
ORACLE®

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions.

The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Your. Open. World.

Current Trends in Real World Database Performance

Andrew Holdsworth
Senior Director Real World Performance, Server Technologies.

ORACLE

Real World Performance 2008

Session ID: S299785
Session Title: Growing Green Databases with Oracle on the UltraSPARC CMT Processor
Track: Database
Venue: Moscone South
Room: Rm 236
Date: 2008-09-22
Start Time: 13:00

Session ID: S298786
Session Title: Current Trends in Real-World Database Performance
Track: Database
Venue: Moscone South
Room: Rm 103
Date: 2008-09-23
Start Time: 13:00

Session ID: S298792
Session Title: Real-World Database Performance Techniques and Methods
Track: Database
Venue: Moscone South
Room: Rm 104
Date: 2008-09-25
Start Time: 12:00

Session ID: S298785
Session Title: Real-World Database Performance Roundtable
Track: Database
Venue: Moscone South
Room: Rm 104
Date: 2008-09-25
Start Time: 13:30

Current Trends in Real World Database Performance

Agenda

- Real World Performance Fundamentals
- The Performance Hacker
- The Usual Performance Issues
- OTLP Performance
- DW/BI Performance
- Hardware Review
- The Performance Core Disciplines Revisited

Real World Performance Fundamentals

Real World Performance Fundamentals

Oracle Open World 2007 Revision

- The goal of a relational database is to make data access easier for everybody.
- It is easy to create functional applications, the skill is creating applications that scale with:
  - Increasing dataset sizes
  - Increasing transaction rates
  - Increasing functional challenges (regulation, security)
  - Increasingly rigorous performance requirements
- Most applications are implemented in a suboptimal manner and have a degree of inefficiency

Real World Performance Fundamentals

Oracle Open World 2007 Revision

- The ever increasing CPU power and drop in memory costs has masked many of these inefficiencies.
- The increasing size of datasets and the trend towards consolidation means disk I/O cannot be neglected
- The bigger the database, the more poor design will be punished in terms of poor response times and throughput.
- There are no special rules for VLDB design. You just need to do a better job.

Real World Performance Fundamentals

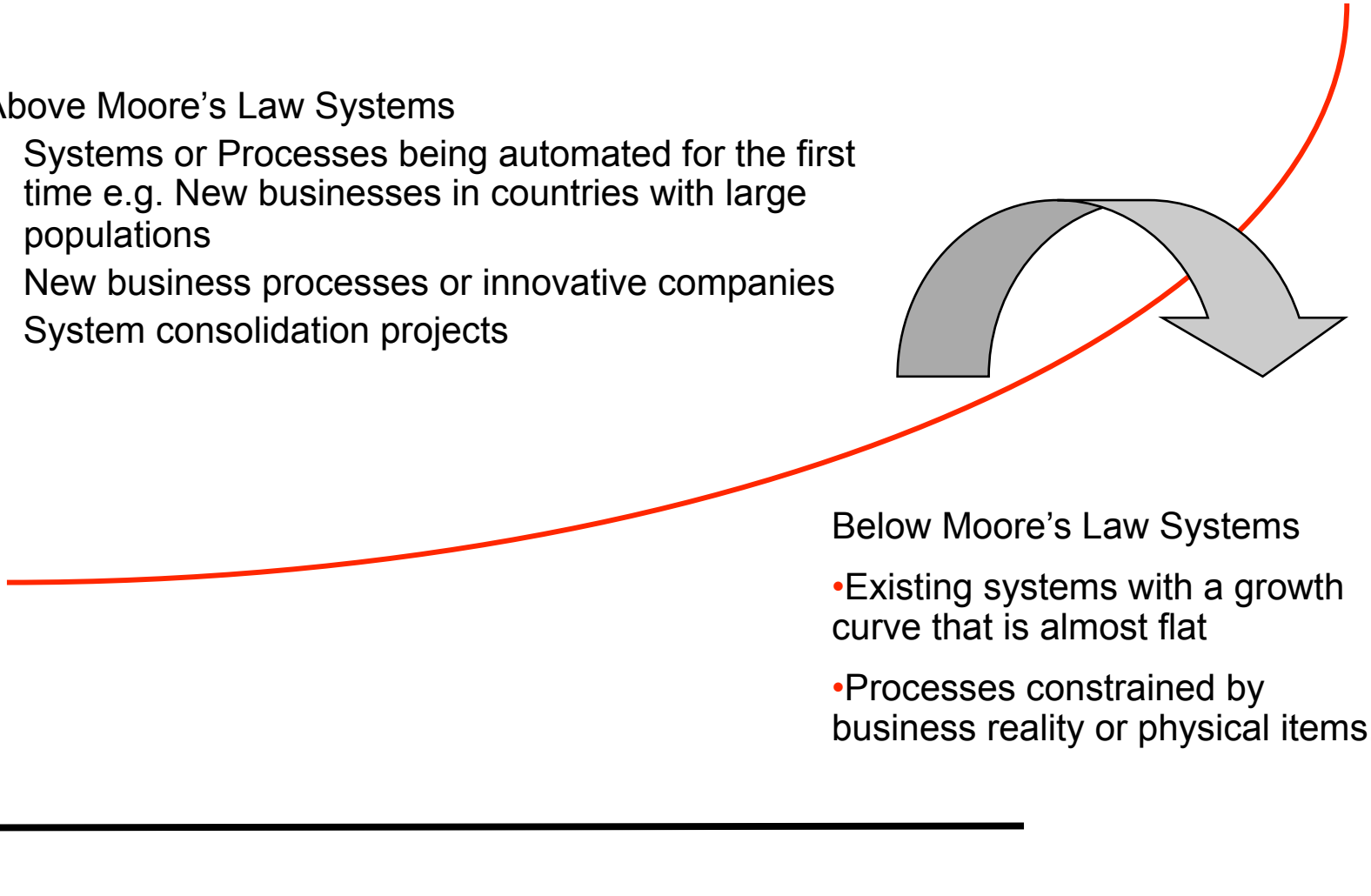
Oracle Open World 2007 Revision

- We have two types of system in ultimate performance terms
  - Systems whose growth curve exceeds Moore's Law
    - Massive automation projects for large populations of users
    - New business opportunities and startup projects
    - The results of mergers and acquisitions
  - Systems whose growth curve does not exceed Moore's Law
    - Conventional core systems (HR, Payroll, Billing)
  - Each year many systems move from above the Moore's Law curve to below the curve
  - Systems falling below the curve are targets for consolidation and re-hosting into commodity environments.

Moore's Law Reality Check

Above Moore's Law Systems

- Systems or Processes being automated for the first time e.g. New businesses in countries with large populations
- New business processes or innovative companies
- System consolidation projects



Below Moore's Law Systems

- Existing systems with a growth curve that is almost flat
- Processes constrained by business reality or physical items.

Real World Performance Fundamentals

Recognize Your System Size and Complexity

- In this session we will talk about when things get big and complicated.
- The majority of databases do not require the specialist techniques we will describe.
- Consistency of performance is more of a challenge than outright performance.
- One of the skills of a developers or DBA is to determine whether there is an issue with performance, and when additional engineering is required.
- Please remember the phrase “If it is not broken don’t fix it”

The Performance Hacker

The Performance Hacker

- This individual claims to be a performance expert but really has very little understanding of how all the system components interact.
- Root cause analysis is not in the hacker's vocabulary
- The hacker gets his information from the internet and likewise tends to write nonsense on blogs/ newsgroups etc.
- The hacker's primary tuning tools are init.ora parameters and O/S configuration parameters.
- The more obscure a parameter, the better. This is how the hacker justifies his very existence.
- The hacker changes multiple things at a time !

The Most Common Performance Issues

The Performance Core Disciplines

- SQL Execution Plans
- Buffer cache efficiency
- Connection and Cursor management
- Contention identification and management
- Hardware Capacity planning

The Most Common Real World Performance Issues

Execution Plans and the Query optimizer

- Probably the biggest issue in relational database performance
- The best execution plan optimizes use of system resources and the SQL statement executes in the shortest elapsed time.
- The first challenge is creating schema statistics that yield good cardinality estimates.
- The next challenge is deciding how much you will allow the schema statistics to evolve over time. Evolution of statistics may allow better execution plans over time but may also allow some poorer ones.
- Can your production environment tolerate any bad execution plans ?

The Most Common Real World Performance Issues

Execution Plans and the Query optimizer

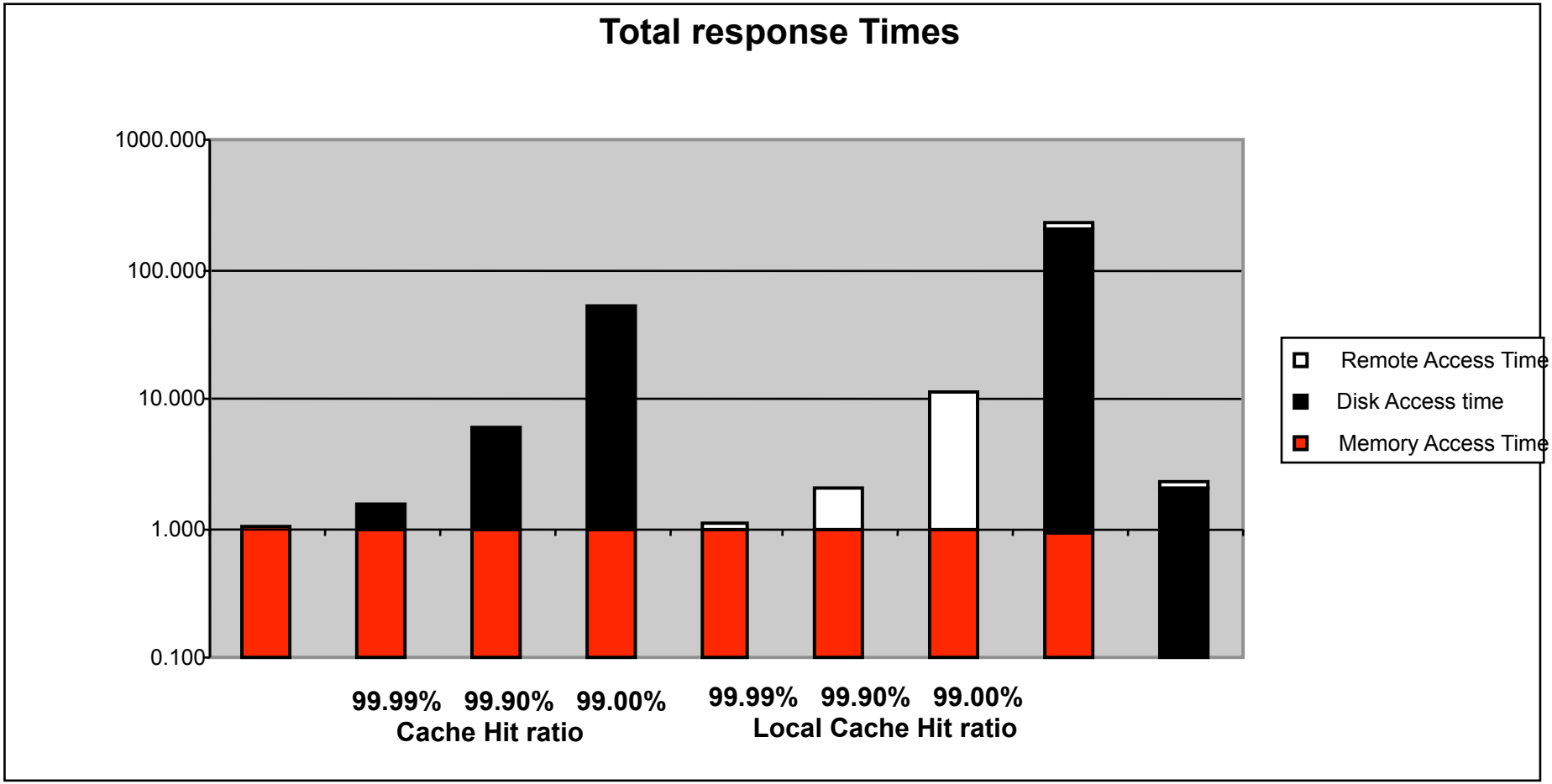
- The 6 big challenges in Query Optimization
  - Data Skew
  - Invalid column High/Low values or Binds out of range
  - Bind peeking effects
  - Data Correlation effects between Columns
  - Cardinality Approximations
  - The debugging process. The risk of making global changes to fix a small number of SQL statements
- There will be an Optimizer Exposé in the Technical Session “Real World Database Performance Techniques and Methods”

The Most Common Real World Performance Issues

Buffer Cache Efficiency

- This is all about where you get your database blocks from.
  - Local Buffer Cache: Elapsed Time 1-10 µsecs
  - Remote Buffer Cache: Elapsed Time ~ 1 msec
  - Disk: Elapsed Time 5-10 msecs
- Note the local cache access is 1000 times faster than other methods, hence the motivation to have a database in cache.
- If times for remote buffer and disk access are much greater than these values it implies block contention or poor hardware capacity planning.

Impact of I/O, Interconnect and SQL Tuning

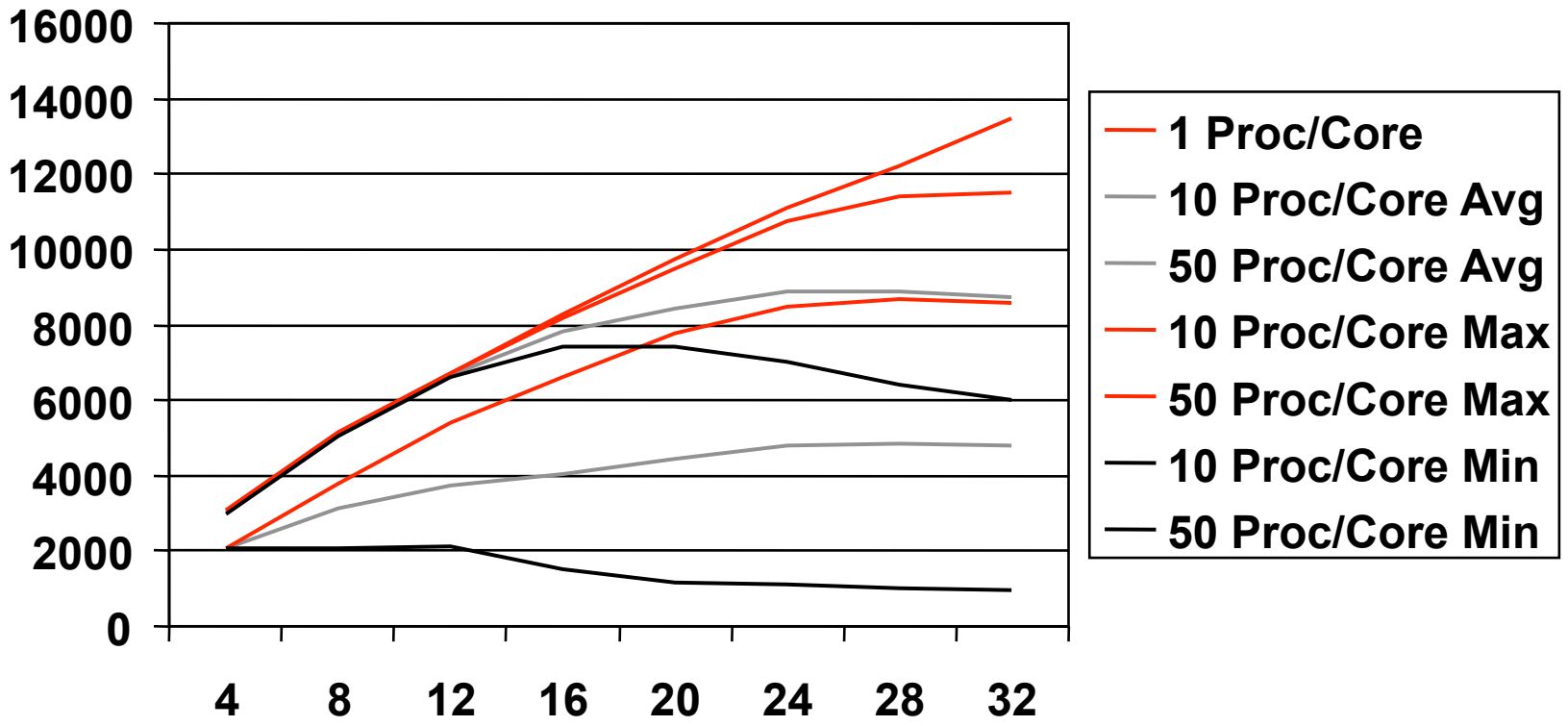


The Most Common Real World Performance Issues

Design of Database Connections, Cursors and Arrays

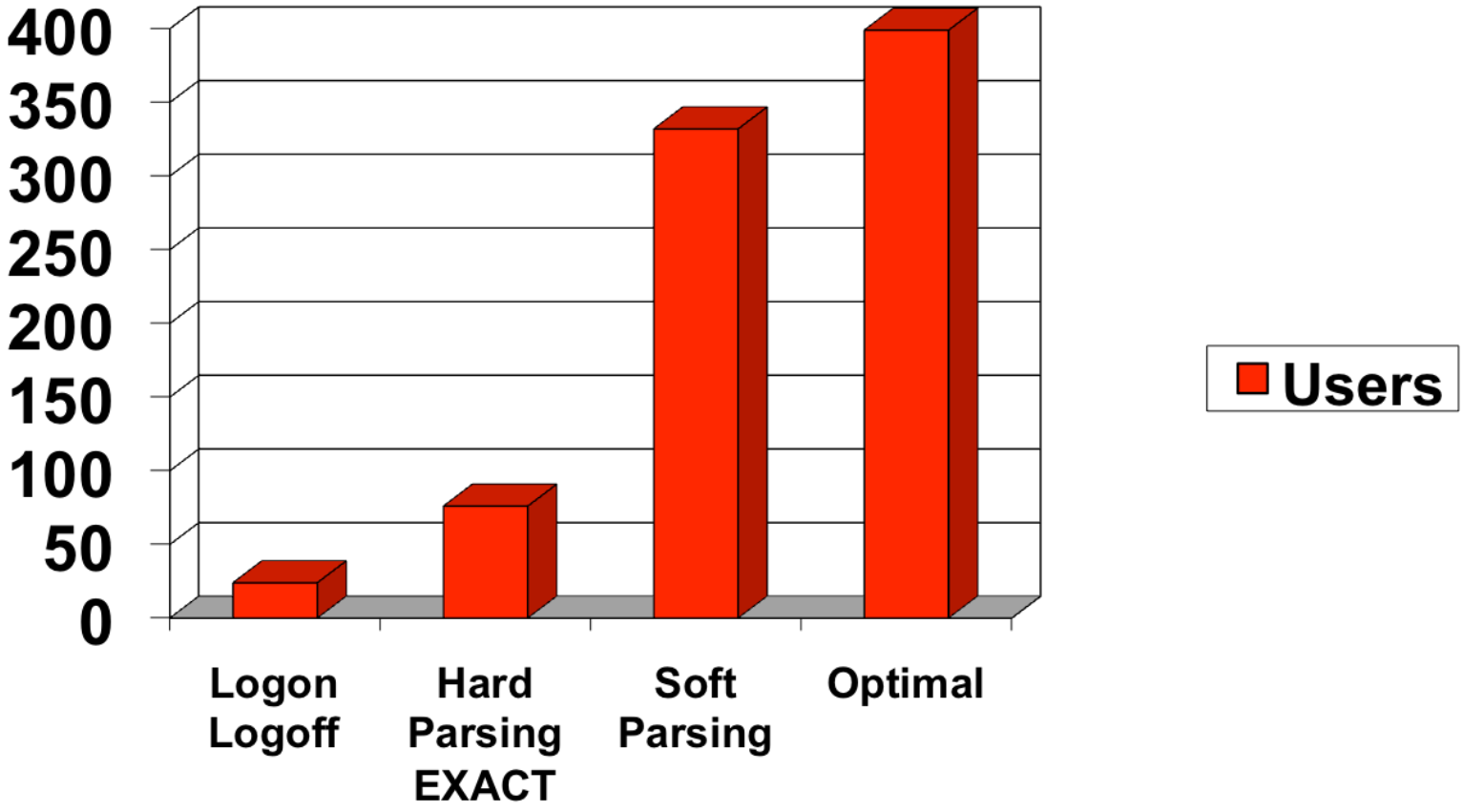
- Most systems today use too many connections to the database and these connections log on/off too often
- The “over” processing of the database makes response times unpredictable, susceptible to connection storms and difficult to fail over.
- Most systems we see these days soft parse their SQL statements. We do do still see the occasional application that uses literals and no bind variables.
- Array interfaces are used only occasionally

The Impact of Connections on Scaling and Throughput/Response Time



Database Performance Basics

Sessions & Cursors (This slide is over 10 Years old!)



The Most Common Real World Performance Issues

Contention Management

- Examples include:
  - Transaction locks
  - Hot blocks
  - Space management
  - Index leaf conflicts
- In many cases, contention within the database is a direct result of running too many processes. This results in DBAs tuning the wrong things and attempting to eliminate wait events that cannot be removed.

The Most Common Real World Performance Issues

Growth Issues

- Dealing with growth is one of the most important issues for applications growing faster than Moore's law
- This can mean increases in the following:
  - Dataset size
  - Concurrency (users)
  - Transaction rate or SQL statement execution rate
  - Application complexity of the system
  - Strategic value of the system

The Most Common Real World Performance Issues

Growth Issues

- When growing any application there will be a point in time when partitioning of data and workloads becomes inevitable.
- This may involve the following design processes
  - Partitioning of data due to dataset size issues
  - Partitioning of data for query performance issues
  - Partitioning of data for contention management
  - Partitioning of workloads for scaling issues
  - Partitioning of databases for risk management issues
- A deeper analysis of these topics will be given in the Technical Session “Real World Database Performance Techniques and Methods”

The Most Common Real World Performance Issues

Capacity Planning

- It cannot be underestimated how difficult this process is if there is no reference system/application to extrapolate from.
- Traditionally all capacity planning estimates were done with respect to initial CPU sizing as this was the most expensive item.
- Today the CPU is very often the cheapest part of the system with storage and associated networking dominating the hardware budget.

OLTP Performance

OLTP Performance

- It is quite easy to achieve high OLTP performance out of the box. The challenge is sustaining it!
- The reality is today Oracle has many live applications running > 5000 SQL statements a second
- This is a massive pipeline of work and any delay in the system can cause backlogs that may never be cleared.
- This puts a premium on keeping good statistics about your database workload and moving away from averages towards percentiles e.g. 95th or 99th percentiles.

OLTP Performance

- The most common delay to OLTP processing we see is the `log file sync` wait sometimes caused by bugs in various versions of Oracle.
- Other delays are caused by database administration functions. These include
  - Partition exchange
  - Schema statistics gathering maintenance
  - Adding data files
- We will be covering OLTP hiccups and delays in the Technical Session “Real World Database Performance Techniques and Methods”

DW/BI Performance

DW/BI Performance

- Data Warehouse appears to be an oft-abused term when describing a database.
- We have seen so-called “Data Warehouses” performing activities associated with OLTP or Operational Data Stores. These include:
  - Billing
  - Transaction validation and verification
  - Customer Service interaction
  - Part of an extensive workflow/reliable queuing network
- In summary these activities do not equate to Classical Data Warehousing.
- These are mission critical activities and should be treated as such with the correct development, testing and support processes.

DW/BI Performance

- In reality a Data Warehouse should be able to fail and the only impacted users are the analysts interrogating the data. It should not prevent normal business operations e.g. order processing, customer interactions.
- The Data Warehouse performance challenges break down into the following topics
  - Loading and maintaining the data (Classical ETL or ELT)
  - Getting good query performance when the queries are known and tested in advance
    - Canned or pre defined reports with OLTP type concurrency and response times required.
  - Getting good query performance when the queries are not defined in advance and there is no chance for testing
    - True Ad Hoc analytical workloads

DW/BI Performance

- Data Loading and maintenance
  - Data loads and ETL processes seem to be the first performance problem for many projects.
  - Much of the problems relate to the design(or lack of) of the actual load process or over complexity of the schema.
  - In many cases fast load utilities such as external tables and SQL*Loader are not used and the the load is implemented in a convoluted PL/SQL or scripting language program which performs data checking as well as the load. In many cases the checking is more resource intensive than the final INSERT. This also may be function of a convoluted schema .
  - In many cases the resources and potential for serialization in the index maintenance process have been neglected or misunderstood. This becomes more dramatic as the datasets get bigger.

DW/BI Performance

- Queries known in advance of deployment
  - These systems tend to get implemented on small hardware configurations often sized like an OLTP system.
  - Advanced knowledge of the queries allows a more conventional development cycle e.g development, testing, production.
  - The database designer can enhance the schema to make the system run more like an OLTP system with focus on minimizing system resources per query. This is usually done by selective denormalization, building specific indexes and materialized views.
  - It may take considerable hardware resources to build indexes and Materialized Views in an acceptable period of time.

DW/BI Performance

- Queries known in advance of deployment
  - In many cases indexes and MVs are taken to excess. With too many indexes and MVs the ETL/Load processes become seriously impacted. Achieving a near real time coherency of data may prove very difficult.
  - If the queries require analysis of entire or very large data sets use of parallel query, MVs or OLAP may be the only way to achieve acceptable performance. This may conflict with other users and ETL processes.
  - Systems supporting thousands of users requiring screen response time to queries is really OLTP and you need to design accordingly. There is no real Ad Hoc unstructured access to large datasets to large numbers of users !

DW/BI Performance

- Performance of true ad-hoc queries
  - If the query is not known there is very little a database designer can do to ensure good performance other than:
    - Get large amounts of hardware (CPU/Memory with balanced I/O and interconnect)
    - Set up to run all operations in parallel and utilize all available hardware
    - Partition tables along possible predicate boundaries(usually dates) and join columns
    - Keep valid Schema statistics
    - Abandon the use of indexes
    - Manage the workload to ensure the machine does not get swamped and query times remain uniform

Hardware Review

Hardware Review

- CPU
  - Getting faster through the addition of cores
  - For quad-core configurations of more than two sockets, we sometimes see problems in getting more than ¾ of the cores to scale
- Memory
  - 4-8 Gigabytes RAM per core
- Storage
  - Still dominating costs
  - Solid State
  - Database Storage for a “paradigm shift” please see tomorrow’s Larry Keynote.
- Interconnect
  - Infiniband is beating out 10 Gigabit Ethernet
- Networking
  - 10 GigE cards do not accept more packets/interrupts than their 1 GigE predecessors
  - This means 4 x 1 GigE cards give more network performance in network chatty applications

The Performance Core Disciplines Revisited

The Performance Core Disciplines Revisited

- To conclude to be good at Real World Performance you must be able to do root cause analysis for the performance problem.
- Understanding the following disciplines and how they are interconnected defines the role of the performance specialist.
  - SQL Execution Plans
  - Buffer cache efficiency
  - Connection and Cursor management
  - Contention identification and management
  - Hardware Capacity planning
- Thank you see you in the other sessions

For More Information

search.oracle.com

or

oracle.com


ORACLE®

The preceding is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions.

The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.