



Open access Journal

International Journal of Emerging Trends in Science and Technology

IC Value: 76.89 (Index Copernicus) Impact Factor: 4.219 DOI: <https://dx.doi.org/10.18535/ijetst/v4i8.60>

Performance Tuning: An Overview

Author

Ankit Gupta¹

Haryana India

Abstract

With the growing era of digitization, penetration of Information Communication Technology in every field has made the human life more comfortable. Every services, thing is on just at the click of the human being. To provide the seamless Service to their client, people every organization is updating their infrastructure, Tune their software to meet the current environment demand. Performance testing plays a major role in this phase of digitization.

Keywords-Performance Tuning, Bandwidth, SQL, Caching, ICT

1. Introduction

In the fastest growing digitization era, on an average every service is going on web. The purpose of Digitization, automation is the availability of information, services on web. The idea behind is to provide the people a clearcut transparency, secure and hassle free service. Large and small, at very scale, every organization is moving toward Automation. People need service on their click. So as more and more people are using e service, the load on Application increase, which decreases the performance of the system. As the load increase the response time also increase which make the user experience as worst. So performance tuning becomes a major part in software development. In Performance tuning scalability is very critical factor. it means as the load on system increases the response time of system should not increase with respect to load. So to get their response time with in their limit, developers keep does their system performance testing and with their result they tune their system as per their requirement.

This whole paper is divided into 3 sections. The first part contain introduction second part contain the Procedure of Performance tuning and the last and finally we conclude the paper.

2. Performance Tuning Procedure

Performance tuning can be done by following a proper approach in which first phase is to identify the problem mean the area where problem is occur ,

bottleneck of the application. in second step try to remove that bottle neck part by various method as shown in figure no 2 and finally in last step regain do performance testing to check whether the step you taken to tune your system go in right direction or not. if not then repeat the step no 2. The figure no 1 shows the whole step by step procedure followed in performance tuning.

2.1 Code Optimization:-In this method we change the code of software so that it takes less times. There are different

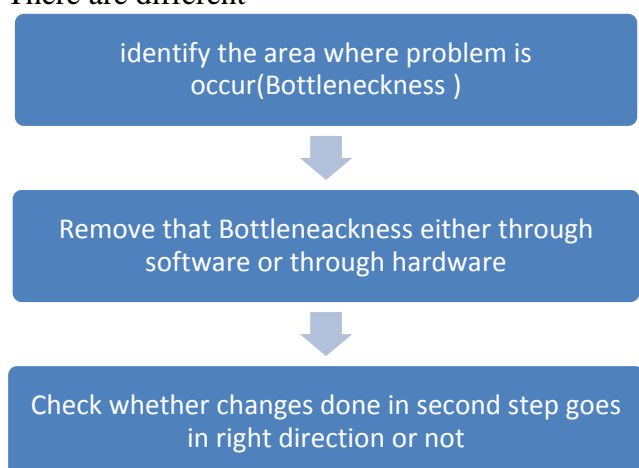


Figure 1: Procedure followed in performance Tuning

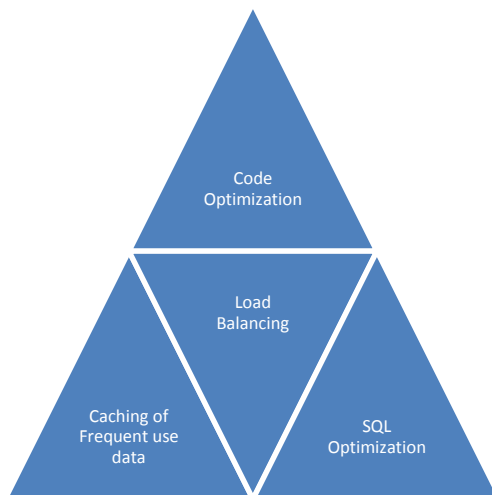


Figure 2: Methods to Tune the Application

ways through which code optimization can be done like removing of dead element, Minimize the use of variables, avoid sum math function like Multiply, division and instead of that try to use Add and Sub, Use of macro etc.

2.2 Load balancing: - In this strategy, we try to distribute the load on different server so that computation efficiency of one server should not cross beyond the limit. In this strategy we put the different web server, application server and distribute the request coming through outside on them with the help of load balancers.

2.3 Caching of Frequent Request Data[2]: - In this, we will cache the data which is frequently demand by the user based upon the previous statics, so that instead of request goes to database it will directly serve it to the requester and provide him data.

2.4 SQL Query optimization:- when programmer writes the query for any application to access the database of its application, he or she somehow intentionally use such a syntax for query which will takes longer times in execution. so writing the query in optimization way is very important part while tuning your application to get better response time. Many steps can optimize SQL Query:-

2.4.1. Proper indexing has to be done:-In this all the tables that are present in your database has to Indexed with respect to primary key and foreign key of that table [1].

Suppose we have table name as Student shown in Table 1. We have to find out name of all students whose marks lie between 10 and 100.

Select Name from Student where $10 < \text{Marks} < 100$.

So if we have properly applied indexing in sorted order on marks of students query run fast and gives us result. As in sorted index no need to check condition one by one for each record. If indexing is not applied then all the records are to scanned one by one with checking the where condition one by one as in table 2

Name	Marks (Indexing on sorted basis)
Ankit	1
Arun	7
Ashish	12
Vipin	14
Vikas	23
Yogesh	33
Amit	43
Kamal	56
Sunil	65
Amrinder	72
Himanshu	82

Table 1 When proper indexing done on Marks in sorted order

Name	Marks (Indexing)
Amit	43
Arun	7
Himanshu	82
Vipin	14
Vikas	23
Yogesh	33
Ankit	1
Kamal	56
Sunil	65
Amrinder	72
Ashish	12

Table 2. When indexing is not done on Marks

2.4.2 Use of only required Attributes in Query:-

While writing SQL query, use only required column values. Do not write * as this will select all the column and query will run slow [3]. Suppose in Table no 3 Database if have to just show the name, class of student then write query as
Select Name, Class

From Student
Where Marks>15
Instead of
Select *
From Student
Where Marks>15

Name	Marks	Class	Color	Grade
Amit	43	1	Red	A
Arun	7	2	Green	O
Himanshu	82	10	Blue	B
Vipin	14	3	Yellow	C
Vikas	23	5	Violet	O
Yogesh	33	6	Red	A+
Ankit	1	1	Brown	B-
Kamal	56	2	Orange	A

Table no 3.Full Detail

As * will pick all column so a lots of data has it become which will slow down or take more time .I am showing it with respect to a small table but in application there are lacks of records and hundreds of column. The time taken by * operation will increase in exponential way.

2.4.3. Minimize the use of correlated Sub Query:-

Correlated Sub query are those having query inside the another query. The output of outer query will be used by inner query for processing as input [4]. Correlated query can be optimized by putting the inner query put as view and then perform joined operation on them [5].

2.4.4 Avoiding Loops in query: - If possible then try to avoid using loops . Instead of using loops administrator can use insert and update command to execute the database, as it will take less time [6].

3. Conclusions

Performance Tuning is a very important task in software field. The ICT success depends upon the response time of software. As the load on software increase, the response time should not increase. In this aspect, tuning becomes a major aspect. Although many defined, strategies come for performance tuning but no one guarantee it completely.

References

1. N. Bruno, S. Chaudhuri and R. Ramamurthy, "Power Hints for Query Optimization", IEEE

International Conference on Data Engineering,2009.

2. X. Lin, "Query Optimization Strategies and Implementation Based on Distributed Database", ComputerScience and Information Technology, 2nd IEEE International conference, (2009).
3. N. Bruno, S. Chaudhuri and R. Ramamurthy, "Power Hints for Query Optimization", IEEE International Conference on Data Engineering, (2009).
4. <http://www.zentut.com/sql-tutorial/understanding-correlated-subquery/>.
5. https://en.wikipedia.org/wiki/Correlated_subquery
6. <https://www.toptal.com/sql/sql-database-tuning-for-developers>