International Journal of Engineering and Management Research e-ISSN: 2250-0758 | p-ISSN: 2394-6962
Volume-10, Issue-5 (October 2020)
www.ijemr.net https://doi.org/10.31033/ijemr.10.5.11

Database Performance Management in Cloud

Navendu Mishra[1], Jayant Sawarkar[2] and Suraj Dubey[3]
[1]Student, Department of Computer Science & Engineering, Birla Institute of Technology & Science, Pilani, INDIA
[2]Assistant Professor, Department of Computer Science & Engineering, Datta Meghe College of Engineering, Navi Mumbai, INDIA
[3]COO, Osource Global Private Limited, Mumbai, INDIA

[2]Corresponding Author: jdsawarkar@yahoo.com

ABSTRACT

Managing large chain of Hotels and ERP database comprises of core areas such as HRMS & PIP.HRMS (Human Resource Management System), which further includes areas such as Soft Joining, Promotion, Transfer, Confirmation, Leave Attendance and Exit, etc. PIP (Payroll Information Portal), wherein employees can view their individual Salary details, submit investment declaration, Reimbursement claim & CTC structuring, etc. Management of Large Chain of Hotels and ERP Database in AWS Cloud involves continuous monitoring with regards to the areas such as Performance of resource usages and optimization techniques relating to the use of PL/SQL. High Availability (HA) of data is accomplished through the Backup and Recovery mechanism and security of the data by Encryption & Decryption mechanism.

Broad Academic Area of Work: CLOUD COMPUTING

***Keywords*--** AWS Cloud, Database, ERP, Hotels, PL/SQL, S3 Bucket and Query

I. INTRODUCTION

Oracle Database works by utilizing the various components of the Hardware resources such as CPU, Memory (RAM), HDD (SAN-STORAGE), Networks, etc. The performance of the database depends upon the resource usages. Queries in a database is written to fetch or get the result, wherein all the resources are utilized.

If the query is a complex query then it will start consuming the CPU, the Memory and the HDD (Read/Write - I/O).In such scenarios Query Optimization or Query Tuning plays a vital role for enhancing the DB performance.

Most of the organizations prefer to use Oracle database on the Linux flavors world-wide, however, Oracle also provides database services for the Windows Operating System users as well.

Linux is more robust, and from the security point of view is more desirable as compared to the Windows O/S.

Oracle database and RAID works very closely. **RAID** is known as **Redundant Arrays of Inexpensive Disks**. This is used for Load Balancing and data protection across multiple disks. Most commonly used RAID levels, are RAID 0, RAID 1, RAID 0+1, RAID 1+0 or 10 and RAID 5.

RAID 1+0 (RAID 0 and RAID 1), is the best RAID option or Oracle. For Oracle database control Files, Online Redo Logs, Archived Redo Logs and Undo Datafiles, RAID 1+0 is preferred, as need to have multiple copies of the same. The Temporary datafiles, RAID 0 is preferred, as if the datafile is lost, the same can be recreated, as it has no data to restore.

Oracle database Datafiles preferred RAID levels are RAID 1+0, RAID 0+1 or RAID 5.RAID levels varies depending upon the price /commercials and performance.
The latest version of 19c provides the database in the form of CDB & PDB's on ASM.

Oracle's **ASM** (**Automatic Storage Management)** is the most widely used and preferred solution to provide data striping and mirroring. Here ASM disks are protected by the hardware RAID, and are defined using external redundancy and has no failure groups. In general two disk groups namely DATA and FRA are used. DATA stores all the database related files and FRA stores the fast recovery area, which includes multiplexed copies of online redo logs and control files.

One of the best feature called **LVM**, which stands for **Logical Volume Manager**, is a highly efficient used framework in the Linux distributions, owing to the facility of providing logical volume management for the Linux kernel. The basic function of LVM, is to provide allocation of disks, mirroring and stripping, and re-sizing the logical volumes.

Tuning can be done at the Memory level, and also at the SQL-PL/SQL level. Increasing the CPU will definitely increase the database server performance.

Also, keeping the datafiles on the SAN box is mandatory not only for performance (I/O), but also in case any damage happens to any of the disks due to block corruption, the other mirrored disks can be used , and hence, there will not be any data loss or any DB outage.

e-ISSN: 2250-0758 | p-ISSN: 2394-6962
Volume-10, Issue-5 (October 2020)
https://doi.org/10.31033/ijemr.10.5.11

RAID levels are used for the data redundancy, owing to the fact that the data is critical.

Oracle GoldenGate is used for the data migration services to replicate data from one database to the other database across different geographic regions world-wide.

Oracle GoldenGate allows to migrate the committed transactions in the database across multiple heterogeneous systems. Oracle GoldenGate comes as a separate software product, which needs installation & configurations.

Database Administrators core responsibility is to manage all the company's database pertaining to the very important and critical areas such as database performance which utilizes resources, database backup and recovery (HA), and security through the encryption and decryption mechanisms, as per the industry standards. The solutions implemented in the database focuses in the three core areas of performance, keeping in view the wide variety of clients world-wide. Before the implementation of any solutions on the production, On the fore-front of the implementation of solutions in the OCI (Oracle Cloud Infrastructure) in conjunction with the AWS cloud. For the migration of databases to a higher version which presents various enhanced database features. Another solution which needs to be implemented for the database is the migration of the current database from Non-Container database to **container database (CDB / PDB's)**, and also to utilize the **ATP (Autonomous Transaction Processing)** of **Oracle** database. **OCI (Oracle Cloud Infrastructure)** provides the **Real Application Clustering (RAC)** environments, wherein the automatic storage management is done by using ASM disks, wherein the High Availability (HA) is provided continuously, without any database outage (downtime).This is a very important feature of Oracle which helps the business to the maximum possible extent.

All the necessary steps are being taken for making sure that the core areas of the database remain within the threshold limit, so that the resource usage is not hampered, and which will automatically enable a better performance of the database in totality.

AWS RDS/EC2, provides solutions which have been successfully implemented curtailing to the company's business areas world-wide, and which has helped the company to establish itself as a fore-front in the Hospitality, ERP, PAYROLL SERVICES, DMS, BE-SPOKE industry.

II. LITERATURE REVIEW

Database Dictionary and Performance Views

In accordance with the database features, data dictionary plays a very important role in the Oracle database, as it stores the meta-data of the database. Meta-data pertains to the logical structure of tables, views, indexes, procedure, function, packages, package body, trigger, etc. A data dictionary can be referred to as centralized repository of information of data of the database, which gets updated automatically and which reflects the changes in the data and structure of the database, auditing of the DB, grants and privileges.

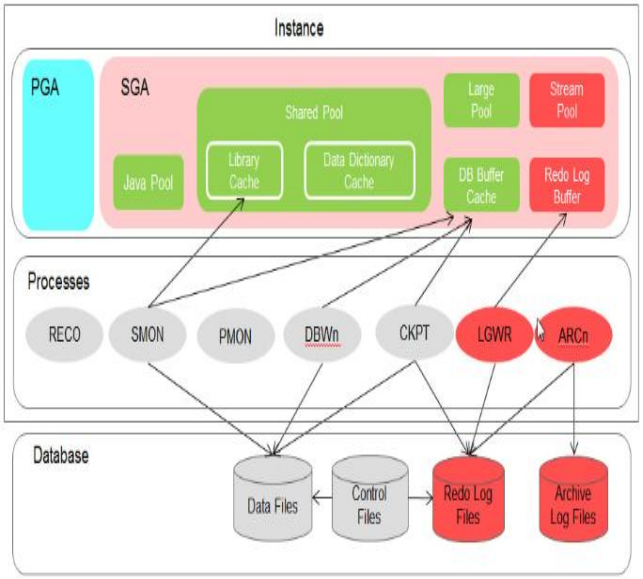

Figure 1: Oracle database structure

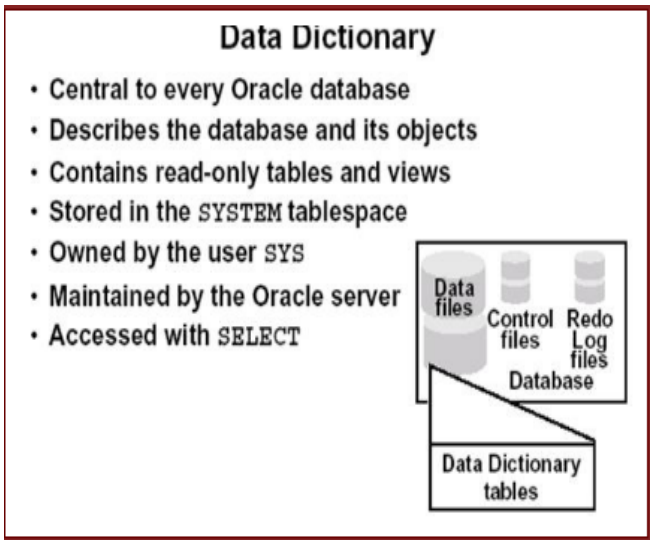

Figure 2: Data dictionary structure

There are basically two types of views which works when the DB is in OPEN mode. One is **V$views**, which are the dynamic performance tables, which are updated continuously, and presents to the users the data as of now. **A DBA view describes** all views in the database. Parameters in Oracle, plays a very pivotal role in the overall working of the database.

There are very important views which stores the Oracle database dictionary information. **V$DATABASE** view, shows the overall detailed information about the database. **V$SESSION** view, shows the overall sessions details in the database. The other data dictionary view is DBA_views, which shows the overall details about various objects like tables, indexes, SGA (System Global Area), the physical storage details (data).**DBA_DATA_FILES** view stored the database files along with the tablespace name and their physical location. **DBA_SEGMENTS**, specifies the total database size which is actual in use. This is very important view, as it presents the exact database size at any point of time. Through this the size of any schema and any tablespace can be known.

This paper covers the aspects of database performance, in AWS cloud & Oracle Cloud Infrastructure (OCI), which is an important factor in the database management area. The work which has been done to prepare this paper has helped to a large extent to enhanced database knowledge not only to the previous versions of database but also to the latest version of Oracle DB 19c, which has many newly added database features (CDB/PDB's)which will definitely help (DBA) and also the developers.

The implementations of all the features such as queries to monitor the database CPU consumption sessions, finding the sessions responsible for the Memory & I/O consumption has helped a lot to monitor the DB server performance with great ease, and which has resulted in the database server performance and health not getting hampered under any circumstances. Details of each and every components of database aspects pertaining to the overall performance of the DB server, as performance is the most vital part in the database functioning.

III. IMPLEMENTATIONS

CPU, MEMORY & I/O usage monitoring plays a very important role pertaining to constantly monitoring the database performance which should always be within the defined threshold limit, so that the performance of the database server is not compromised or hampered under any circumstances. There are **events** which are related to the **CPU**, **MEMORY**&**I/O** such as, **cursor: pin S, db file scattered read, db file sequential read, enq: TX - row lock contention, latch: row cache objects, buffer busy waits, enq: RO - fast object reuse, direct path read, library cache: mutex X**, etc. These database events are optimized through the use of various PLSQL techniques, and which forms the part of the database performance tuning. Every newer versions of the Oracle database comes with enhanced database performance techniques, and by applying which the DB server performance is increased substantially, which directly increases the application performance.

The database events which are responsible for the CPU consumption, hampers the database server performance which leads to slowness. There needs to be continuous monitoring of CPU events through the use of tools like Oracle SQL Developer and TOAD, which have got much in-built functionality to monitor the health of the database server. **resmgr: cpu quantum** is the main event which needs to be resolved by the process of query tuning to have the CPU usage within the threshold limit and thereby enforcing the mechanism that the DB server performance is not at all hampered. Also up-scaling the CPU resources during peak hours, also reduces the CPU contention on the database server.

Memory tuning in Oracle database is the key factor for the performance of the database server to the larger extent. **MEMORY_MAX_TARGET** and **MEMORY_TARGET**, are the two **MemoryTuning** parameters in **Oracle 12c** and higher versions of Oracle.

MEMORY_MAX_TARGET is not a dynamic parameter whereas MEMORY_TARGET is. It means that out of the available total physical memory, the MEMORY_MAX_TARGET is the, maximum database memory which has been assigned from the physical server memory, and which requires reboot (database bounce) to take effect of the same in the database, whereas MEMORY_TARGET is the database memory which can be assigned to a value equal to that of the MEMORY_MAX_TARGET

If MEMORY_TARGET, is defined or set, then Oracle will perform AMM (automatic memory management) automatically, and both SGA (System Global Are) & PGA (Program Global Area) are managed within the allocated memory. There is no need to set SGA_TARGET &s SGA_MAX_SIZE parameters.

I/O monitoring is also very important as far as the database server performance is concerned and managing the I/O, which refers to the Physical reads from and Physical writes to the SAN storage disks. Also many times there are scenarios, wherein the same query is being used again and again, and then in that case the Oracle will read the data from Memory (Buffer), and which is known as Soft Parsing, rather than reading from the disks (Hard Parsing) which directly increase the query performance.

Oracle has inbuilt mathematical functions and other query clauses which provides the output as desired.

When there is increase in the block changes, then the Oracle database server performance is hampered.

The increase in the block changes means there is too much of physical writes directly due to INSERT SQL statements, during this point DBA has to take the appropriate action and see that the block changes in the

e-ISSN: 2250-0758 | p-ISSN: 2394-6962
Volume-10, Issue-5 (October 2020)
https://doi.org/10.31033/ijemr.10.5.11

database is within the threshold limit depending, so that the database server health is not compromised.

The crux is to get the actual physical block changes which happens in the database and which results into the utilization of the available read/write (IOPS).

Through block changes (due to ACTIVE sessions) in the database, it can be obtained which sessions are actually consuming the maximum memory and physical storages which happens only due to **DML's** (**INSERT/UPDATE/DELETE/MERGE**).

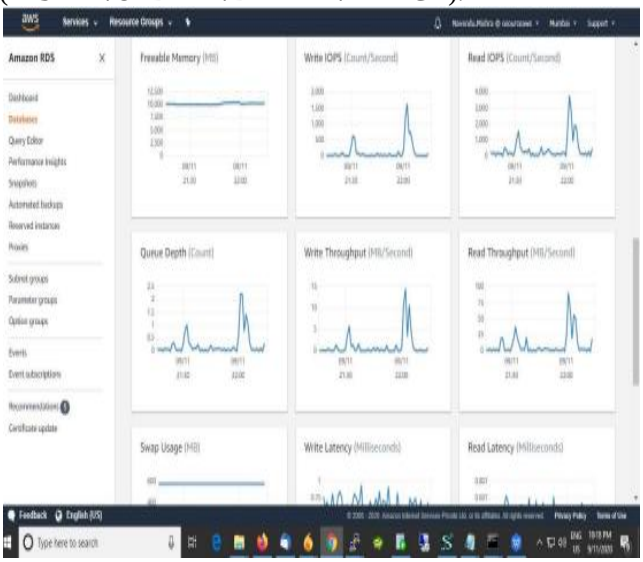

Figure 3: CPU, Memory, I/O and Storage

Fragmentation

Fragmentation in the database occurs mostly due to DELETE, but also involves INSERT/UPDATE/MERGE. Fragmentation degrades the overall database performance, and hence steps needs to be taken care to remove the fragmentation completely on a timely basis. By removing the fragmentation, the database performance in terms of query retrieval is increased and also the un-used storage is retrieved and which adds up to the total free space in the database. As the schema size grows, fragmentation increases, so it's mandatory that the task of schedule of fragmentation removal from a schema needs to be carried out twice a year, so that only the used space is consumed by the Oracle server, and the unused space which the Oracle server has, can be reclaimed. DBA should always see that the schema or tablespace fragmentation is very minimal so that the DB performance is maximum.

Index Rebuild

Index rebuild is a continuous on-going activity which a DBA has to perform in a continuous and timely phased manner, so that the query retrieval is faster, and the database performance is increased. Index rebuilds has been a successfully implemented methodology in the database technology for the enhanced output of the query, which has millions of records. Index rebuild is not applicable for the BLOB's (Binary Lob's) like PDF, Image, Videos files, etc. During the index rebuild activity, it's very important to make sure that the schema for which the index is being rebuild, no sessions of that schema should be there. Also index rebuild generates archive logs files because Oracle drops and recreates the index which creates the online redo log called Archive logs, and, which stores the actual database changes.

Statistics Gather

Database statistics gathering is an activity which needs to be performed by the DBA, and which depends upon the database objects which gets accessed. Statistics gather for all those objects which have become stale and is mandatory to be carried out, so that the data dictionary gets updated, and the updated database details can be retrieved at any point of time. Statistics gathering is also a very important factor in the enhancement of the database performance overall. Statistics gather is also an on-going activity and which needs to be carried out in a timely phased manner, which needs to be carried out in the scheduled DB maintenance time only.

Query Tuning by Using SQL & PL/SQL

SQL is used for faster data retrieval, data manipulation, transaction control. There are various statements which relates to SQL such as SELECT, INSERT, UPDATE, DELETE, MERGE, ALTER, DROP, COMMIT, TRUNCATE, etc. Many advanced SQL features have been incorporated in the latest version of Oracle, and due to this the capability of SQL has increased a lot.PL/SQL is an extension of SQL with features incorporating to database programming language, which is being used widely. Integration, Improved performance, program development in modules, code clarity and portability are some of the advantages of PL/SQL. By using PL/SQL database objects like Procedure, function, package, package body, trigger, table, views, etc., can be created easily and which can be used in the application. Query tuning plays a very important role in the overall performance of the database.

Below are the Query tuning rules which needs to be followed in PL/SQL for enhancing the performance of the database.

- *Use **ANSI** Joins Instead of Oracle Joins.*
- *Avoid **WHERE** Clauses with Functions where ever it is defined.*
- *Use of **CASE** Instead of Multiple Unions for multiple statements in a Query.*
- *Minimize the Use of **DISTINCT** clause in every statement for duplicity of Data.*
- *Use of **UNION ALL** instead of **UNION** in every statement of result data to be adopted.*

e-ISSN: 2250-0758 | p-ISSN: 2394-6962
Volume-10, Issue-5 (October 2020)
https://doi.org/10.31033/ijemr.10.5.11

- *If possible used Global Temporary Table in place of simple table Creation.*
- *Always Specify Columns Name in **INSERT** Statements.*
- *Avoid Object Names with "Spaces".*
- *Index all the predicates in "**WHERE**", "**ORDER BY**" and "**GROUP BY**" clauses.*
- *Avoid using wild-card (%) at the beginning of a predicate.*
- *Avoid unnecessary columns in **SELECT** clause of statement.*
- *Use "**inner join**", instead of "**outer join**" if possible in statements.*
- *Tables should always have a Primary Key defined in columns.*
- *Do not perform **DML** statement in Function.*
- *"**As**" keyword is used instead of "**IS**" in standalone Procedure.*
- *For error discovery the below command is used show errors procedure < procedure name>.*
- *Alternatively, you can type, SHO ERR (short for **SHOW ERRORS**) to see the most recent compilation error.*
- *"**Hard Coded**" Values is not be written at procedural Level. Avoid using it in the Code level of statements.*
- *Avoid dynamic creation table at the procedural level of statements.*
- *Use of "**Exist**" and "**Not Exists**" in place of **In** and **Not In** operator for select statements.*
- *Avoid "**Having Clause**" in select statements.*

Top 10 database events which are used to define the overall load on the server, which is shown in the below graph.

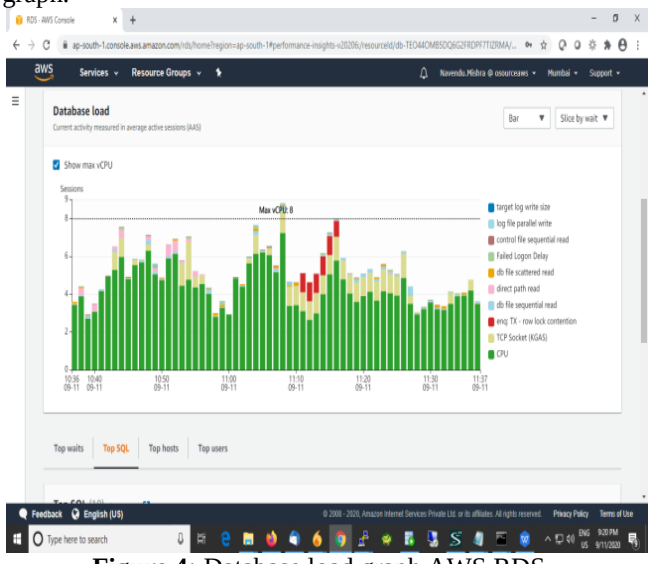

Figure 4: Database load graph AWS RDS

IV. CONCLUSIONS AND RECOMMENDATIONS

This paper was immensely helpful in implementing the various features with regards to the management of the **large chain of hotels** & **ERP Database world-wide**. The widely regarded and mandatory aspects of the database management in AWS cloud pertains to the performance of the DB server. Owing to the growing demand of the database size and taking into account the core areas of database it's very much required to migrate to **19c**.By migrating to **19c** the company will get the benefit of **Oracle's ATP (Autonomous Transaction Processing)** feature including **Container databases (CDB/PDB's)** and which provides the HA (High availability). This is highly appreciated and used globally. There is vast features in **Oracle 19c,** and by utilizing which will increase the overall performance, backup and recovery & security of the database at large.

As far as recommendations is concerned, we would like to take this opportunity to implement new features through the use of latest version of **Oracle 19c in AWS cloud / OCI (Oracle Cloud Infrastructure)**, which also supports **Autonomous Transaction Processing (ATP)** feature which reduces the manual intervention of the DBA. Also would like to implement other available feature in **Oracle AWS cloud / OCI Container database (CDB) and pluggable databases (PDB's)**.

Oracle database 19c has new features as compared to **Oracle 12c**, and depending upon the current industry trend it's the right time to migrate to **Oracle Cloud Infrastructure (OCI)**, which is a great player in the database cloud environment management.

REFERENCES

[1] Badger, L., Grance, T., Robert, P., & Voas, J. (2012). Computer security. DRAFT cloud computing synopsis and recommendations. *NIST Special Publication*, pp. 800.

[2] Barbara, J., Jo, A., Cynthia, S., & Adam, T. (2009). Collaboration using cloud computing and traditional systems. *Western Carolina University*, *10*(2), 27-32.

[3] Dykstra, J. & Alan, T. (2012). Acquiring evidence form infrastructure-as-a-service cloud computing: Exploring the evaluation tools, trust and techniques. *Digital Investigation*, S90-S98.

[4] Hofer, C. & Karagiannis, G. (2011). *Cloud computing services: Taxonomy and comparison*. Available at: http://link.springer.com/article/10.1007%2Fs13174-011-0027-x.

[5] Lazewski, V., Javier, D., Fugang, W., & Geoffrey, C. (2012). *Comparison of multiple cloud frame works*. Available at:

e-ISSN: 2250-0758 | p-ISSN: 2394-6962
Volume-10, Issue-5 (October 2020)
https://doi.org/10.31033/ijemr.10.5.11

http://www.academia.edu/2929921/Qualitative_Comparison_of_Multiple_Cloud_Frameworks.
[6] R. Buyya, C. Vecchiola, & S. T. Selvi. (2013). *Mastering cloud computing: Foundations and applications programming*. Elsevier.
[7] P. K. Paul, K. Kumar, & D. Chaterjee. (2014). Cloud computing emphasizing emerging possibilities to the entire information infrastructure. *Trends in Information Management (TRIM), 9*(2).
[8] *Worldwide cloud computing market shares, strategies, and forecasts, 2009-2015*. (2014 Dec). Available at: http://www.researchandmarkets.com/reports/1057978/worldwide_cloud_computing_market_shares.
[9] P. Mell & T. Grance. (2011). *The NIST definition of Cloud computing*. NIST Special Publication.
[10] J. Harris & A. Alter. (2010). Cloud rise: Rewards and risks at the dawn of cloud computing. *Accenture Institute for High Performance Research Report*, pp. 5-30.