CSCI317 Database Performance Tuning

Principles of Database Performance Tuning

Dr Janusz R. Getta

School of Computing and Information Technology -
University of Wollongong

Principles of Database Performance Tuning

Outline

- Database performance
- Performance tuning
- Principles
- What do we try to optimize ?
- What do we try to optimize first ?
- Performance tuning techniques

Database performance

Database performance is the rate at which the DBMS handles the demand for information

The factors that influence database performance:

(1) Workload

- Workload is a combination of online transactions, batch jobs, ad hoc queries, business intelligence queries and analysis, utilities and system commands directed through DBMS at any given time
- Workload defines the demand; it can fluctuate from day to day, hour to hour, minute to minute; sometimes it can be predicted; at the other times it is unpredictable

(2) Throughput

- Throughput defines the overall capability of the system to process data
- It is a composite of I/O speed, CPU speed, parallel capabilities of a computer, and efficiency of the operation system and system software.

Database performance

(3) Resources

- Resources are the hardware and software tools like memory, disk space, cache controllers, microcode, etc. available to the system

(4) Optimization

- Query optimization is included in DBMS such that up-to- date and accurate database statistics for the query optimizer allow for achieving near optimal implementation of SQL queries
- There are other factors that can be optimized like database structures, system parameters, etc.Some optimization aspects are outside the scope and control of relational optimizer, e.g. efficient script coding, proper application design, etc.

(5) Contention

- Contention is a situation in which two or more components of a workload are attempting to use a single resource in a conflicting way, e.g. updates of the same data item
- DBMS uses locking mechanisms to enable multiple and concurrent users to access and to modify data in a database; using locks guarantee the integrity of data

Principles of Database Performance Tuning

Outline

Database performance

Performance tuning

Principles

What do we try to optimize ?

What do we try to optimize first ?

Performance tuning techniques

Performance tunning

Database performance tuning is an "activity" of making database applications run "more quickly" "Activity" means an act of making a modification to hardware or software system properties

"More quickly" means:

- higher throughput,
- shorter response time,
- larger number of users serviced in the same period of time,
- larger database loads handled in the same period time,
- faster backup and recovery

Principles of Database Performance Tuning

Outline

Database performance

Performance tuning

Principles

What do we try to optimize ?

What do we try to optimize first ?

Performance tuning techniques

Principles

(1) Think globally; fix locally

(2) Partitioning breaks bottlenecks

(3) Start-up costs are high; running costs are low

(4) Render unto server what is due unto server

(5) Be prepared for trade-offs

(1) Think globally; fix locally

Global identification of the problem

- A common approach to performance tuning is to look at processor utilization, input/output (I/O) activities, paging, network traffic, etc.

Local and minimal intervention

- A high level of processor utilization does not mean that we have to buy a new machine, high I/O activity does not mean that we have to buy more persistent storage devices, etc.
- A high level of processor utilization may mean that we have to improve a badly written application, high I/O activity may mean that we have to increase the size of a buffer

(2) Partitioning breaks bottlenecks

"Bottleneck" is a component of the system that limits the overall performance

Partitioning is a technique of reducing the load by dividing it over more resources or by spreading it over time

For example, partitioning of a large relational table and related index into smaller relational tables and smaller indices, partitioning long transactions into sequences of smaller transactions, partitioning a large `SELECT` statement into a combination of smaller `SELECT` statements

(3) Start-up costs are high; running costs are low

It is relatively more expensive to initiate an operation than repeat it for a longer time

For example, sending 1 Kbyte across a network is only a bit more expensive then sending 1 byte packet or reading one data block takes more or less the same time as reading entire group of adjacent data blocks

Achieve the objective with the fewest possible start-ups

For example, keep connections to a database server, keep query execution plans, keep frequently used data blocks in transient memory, etc.

(4) Render unto server what is due unto server

A typical dilemma is who suppose to do a job: client or server ?

The allocation of work between the database system (the server) and the application program (the client) may have a decisive impact on performance

For example, evaluation of consistency constraints that do not need access to a database should be performed at the client side, larger tasks performed on the server side reduce amount of data transmitted over a network

(5) Be prepared for trade-offs

Increased performance requires an appropriate combination of transient and persistent memory and computational resources,

For example, more transient memory invested in data buffer caches reduces the total number of transmissions from persistent memory and leaves less memory for storing query execution plans and data dictionary caching

Indexing speeds up query processing and delays updates, deletions, and insertions

Partitioning speeds up processing of one class of queries and delays the others

Principles of Database Performance Tuning

Outline

Database performance

Performance tuning

Principles

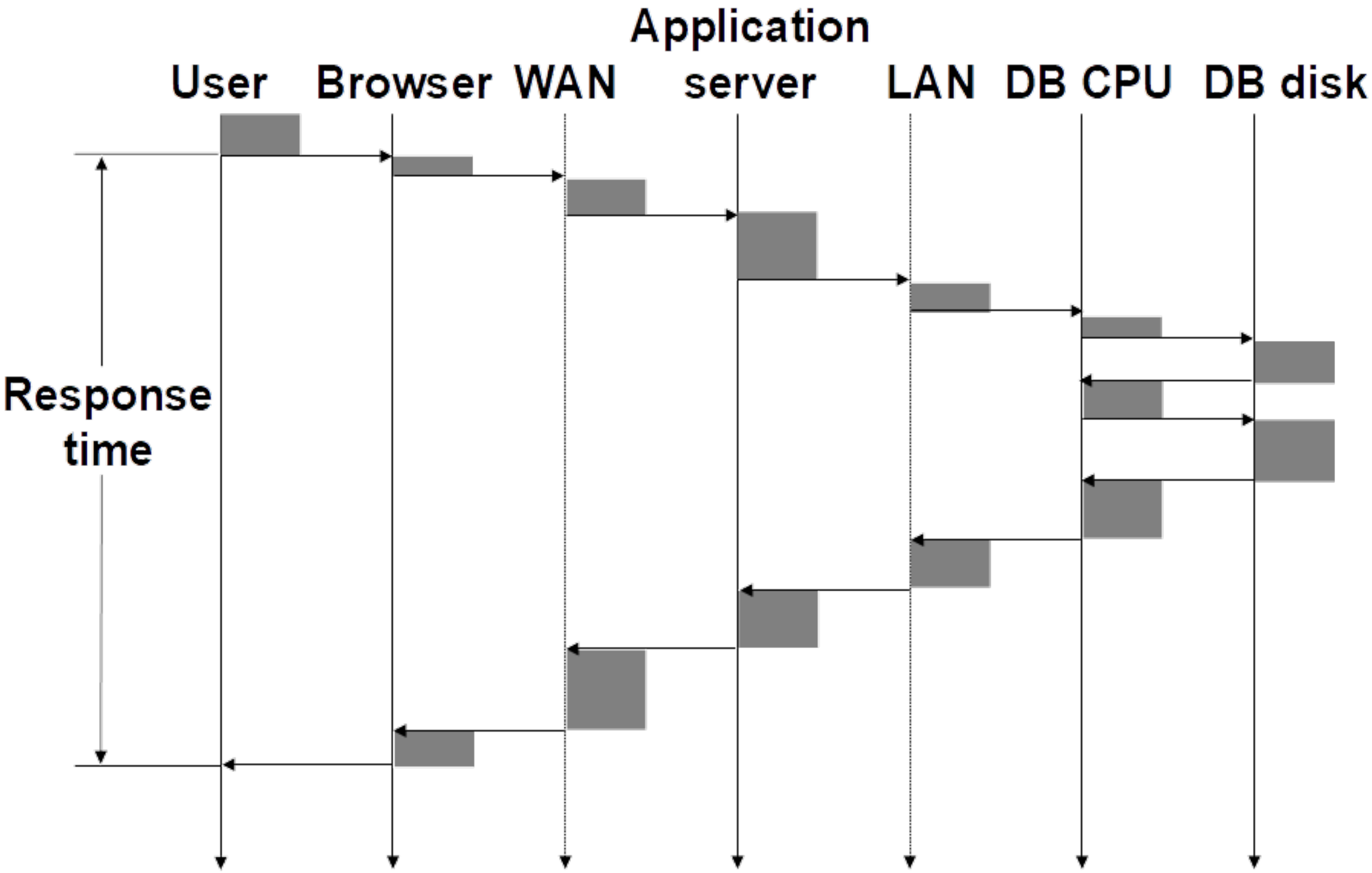
What do we try to optimize ?

What do we try to optimize first ?

Performance tuning techniques

What do we try to optimize ?

Response time



Principles of Database Performance Tuning

Outline

Database performance

Performance tuning

Principles

What do we try to optimize ?

What do we try to optimize first ?

Performance tuning techniques

What do we try to optimize first ?

Amdahl's Law [Gene Amdahl, 1967]

The performance enhancement possible with a given improvement is limited by the fraction of the execution time that the improved feature is used

A performance improvement is proportional to how much database application uses the thing you improved

Go for the "biggest bang" first !

Principles of Database Performance Tuning

Outline

Database performance

Performance tuning

Principles

What do we try to optimize ?

What do we try to optimize first ?

Performance tuning techniques

Performance tuning techniques

(1) Data modelling techniques:

- denormalization, partitioning, implementation of generalization, multimodel designs

(2) Persistent storage techniques:

- storage allocation, defragmentation, indexing, partitioning, clustering, materializations

(3) SQL programming techniques:

- optimization of query processing plans , implementation of relational algebra operations, "��hints"��, stability of query processing plans

(4) Application of advanced/specialized SQL:

- Multitable `INSERT` statement, `MERGE` statement, `CONNECT BY` clause, `WITH` statement

(5) SQL + host language + application language programming techniques:

- optimization of application processing plans , optimization of data transmission

Performance tuning techniques

(6) Database server tuning techniques:

- tuning data buffer caches , tuning logs archives backups, tuning system initialization parameters

(7) Concurrency control techniques:

- isolation levels, transaction chopping, savepoints

(8) Hardware related techniques:

- Clustered systems , main memory, GPU, and MVN database systems

References

D. Shasha and P.Bonnet Database Tuning Principles, Experiments, and Troubleshooting Techniques, Morgan Kaufmann, 2003, chapter 1, (closed collection in UoW Library)

G. Harrison Oracle Performance Survival Guide, Prentice Hall, 2009, chapter 1

L. Nossov, H. Ernst, V. Chupis, Formal SQL Tuning for Oracle Databases, Springer, 2018

S. R. Alapati, Oracle 12c Performance Tuning Recipes, A Problem-Solution Approach, Apress, 2014