STeP-IN SUMMIT 2007

International Conference On Software Testing

Testing an Application to identify Oracle Database bottlenecks

By

S. Ramkumar

ArisGlobal Software Pvt. Ltd.

Practices Experience Knowledge Automation

Produced By

STeP-IN Forum

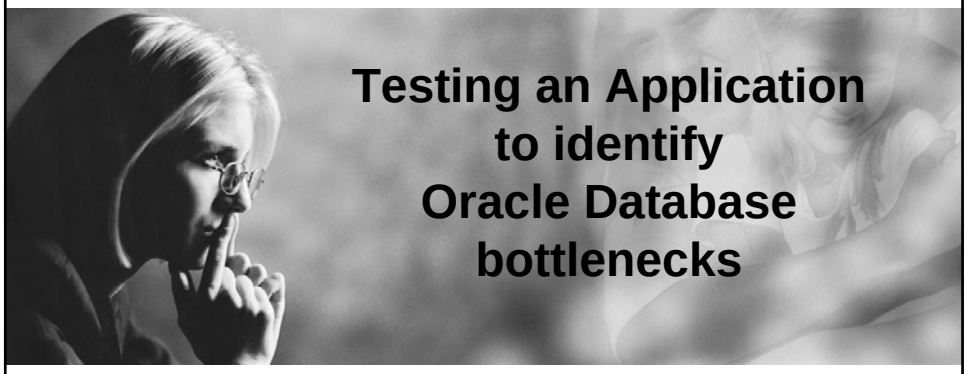
www.stepinforum.org

Hosted By

QSIT The Knowledge Corporation

www.qsitglobal.com

@ris global



Testing an Application to identify Oracle Database bottlenecks

S. Ramkumar
Database Architect
ArisGlobal Software Pvt. Ltd.

The power to protect
Your products
& Your patients

Agenda

- As Usual – lets get back to BASICS
  - DESIGN
  - Use the RIGHT DATATYPE
- Better alternatives than TABLEs
- Is your Application SCALABLE
- Do you have LEAKS in your Database actions
- Post Development IMPROVEMENT
- When doing a PERFORMANCE TEST …..
- TOOLS for you

@ris global

2

Lets get back to BASICS

- Understand the Data Requirement – OLTP or DWH
- Use the right model – E/R Vs Dimensional Modelling
- Put Constraints in the database
- Use the most appropriate data types

@ris global

3

Use the Right datatype

- Use the most appropriate data types
  - Its obvious but just go back and check your applications
  - Date for Date data
  - Number for Numeric data
  - Size for Character Columns

How would you transport them

You can choose !!!

@ris global

4

Use the Right datatype

- How many of you can recall what these errors are?
  - ORA-01722:
  - ORA-01858:

@ris global

5

Use the Right datatype

- You probably have one of those systems if you know these error codes!
  - ORA-01722: "invalid number"
  - ORA-01858: "a non-numeric character was found where a numeric was expected"

@ris global

6

If you don't Get back to Basics

- All you achieve is:
  - You lose the edit upon insertion to the database (*garbage in, garbage out*)
  - You *increase* your storage needs
  - You *decrease* data integrity
  - You *lose performance*
  - You *confuse* the optimizer

@ris global

7

Better Alternatives than TABLEs

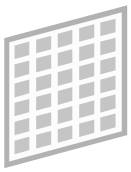
- Index Organized Tables
- Materialized Views
- Clusters
- Table Partitioning

@ris global

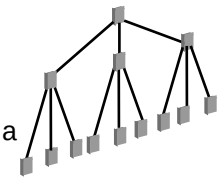
8

Tables and Indexes

- TABLE
  - Stores data as given
  - No order of storage
  - No optimization for retrieval
  - Stores all rows in the order of columns



- INDEX
  - A catalyst to fetch data from tables
  - Useful for columns in 'WHERE' clause
  - Different types available
  - Contains the column values that is indexed and a pointer(ROWID) to that row in the table

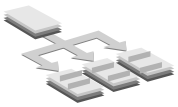


@ris global

9

Index Organized Tables

- This is a specialized storage structure
- Mandatory to have a Primary Key
- Rows are ordered as per Primary Key and complete row is stored in the form of Index
- Benefits
  - No need to access the index and table
  - Useful for lookup tables
  - Useful for tables queried on primary key only
  - Results in tremendous improvements when this table is accessed very frequently
  - Decrease need for indexes

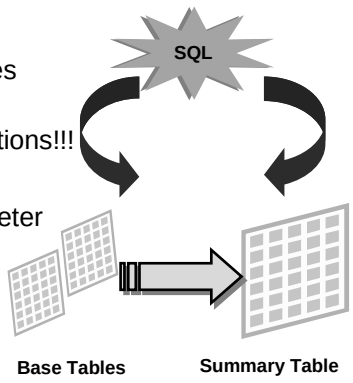


@ris global

10

Materialized Views

- Helpful when
  - Large tables need to be queried
  - Not many changes occur to these tables
  - Queries look for aggregate value
  - You don't need to change your applications!!!
- Need to use
  - QUERY_REWRITE_ENABLED parameter
  - ENABLE QUERY REWRITE in MV
- Benefits
  - The amount of data to be looked is very less and hence improvement
- Also use DIMENSIONS in case of Star Schema



@ris global

11

Clusters

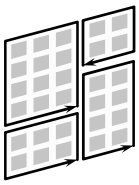
- Generally one table has a separate storage space
- In a Cluster, Multiple tables use the same storage Space
- Useful for Lookup Tables that are related
- Data is grouped and kept together for multiple tables
- Not useful if the tables are volatile

@ris global

12

Partitioned Tables

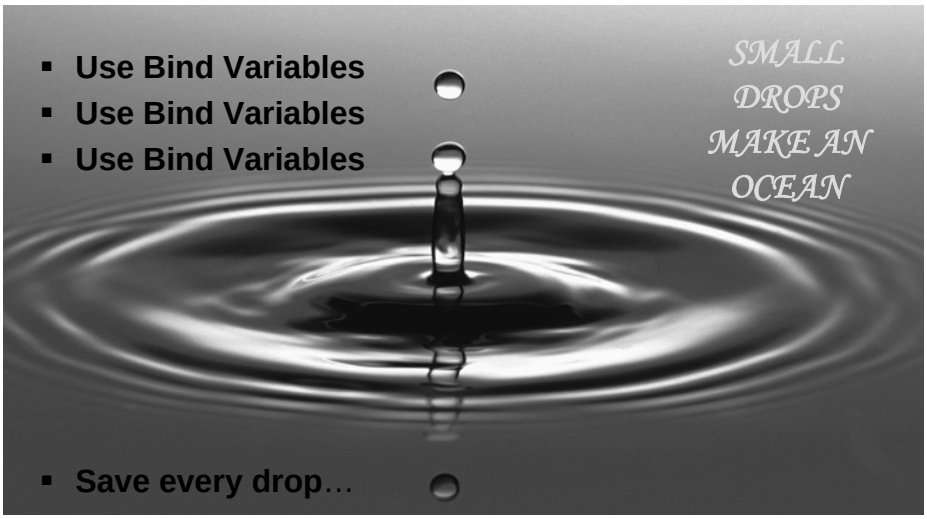
- One table is split to occupy multiple storage segments
- The database is aware which data is in which partition
- Benefit of not going through the entire table to fetch particular data
- Helpful when most of the queries restrict based on the partition key
- Useful for Large Tables
- Multiple options available
  - Hash
  - Range
  - Composite
  - List



@ris global

13

Is your Application SCALABLE



@ris global

14

Example of Using Bind Variables

```
SCOTT@utf> set timing on
SCOTT@utf> declare
  2         type rc is ref cursor;
  3         l_rc rc;
  4         l_dummy all_objects.object_name%type;
  5         begin
  6         for i in 1 .. 1000
  7         loop
  8            open l_rc for  'select object_name from all_objects where object_id = :x'  using i;
  9           fetch l_rc into l_dummy;
 10            close l_rc;
 11        end loop;
 12     end;
 13  /

PL/SQL procedure successfully completed.

Elapsed: 00:00:01.00
```

@ris global

15

Example of not Using Bind Variables

```
SCOTT@utf> set timing on
SCOTT@utf>  declare
  2         type rc is ref cursor;
  3         l_rc rc;
  4         l_dummy all_objects.object_name%type;
  5      begin
  6         for i in 1 .. 100
  7         loop
  8            open l_rc for  'select object_name  from all_objects  where object_id = ' || i;
  9            fetch l_rc into l_dummy;
 10             close l_rc;
 11         end loop;
 12     end;
 13  /

PL/SQL procedure successfully completed.

Elapsed: 00:00:06.01
```

@ris global

16

So what's the problem?

When using Bind Variables Oracle parsed

select object_name from all_objects where object_id = :x

Once and reused it the remaining 999 times

When not using Bind Variables Oracle parsed

select object_name from all_objects where object_id = 1

select object_name from all_objects where object_id = 2

select object_name from all_objects where object_id = 3

…. 100 times

@ris global

17

Where to Use Bind Variables

- Not required for PL/SQL variables
- Required in PL/SQL if Dynamic SQL is used
  - Execute Immediate
  - DBMS_SQL
  - Ref Cursors
- Required in Java or other Languages
  - Use Prepared Statements
  - The developer should know about it

@ris global

18

Does your application Leak!!!

- Sessions not being closed
  - ORA-00020 maximum number of processes (*string*) exceeded
- Cursors not being closed
  - ORA-01000 maximum open cursors exceeded

- These result in Extra Memory usage by Oracle which results in the memory leak

@ris global

19

Example of Application Leaks

- In the n-tier architecture Connection Pool is a common phenomenon

If you don't :

- Return the Connection to the Pool
- Close connections when you have finished using
- Close Cursors after using them
- Close Cursors in Exceptions Block

@ris global

20

How to identify Leaks

- Connection not closed

```
SELECT count(*),username,program,machine
FROM v$session
GROUP BY username,program,machine
```

- Cursors not getting closed

```
SELECT count(*),sid,sql_text,user_name,sid
FROM v$open_cursor
GROUP BY sql_text,hash_value,user_name,sid
ORDER BY count(*) desc
```

@ris global 21

Post Development Improvements

- Create new indexes
  - As the application is tested some queries may take time
  - Identify the queries and add indexes
  - 90% of the time an index solves the problem
  - Remaining times you may need to change the code
- Use Cost Based Optimizer
  - When there is no Statistics Oracle use Rule Based Optimizer
  - Need to collect Statistics to use Cost Based Optimizer(CBO)
  - Infact this has to be used in Development also!!!

@ris global 22

Post Development Improvements

- Use the right Oracle Parameters
  - For OLTP applications
    - Optimizer_index_cost_adj = 25
    - Optimizer_index_caching = 75
  - For DWH applications
    - Optimizer_index_cost_adj = 100
    - Optimizer_index_caching = 0
  - Other Parameters
    - Allocate Proper memory to the Instance – SGA
    - Use Shared Server if there will be a lot of sessions on the Database
    - Consider Parallelism if yours is a DWH application

@ris global

23

When doing a Performance Test….

Have realistic / production

- Data Volume
- User Load
- Use Cases/ Test cases
- Test Case Load Distribution
- Bandwidth / Latency

@ris global

24

Tools and Utilities – Most trusted Friends

- SQL*Plus Autotrace
- Explain Plan
- SQL_TRACE and TKPROF
- Statspack
- Oracle Documentation
- http://asktom.oracle.com
- www.oracleplsqlprogramming.com
- Effective Oracle by Design – Thomas Kyte
- Cost Based Oracle Fundamentals – Jonathan Lewis
- Expert Oracle Database Architecture – Thomas Kyte

@ris global 25

"Question Authority"

- Make the experts prove everything
- Everything can (and should) be proven
- Things change, expect that
- Statements that should raise your eyebrows:
  - It is my opinion...
  - I claim...
  - I think...
  - I feel…

@ris global 26

What we did

- Got back to BASICS
  - DESIGN
  - RIGHT DATATYPE
- Got introduced to better alternatives than TABLEs
- How to make your application SCALABLE
- How to check if you have LEAKS in your Database
- What can be done to improve Post Development
- How to ensure an effective PERFORMANCE TEST …..
- New Friends for you

@ris global

27

ramkumar.swaminathan@arisglobal.co.in

Questions and Answers

@ris global

28