Database processes PL/SQL program units (procedures, functions, packages, anonymous blocks, and database triggers) in much the same way it processes individual SQL statements. Oracle Database allocates a shared area to hold the parsed, compiled form of a program unit. Oracle Database allocates a private area to hold values specific to the session that runs the program unit, including local, global, and package variables (also known as package instantiation) and buffers for executing SQL. If more than one user runs the same program unit, then a single, shared area is used by all users, while all users maintain separate copies of their own private SQL areas, holding values specific to their own sessions.

Individual SQL statements contained in a PL/SQL program unit are processed just like other SQL statements. Despite their origins in a PL/SQL program unit, these SQL statements use a shared area to hold their parsed representations and a private area for each session that runs the statement.

The SQL query result cache and PL/SQL function result cache are new to Oracle Database 11*g*. They share the same infrastructure, appear in the same dynamic performance (V$) views, and are administered using the same supplied package.

Results of queries and query fragments can be cached in memory in the SQL query result cache. The database can then use cached results to answer future executions of these queries

The Oracle 11G Performance Tuning By Rajeev A. Parida

Publisher : Laxmi Publications ISBN : 9789382828319 Author : Rajeev A. Parida