In This Issue:

Explain Plan Tips
Oracle Identity Management
Tuning Oracle at the Block Level
Book Review: Oracle Database 10g RMAN Backup & Recovery
Book Review: Oracle JDeveloper 10g for Forms and PL *SQL Developers
PROFESSIONAL SERVICES

Business Intelligence
By skillfully implementing and leveraging Oracle’s Business Intelligence portfolio of technology and applications, we can help organizations make better decisions, gain better insight, and drive better business results more quickly. Oracle provides the industry’s first integrated Enterprise Performance Management System.

Oracle Licensing and Advising
We have years of experience in advising customers on the optimal Oracle Licensing structure. We will guide you through the complexity of licensing rules and help you choose the best option that will help you maximize your investment. Our guidance will take you from infrastructure architecture, technology requirements and options for optimal licensing strategies.

Infrastructure Architecture Services
Our infrastructure services include design, sizing, capacity planning and licensing considerations including but not limited to E-Business Suite, Business Intelligence and custom applications. We take a holistic approach by also providing application optimization recommendations and consolidation considerations.

Oracle on Linux and Virtualization Strategies
We help our customers realize significant business value through the implementation of innovative and cost-effective technologies that provide a single source for enterprise quality support. We help our customers reduce cost and lower operating expenses.

Business Solutions
SageLogix offers a complete portfolio of custom development and implementation services. We work as an extension to your team to architect, mentor and implement business solutions that leverage Oracle technologies.

ORACLE AND LINUX MANAGED SERVICES

Our Managed Services provides our customers the ability to entrust the administration and maintenance of their complex Oracle environments to SageLogix.

- Complete Outsourcing or Compliment Existing Team
- Best Practices
- 24 x 7 Monitoring
- Increase Operational and Staff Efficiencies
- Adoption of Additional Technologies (RAC, DataGuard, Virtualization...)

SageTools
SageTools are a powerful, comprehensive and configurable toolset and methodology that represents decades of cumulative experience in the management of Oracle based environments focusing on:

- Proactive Detection and Correction
- Reducing Business Risk and Unplanned Outages
- Performance Optimization and Consistency
- Optimal Resource Utilization Monitoring

REST EASY...
YOUR CRITICAL BUSINESS SOLUTIONS ARE BEING MANAGED
24 x 7 IN OUR HANDS

THE ORACLE “GO TO” COMPANY
sales@sagelogix.com
toll free: 877.846.6731
www.sagelogix.com
By the time you read this newsletter the term of the current RMOUG Board will have come to a close. In the last year the Board has accomplished some major achievements; RMOUG has filed with the IRS for Non Profit status. That paperwork is under review by the IRS at this time. As a non profit organization it will better assist us in our quest for donations to the Stan Yellott Scholarship Fund. RMOUG again gave out a number of scholarships from the fund this past year. We saw another successful Training Days event along with our Quarterly Educational Workshops. We had a SIG Meeting in Colorado Springs with a presentation by Tim Gorman, and plans are in the works for a Ft. Collins SIG Meeting. We published, and distributed improved newsletters and also continued work on improving the RMOUG website. I would like to take this opportunity to thank my fellow Board Members for all their hard work this past year in helping to achieve those milestones. Peggy King Vice President and Training Days Director, Linda Seley Treasurer, Allison Leech Newsletter Director and Vendors Director, Jon Arnold Secretary and Programs Director, Chip Briggs SIGS Director, Barbara Lewis Director of IS, Ron Bich Director of Membership, and John Peterson Director of Education. I must also mention Heidi Kuhn our Administrative Assistant whose hard work and dedication for a number of years helps to keep us functioning in an efficient manner. The new Board that will be seated at the June Board Meeting includes some returning members along with some new faces. They are; Ron Bich, Pat Van Buskirk, Niklas Iveslatt, Peggy King, Allison Leech, Barbara Lewis, John Peterson, Kathy Robb, Connie Schafer, Bill Schwartzkopf, and Linda Seley. I would like to thank you for your continued support of RMOUG and to remind you that once again we are always looking for presentations at the Quarterly Workshops, and also Newsletter articles.

Please submit any ideas to either; ProgramsDir@rmoug.org or NewsletterDir@rmoug.org.

Regards

Bill Schwartzkopf
"We can never replace all the great things you do but we will help carry on all the things you started", George T

The Rocky Mountain Oracle User's Group (RMOUG) is dedicated to helping others. We give our time and put forth effort not for financial gain, but to improve the professional, technical and personal lives of our colleagues. Stan Yellott personified this dedication by helping not only his professional colleagues, but by extending the effort to high school and college students, the next generation of IT professionals. The Stan Yellott Scholarship Fund is founded on the principle of going above and beyond to assist our current and future colleagues.

RMOUG started to award annual college scholarships to high school students. The Stan Yellott Scholarship Fund continues the educational mission by assisting deserving students.

Scholarships of $1000 each are awarded semi-annually in November and March to students interested in pursuing studies related to Information Technology.

An application can be filled out or downloaded at www.rmoug.org

Summer 2008, Stan Yellott Scholarship Recipient

Thang Trinh

Benedict Spinoza said, “Desire is the essence of a man.” As for me, that desire is to be a computer expert, and I knew that I would enter the computer field in the future.

I started to pursue my dream by getting a B.S. degree in Information Technology at University of Natural Sciences in Vietnam, in 2005. However, as Yoshida Kenko said, “Ambition never comes to an end,” I decided to continue my education by pursuing Master degree in Electrical and Computer Engineering in the U.S.
Although I had prepared a lot, I couldn’t imagine the many hardships I would endure upon coming to the United States. The first challenge was the language barrier. The second was the unfamiliar educational environment. However, with the desire to learn new things and the encouragement from my family and friends, I have overcome all of the obstacles. The last challenge, one that concerns me the most, is the financial difficulties. The living cost and tuition here, which is around thirty five thousand dollars per year, are sometimes too expensive for me to afford.

I am very lucky to have the opportunity to study in a country which is well known for advanced technology. Therefore, I always try my best to achieve my goals. I have been able to maintain a 4.0 GPA. Moreover, I currently have a part time job on the campus. The last challenge, one that concerns me the most, is the financial difficulties. I am still able to tutor my friends. I am so proud of what I’ve achieved in the United States so far.

**Charles T. Gooding**

Charles T. Gooding III is currently a junior studying Computer and Information Technology at Purdue University. By the time he graduates, he will have taken over eighteen technology courses, with five specifically focused on databases. He feels that his biggest academic achievement has been the amount that he has learned and accomplished through his challenging courses. After graduating in May 2009, he plans on pursuing a career as a database administrator.

**Dorothy Trujillo**

Dorothy Trujillo graduated from the Community College of Denver with an Associates of Arts degree and with honors from Phi Theta Kappa. She now attends the University of Colorado at Denver where she will get her Bachelors degree in Information Systems. For the last five years she has been working for the United State Geological Survey on the Geochemical Database.
Book Review: Oracle Database 10g RMAN Backup & Recovery

Matthew Hart and Robert G. Freeman, co-authors of Oracle Database 10g RMAN Backup & Recovery, have put together a complete and concise description of RMAN which is applicable for DBAs and Sys Admins from the raw beginner to the advanced professional. This new book is a follow-up to their previous Oracle 9i RMAN Backup & Recovery and focuses on 10g Release 2, pointing out differences in functionality between Releases 1 and 2. One of the features deals with Oracle’s new Flashback Technology.

As quoted in About the Authors, Robert Freeman “has been an Oracle DBA for so long he can’t remember now when he actually entered SQL*Plus for the first time.” He is the author of many Oracle books, including Portable DBA: Oracle.

Matthew Hart has worked with RMAN since its inception and is the co-author of the Oracle Press books. Headquartered in Kansas City, Missouri, Robert works with Oracle customers to perfect backup and recovery strategies.

10g RMAN Backup & Recovery is divided into four parts. Part I provides an introduction to RMAN’s backup and recovery principles and delves deeply into the architecture.

Part II deals with setting up RMAN for initial usage, including an extremely complete configuration guide and details for creating and maintaining the recovery catalog. The media section covers tape and disk management, including Oracle Secure Backup, Veritas NetBackup, EMC NetWorker Module and IBM’s Tivoli Storage Manager.

Part III is about using RMAN effectively, including basic backups, restoring and recovery. A section on advanced recovery topics reviews incomplete recoveries, point-in-time recovery, verifying backups and cross-platform database movement. Oracle Enterprise Manager, for both Grid Control and Database Control are explained, along with instructions for installing and configuring both.
Oracle’s new Flashback Technology is appropriately titled “Surviving User Errors” and details the flashback query, flashback table, flashback drop and flashback database. Also included in Part III are instructions for maintaining, tuning, monitoring and reporting for RMAN.

Part IV outlines topics beyond basic backup and recovery - cloning the target database, RMAN usage with Data Guard and the standby database, RAC backup and recovery challenges, and RMAN in the Sync and Split Technology. Several case studies describe examples of recovering from database failures with and without the recovery catalog.

A complete list of command syntax for RMAN is included in the Appendixes, along with details for setting up an RMAN test environment and a listing of V$ and rc_* views.

Each chapter contains mini “Workshops” which provide step-by-step recipes to get the job done quickly, from setup, configuration, backup and restore.

This is an excellent book with very few discrepancies. I did have a problem with the explanation of incremental backups in Chapter 9, in that they used level 2 backups in their figures, with level 1 in the text. Per my RMAN expert, Tim Gorman, Oracle considers level=2 to be obsolete and have excluded it from the documentation. However, it is still alive and well, even in 11g. I found a complete explanation in the following:

http://download-west.oracle.com/docs/cd/A97630_01/server.920/a96566/rcmconc1.htm.

I can highly recommend this book for DBAs at every level of knowledge and experience. It is well written, clear, concise, and a book you’ll want to have by your keyboard.

Publisher: Oracle Press, 2007
Retail Price: $59.99

Pat Van Buskirk is the Oracle DBA for the National Radio Astronomy Observatory in New Mexico. She has been involved with Oracle since 1990, as an applications programmer, Forms developer, project manager and DBA. She was originally a member of the San Diego Oracle User’s Group.
Title: Oracle JDeveloper 10g For Forms and PL*SQL Developers: A guide to web development with Oracle ADF

Author: Peter Koletzke and Duncan Mills

Suggested Retail Price: $49.99 ($35 on Amazon.com)

Audience: Developer or anyone else needing to understand ADF.

Skill Level: Intermediate

Topic: Developing ADF solutions using JDeveloper and using the ADF BC framework for database connectivity.

Ease of reading and understanding: The book is well written and the text is easy to read and follow. There are a few mistakes that are not covered by the errata yet, but they do not stop you for long.

Is the topic well covered and complete: Yes. As an introduction to building ADF solutions with JDeveloper the book does a good job to explain the basics.

Overall impression of the books content and subject matter, author's style etc: This book start out with reviewing all the different areas from a theoretical perspective. There are quite a few areas you need to understand due to ADF being a layered development framework. When the theory required for those areas are covered, the book moves into building a simple two page application step by step. This is a great review of a lot of the mechanics used to build an application with ADF. After this simple application is completed, the book spend several chapters on building a fairly complicated application. The application used is based around HR, and includes enough depth to show, and let the reader practice, more advanced technologies used when building real database driven ADF applications.
Tuning Oracle at the Block Level

ABSTRACT
This paper is not for beginners. It is a look at Oracle moving at the block level. It is not meant to be an all encompassing detailed paper. That would take weeks or even months to write. But, it is meant to give you a taste of some of what I’ve seen so that you can look into it and take it to the next level. This subject is only meant for advanced DBAs with several years of experience. This paper will review some of the concepts concerning how blocks move from disk to memory, how they become current blocks, how they become Consistent Read (CR) versions and other interesting information. This presentation specifically covers dumping blocks of data, reading block dumps, the cache buffer operation, general block level concepts, the current and CR version of blocks, delayed block cleanup, the hot/cold side of the LRU and some useful scripts along the way.

Why you should understand some block level concepts? You wouldn’t use this for everyday use, but when you really need to dig deep to solve a difficult problem that is absorbing much or most of your system. Another reason to better your understanding is the large increase in hardware memory and the move toward memory resident databases for some of the smaller databases out there. Also, with a move toward RAC/Grid computing, problems with blocks moving through the interconnect could be where your future problems will be hiding. Lastly, when you can visually see how something works at the micro level, it can help you understand it better at the macro level.

TERMINOLOGY:
Since everyone looks at terminology differently, I want to define how I’ll use some terms within the paper. Within the “Instance Activity” section of a statspack report or AWR report (Automatic Workload Repository Report – very similar to statspack but uses different internal tables/views), you can find several items that tell you how much information is passing through your system. The “Session Logical Reads” are all reads which are cached in memory. This includes both the consistent gets and also the db block gets which are also in the report. The “Consistent Gets” are the reads of a block that are in the cache. They should NOT be confused with a consistent read (cr) version of a block in the buffer cache which is a block at a consistent point in time or SCN (note that usually the current version of a block is the one in the buffer cache and usually your buffer cache is about 95% current blocks.). Also in a statspack or AWR report you’ll see the “Db block gets” which are the blocks gotten to be changed. You MUST use the CURRENT (curr) version of a block and not a cr block when you are going to change it. These can be either blocks that are read from disk, the disk cache or O/S cache. The fact that these reads are not all truly read from disk is why people are in constant debate on how fast memory or disk is. People think that they’re reading blocks very fast from disk, giving them the illusion that it’s almost as fast as reading them from memory, when in fact they ARE reading them from memory (the disk cache memory or the O/S cache memory). To Oracle though, it’s a disk read, since it’s a read outside Oracle’s buffer cache, causing the debate to go on. There are also “physical reads direct” which bypass cache using things like Parallel Query (these are also not in hit ratios). Here’s an example of the Instance Activity section:

<table>
<thead>
<tr>
<th>Statistic</th>
<th>Total</th>
<th>per Second</th>
</tr>
</thead>
<tbody>
<tr>
<td>branch node splits</td>
<td>7,162</td>
<td>0.1</td>
</tr>
<tr>
<td>consistent gets</td>
<td>12,931,880,777</td>
<td>152,858.8</td>
</tr>
<tr>
<td>current blocks converted for CR</td>
<td>75,709</td>
<td>0.9</td>
</tr>
<tr>
<td>db block changes</td>
<td>343,632,442</td>
<td>4,061.0</td>
</tr>
<tr>
<td>db block gets</td>
<td>380,323,756</td>
<td>4,613.0</td>
</tr>
<tr>
<td>hot buffers moved to head of LRU</td>
<td>197,282,394</td>
<td>2,331.7</td>
</tr>
<tr>
<td>lead node 90-10 splits</td>
<td>26,429</td>
<td>0.3</td>
</tr>
<tr>
<td>lead node splits</td>
<td>840,436</td>
<td>9.9</td>
</tr>
<tr>
<td>logn cumulative</td>
<td>21,369</td>
<td>0.3</td>
</tr>
<tr>
<td>physical reads</td>
<td>504,643,275</td>
<td>5,965.1</td>
</tr>
<tr>
<td>physical writes</td>
<td>49,724,268</td>
<td>587.6</td>
</tr>
<tr>
<td>session logical reads</td>
<td>13,322,170,917</td>
<td>157,472.5</td>
</tr>
<tr>
<td>sorts (disk)</td>
<td>4,132</td>
<td>0.1</td>
</tr>
<tr>
<td>sorts (memory)</td>
<td>7,938,085</td>
<td>93.6</td>
</tr>
<tr>
<td>sorts (rows)</td>
<td>906,107,041</td>
<td>10,711.7</td>
</tr>
<tr>
<td>table fetch continued row</td>
<td>25,506,365</td>
<td>301.5</td>
</tr>
<tr>
<td>table scans (long tables)</td>
<td>111</td>
<td>0.0</td>
</tr>
<tr>
<td>table scans (short tables)</td>
<td>1,541,085</td>
<td>10.2</td>
</tr>
</tbody>
</table>

Tuning the RAC Cluster Interconnect
When you have RAC issues, the problem are the same times TWO (for two nodes)! Reading a lot of blocks because a query is poorly tuned becomes a worse problem when all those blocks pass from one node to the other across the interconnect. Here is an example
of the Top Wait section from a two node RAC system moving blocks from node to node causing “global cache cr request” waits. The users are waiting for a cr version of a block from the other node.

Top 5 Timed Events

<table>
<thead>
<tr>
<th>Event</th>
<th>Waits</th>
<th>Time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>global cache cr request</td>
<td>520</td>
<td>154</td>
</tr>
<tr>
<td>CPU time</td>
<td>72.20</td>
<td></td>
</tr>
<tr>
<td>global cache null to x</td>
<td>478</td>
<td>1</td>
</tr>
<tr>
<td>control file sequential read</td>
<td>.52</td>
<td></td>
</tr>
<tr>
<td>control file parallel write</td>
<td>.28</td>
<td></td>
</tr>
</tbody>
</table>

Current & CR Versions

An internal table called the buffer hash table (x$bh) holds block headers. There is a hash chain which blocks are linked to that are protected by a CBC latch (cache buffers chains latch). This links to the actual address located in memory (the memory set up with db_cache size, which is the cache used for data). For a given block in Oracle, only one version of a block is CURRENT and there are no more than 5 other CR versions of the block (as of V9). So there are only six versions of a given block (maximum) in memory at a time. Later in this paper, I will tell you how to control this with an undocumented parameter. Oracle recommends not to use the undocumented parameters unless you are directed by Oracle support or your database may not be supported. When you perform a DML (Data Manipulation Lock) transaction, which is an INSERT, UPDATE or DELETE, you always need the CURRENT version of a block. Oracle has something in Oracle 10g called "in memory update" (IMU) which can give you some hard to understand results when you are viewing information at the block level. When you are querying a block for the first time you always use the CURRENT version of a block. It the block is being used, you will build a CLONE of the block called a CONSISTENT READ (CR) version by applying any undo needed to the CURRENT version of the block to get it to a point in time that makes it useful to you (perhaps you need a version of the block before the DML was performed and not committed by another user). This complex & Oracle patented process may include reading the ITL (Interested Transaction List – This is populated when someone does a DML on a block), and mapping the record to the UNDO HEADER, or directly to the UNDO BLOCK and then applying the UNDO to get the correct CR version that you need. So let’s take a look at how this happens:

- User 1 updates a record in block 777 (user1 has not committed)
- User 2 queries the same block and sees that the lock byte is set for a row being queried
- User 2 goes to the ITL portion of the block and get the XID (transaction ID)
- The XID maps to the UNDO block which holds the information before the update was done
- A clone of the block is done (call it block 778)
- The UNDO information is applied to the block rolling it forward, but to where it used to be
- Block 777 is a CURRENT block
- Block 778 is a CONSISTENT READ block before the User 1 update occurred
- If another user wants to do a query before the commit, they can also read the CR version.

Note ESPECIALLY the fact that the block is not ROLLED BACK to what it was, but it is ROLLED FORWARD to what it used to be. While the result is the same, how Oracle performs this operation is CRITICAL to understanding how Oracle works. They are always moving forward in time (this is why the REDO works – it’s always applying things forward sequentially). There are also links to all blocks for the LRU (Least Recently Used) and LRU-W (Least Recently Used – Write) to help make buffer replacement and writing much faster. This is also maintained in the buffer headers.

Biggest Problems

The biggest problems can be quickly found in the Statspack or AWR reports. One of the first places that you can check is the Top 5 Wait Events. The Top 5 Wait Events section of statspack is probably the most revealing section in the entire report when you are trying to quickly eliminate bottlenecks on your system. This section of the report shows the Top 5 Wait Events, the full list of Wait Events, and the Background Wait Events. Identifying major wait events will help you to target your tuning efforts to the most burning issues on your system. ITIMED_STATISTICS is true, then the events are ordered in time waited, if false, then the events are ordered by the number of waits.

In the listing below, we see a large number of waits related to reading a single block (db file sequential reads) and also waits for latches (latch free). We also see some pretty high waits for some of the writing to both datafiles and log files as well as other potential issues with log file contention. To solve these issues (identify which are truly major issues), we must narrow this down these issues by investigating the granular reports within other sections of STATSPACK.

Top 5 Wait Events

<table>
<thead>
<tr>
<th>Event</th>
<th>Waits</th>
<th>Time (s)</th>
<th>% Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>db file sequential read</td>
<td>15,877,104</td>
<td></td>
<td>82.24</td>
</tr>
<tr>
<td>latch free</td>
<td>4,016,773</td>
<td></td>
<td>99.55</td>
</tr>
<tr>
<td>log file sync</td>
<td>1,057,224</td>
<td></td>
<td>2.70</td>
</tr>
<tr>
<td>log file parallel write</td>
<td>1,054,006</td>
<td></td>
<td>1.65</td>
</tr>
<tr>
<td>db file parallel write</td>
<td>1,221,755</td>
<td></td>
<td>1.49</td>
</tr>
</tbody>
</table>

The solution for waits depend on the type of wait. Here are some of the most common problems; explanations and potential solutions are given below.

DB File Scattered Read – This generally indicates waits related to full table scans. As full table scans are pulled into memory, they are scattered throughout the buffer cache since it is usually unlikely that they fall into contiguous buffers. A large number indicates that there may be missing or suppressed indexes. This could also be preferred since it may be more efficient to perform a full table scan than an index scan. Check to ensure full table scans are necessary when you see these waits. Try to cache small tables to avoid reading them in over and over again.

DB File Sequential Read – This generally indicates a single block read (an index read for example). A large number could indicate poor joining orders of tables or unselective indexing. This number will certainly be large (normally) for a high-transaction, well-tuned system. You should correlate this wait with other known issues within the statspack report such as inefficient SQL. Check to ensure index scans are necessary and check join orders for multiple table joins. The DB_CACHE_SIZE will also be a determining factor in how often these waits show up; hash-area joins causing problems should show up in the PGA memory but similarly are
Free Buffer Waits - This indicates your system is waiting for a buffer in memory, because none is currently available. Waits in this category could indicate that you need to increase the DB_BUFFER_CACHE, if all your SQL is tuned. Free buffer waits could also indicate that unselective SQL is causing data to flood the buffer cache with index blocks, leaving none for this particular statement that is waiting for the system to process. This normally indicates that there is a substantial amount of DML (insert/update/delete) is being done and the Database Writer (DBWR) is not writing quickly enough, the buffer cache could be full of multiple versions of the same buffer, causing great inefficiency. To address this, you may want to consider accelerating incremental checkpointing, using more DBWR processes, or increasing the number of physical disks.

Buffer Busy Wait – This is a wait for a buffer that is being used in an unshareable way or is being read into the buffer cache. Buffer busy waits should not be greater than 1 percent. Check the buffer wait statistics section (or V$WAITSTAT) to find out if the wait is on a segment header. If this is the case, increase the freelist groups or increase the pctused to pctfree gap. If the wait is on an undo header, you can address this by adding rollback segments; if it’s on an undo block, you need to reduce the data density on the table driving this consistent read or increase the DB_CACHE_SIZE. If the wait is on a data block, you can move data to another block to avoid this hot block, increase the freelists on the table, or use Locally Managed Tablespaces (LMT’s). If it’s on an index block, you should rebuild the index, partition the index, or use a reverse key index. To prevent buffer busy waits related to data blocks, you can also use a smaller block size: fewer records fall within a single block in this case, so it’s not as “hot.” When a DML (insert/update/delete) occurs, Oracle Database writes information into the block, including all users who are “interested” in the state of the block (Interested Transaction List (ITL)). To decrease waits in this area, you can increase the intrans, which will create the space in the block to allow multiple ITL slots. You can also increase the pctfree on the table where this block exists (this writes the ITL information up to the number specified by maxtrans, when there are not enough slots built with the intrans that is specified).

Latch Free – Latches are low-level queueing mechanisms (they’re accurately referred to as mutually exclusion mechanisms) used to protect shared memory structures in the System Global Area (SGA). Latches are like locks on memory that are very quickly obtained and released. Latches are used to prevent concurrent access to a shared memory structure. If the latch is not available, a latch free miss is recorded. Most latch problems are related to the failure to use bind variables (library cache latch), redo generation issues (redo allocation latch), buffer cache contention issues (cache buffers Iru chain) and hot blocks in the buffer cache (cache buffers chain). There are also latch waits related to bugs; check MetaLink for bug reports if you suspect this is the case (oracle.com/support). When latch miss ratios are greater than 0.5 percent, you should investigate the issue. I will cover latch waits in detail in my next Oracle Magazine column; the topic requires an article in itself.

Enqueue - An enqueue is a lock that protects a shared resource. Locks protect shared resources such as data in a record, to prevent two people from updating the same data at the same time. It includes a queueing mechanism, which is FIFO (first in, first out). Note that Oracle’s latching mechanism is not FIFO. Enqueue waits usually point to the ST enqueue, HW enqueue, and the TX4 enqueue. The ST enqueue is used for space management and allocation for dictionary-managed tablespaces. Use LMTs, or try to pre-allocate extents or at least make the next extent larger for problematic dictionary-managed tablespaces. HW enqueues are used with the high-water mark of a segment; manually allocating the extents can circumvent this wait. TX4 are the most common enqueue waits. TX4 enqueue waits is usually the result of one of three issues. The first issue is duplicates in a unique index; you need to commit/rollback to free the enqueue. The second is multiple updates to the same bitmap index fragment. Since a single bitmap fragment may contain multiple rowids, you need to issue a commit or rollback to free the enqueue when multiple users are trying to update the same fragment. The third and most likely issue is when multiple users are updating the same block. If there are no free ITL slots, a block level lock could occur. You can easily avoid this scenario by increasing the intrans and/or maxtrans to allow multiple ITL slots and/or by increasing the pctfree on the table. Last, there is a way to get TM locks, which are table locks. If you have foreign keys, be sure to index them to avoid this general locking issue.

Log Buffer Space – This wait occurs because you are writing the log buffer faster than LGWR can write it to the redo logs, or because log switches are too slow. To address this problem, increase the size of the log files, or increase the size of the log buffer, or get faster disks to write to. You might even consider using solid-state disks, for their high speed.

Log File Switch – All commit requests are waiting for ‘logfile switch (archiving needed)’ or ‘logfile switch (chkpt. Incomplete)’. Ensure that the archive disk is not full or slow. DBWR may be too slow due to I/O. You may need to add more or larger redo logs and you may potentially need to add database writers if the DBWR is the problem.

Log File Sync – When a user commits or rolls back data, the session’s redo is flushed to the Redo Logs from the Log Buffer by the LGWR. This process must wait for this to successfully complete. To reduce this, try to commit more records (try to commit a batch of 50 instead of one at a time if possible). Put redo logs on a faster disk, or alternate redo logs on different physical disks, to reduce the archiving effect on LGWR. Don’t use RAID 5, since it is very slow for applications that write a lot; potentially consider using filesystem direct I/O or raw devices, which are very fast at writing information.

Idle Events – There are also several idle wait events listed after the output that can be ignored. Idle events are generally listed at the bottom of each section and include things like SQL*Net message to/from client and other background related timings. Idle events are listed in the stats$Idle_event table.

Wait Events Quick Reference:

<table>
<thead>
<tr>
<th>Wait Problem</th>
<th>Potential Fix</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sequential Read</td>
<td>Indicates many index reads – tune the code (especially joins)</td>
</tr>
<tr>
<td>Scattered Read</td>
<td>Indicates many full table scans – tune the code; cache small tables</td>
</tr>
<tr>
<td>Free Buffer</td>
<td>Increase the DB_CACHE_SIZE; shorten the checkpoint; tune the code</td>
</tr>
<tr>
<td>Buffer Busy</td>
<td>Segment Header – Add freelists or freelist groups</td>
</tr>
<tr>
<td>Buffer Busy</td>
<td>Data Block – Separate ‘hot’ data; use reverse key indexes; smaller blocks</td>
</tr>
<tr>
<td></td>
<td>Data Block – Increase intrans and/or maxtrans</td>
</tr>
</tbody>
</table>

Note that Oracle's latching mechanism is not FIFO. Enqueue waits usually point to the ST enqueue, HW enqueue, and the TX4 enqueue. The ST enqueue is used for space management and allocation for dictionary-managed tablespaces. Use LMTs, or try to pre-allocate extents or at least make the next extent larger for problematic dictionary-managed tablespaces. HW enqueues are used with the high-water mark of a segment; manually allocating the extents can circumvent this wait. TX4 are the most common enqueue waits. TX4 enqueue waits is usually the result of one of three issues. The first issue is duplicates in a unique index; you need to commit/rollback to free the enqueue. The second is multiple updates to the same bitmap index fragment. Since a single bitmap fragment may contain multiple rowids, you need to issue a commit or rollback to free the enqueue when multiple users are trying to update the same fragment. The third and most likely issue is when multiple users are updating the same block. If there are no free ITL slots, a block level lock could occur. You can easily avoid this scenario by increasing the intrans and/or maxtrans to allow multiple ITL slots and/or by increasing the pctfree on the table. Last, there is a way to get TM locks, which are table locks. If you have foreign keys, be sure to index them to avoid this general locking issue.
Latch Issues

Latches are low-level queueing mechanisms (they’re accurately referred to as mutually exclusion mechanisms) used to protect shared memory structures in the SGA (memory). Latches are like locks on memory that are very quickly gotten and released. Latches are used to prevent concurrent access to a shared memory structure. If the latch is not available, then a latch free miss is recorded. Most latch problems are related to NOT using bind variables (library cache latch), redo generation issues (redo allocation latch), buffer cache contention issues (cache buffers lru chain) and hot blocks in the buffer cache (cache buffers chain). There are also latch waits related to bugs, so check Metalink as well. When latch miss ratios are greater than 0.5%, you should investigate the issue.

There are two types of latches, a willing to wait and not willing to wait latch. Latches that are willing to wait will try to acquire a latch. If none are available, it will spin and then request the latch again. It will continue to do this up to the _SPIN_COUNT initialization parameter (note that spinning costs CPU). If it can’t get a latch after spinning up to the _SPIN_COUNT, it will go to sleep. It will then sleep, not doing anything for a while, and then will wake up after one centisecond (one hundredth of a second). It will then start this process again, spinning up to the _SPIN_COUNT and then sleeping for twice as long (two centiseconds). After doing this again it will double again. So the pattern is 1,2,4,8 etc. It will do this until it gets the latch. Every time the latch sleeps, it will create a latch free wait. An example of a “willing to wait” latch is a library cache latch. Some latches are “not willing to wait.” This type of latch does not wait for the latch to become available. They immediately time out and retry to obtain the latch. A redo copy latch is an example of a “not willing to wait” latch. A not willing to wait latch will generate information for the immediate_gets and the immediate_misses columns of the V$LATCH view and also in the statspack report. The hit ratio for these latches should also approach 99% and the misses should never fall below 1 percent misses.

By viewing this section of statspack or querying the V$LATCH view, you can see how many processes had to wait or sleep and the number of times they had to sleep. V$LATCHHOLDER, V$LATCHNAME and V$LATCH_CHILDREN are also helpful in investigating latch issues. Here is a partial listing of the latch activity section; there are three sections (latch activity, latch sleep and latch miss) of the statspack report (this one has a library cache problem):

| Latch | Got | Put | Avg | Wait | P
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>CXL freelist latch</td>
<td>4,904</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>cache buffer handles</td>
<td>988,692</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>cache buffers chain</td>
<td>761,700,503</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>cache buffers lru chain</td>
<td>9,111,269</td>
<td>0.1</td>
<td>0.8</td>
<td>19,824,466</td>
<td>3.0</td>
</tr>
<tr>
<td>library cache</td>
<td>43,602,665</td>
<td>0.2</td>
<td>2.0</td>
<td>213,596</td>
<td>2.0</td>
</tr>
<tr>
<td>redo allocation</td>
<td>12,446,906</td>
<td>0.2</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>redo copy</td>
<td>320</td>
<td>0.0</td>
<td>0.0</td>
<td>10,335,436</td>
<td>0.0</td>
</tr>
<tr>
<td>user lock</td>
<td>1,573</td>
<td>0.3</td>
<td>1.2</td>
<td>0.0</td>
<td>0.0</td>
</tr>
</tbody>
</table>

Latch issues to look for and remember:

Latch Free - When ‘latch free’ is high in the wait events section of the report, then there are problems that need to be investigated in the latch section of the report. This section will help you look for which latches are a problem. A sleeping latch (couldn’t get the latch and sleeping until the next try) or spinning latch (waiting and retrying based on spin count) latches.

Library Cache and Shared Pool - The library cache latch serializes access to objects in the library cache. Every time a SQL or PL/SQL procedure, package, function or trigger is executed, this latch is used. It is also used intensively during parse operations. A single shared pool latch protected the allocation of memory in the library cache in Oracle8i; since Oracle9i, there are 7 children latches for this. Contention for the ‘shared pool,’ ‘library cache pin,’ or ‘library cache’ latches primarily occur when the shared pool is too small or when statements are not reused. Statements are not usually reused when bind variables are not used. Common but not exact SQL floods the Shared Pool. Increasing the size of the shared pool only makes the latch problem worse. You can also set the CURSOR_SHARING=FORCE (or SIMILAR in 9i) initialization parameter to help fix this issue and to reduce problems when bind variables are not used. But, the shared pool and library cache latch issues also occur when space is needed in the library cache when it is set too small for the number of SQL statements that need to be processed. While space is being freed up in order to load a SQL or PL/SQL statement, the latch is being held exclusively and other users must wait. You can help to reduce contention by increasing the shared pool or by pinning large SQL and PL/SQL statements in memory using the DBMS_Shared_Pool.Keep procedures.

Redo Copy - The number of “redo copy” latches has a default of 2*CPU_COUNT, but can be set using the _LOG_SIMULTANEOUS_COPIES initialization parameter. Increasing this parameter may help to reduce contention for the redo copy latch. The redo copy latch is used to copy redo records from the PGA into the redo log buffer.

Redo Allocation – The redo allocation latch (allocates the space in the redo log buffer) contention can be reduced by using the NOLOGGING feature which will reduce the load on the redo log buffer. You should also try to avoid unnecessary commits.
**Row Cache Objects** - The “row cache objects” latch contention usually means that there is contention in the data dictionary. This may also be a symptom of excessive parsing of SQL statements that depend on public synonyms. Increasing the Shared Pool usually solves this latch problem. You usually increase the shared pool for a library cache latch problem well before this one is a problem.

**Cache Buffers Chains** - Buffers get “hashed to a chain” which means that several buffers can end up on the same chain of buffers going through the same latch. The ‘cache buffers chains’ latches are used to protect a buffer list in the buffer cache. These latches are used when searching for, adding or removing a buffer from the buffer cache. Contention indicates a ‘hot block’ or bad setting for _db_block_hash_buckets prior to 9i. The “cache buffers chains” latch is needed to scan the SGA buffer cache for database cache buffers. Hot blocks (often accessed blocks) in the buffer cache causes “cache buffers chains” latch issues. Hot blocks may also be a symptom of poorly tuned SQL statements. A hot record creates a hot block that can cause issues for other records inside that block as well as any block “hashed” to the same chain. To find the hot block, query v$session_children for the address and join it to v$sbh to identify the blocks protected by this latch (this will show all blocks that are affected by the hot block). You can identify the object by querying DBA_EXTENTS based on the file# and dbablk found from v$sbh. Using a reverse key index, if the hot block is on an index, will move sequential records to others blocks so that locked up by they are not locked up by the hot block in the chain. If the hot block is the index root block, a reverse-key index won’t help. Setting _DB_BLOCK_HASH_BUCKETS to twice the number of buffers (DB_CACHE_SIZE/DB_BLOCK_SIZE) and then up to the next prime number larger than that will usually eliminate this problem. Prior to Oracle9i, this parameter had a default that caused tremendous contention for this latch; the default is correctly set to a prime number in Oracle9i.

**Cache Buffers LRU Chain** - The “cache buffers lru chain” latch is used to scan the LRU (least recently used) chain containing all of the blocks in the buffer cache. A small buffer cache, excessive buffer cache throughput, many cache based sorts, and DBWR not keeping up with the workload are all culprits that can cause this issue. Try to fix the queries that are causing the excessive logical reads. You can increase the initialization parameter DB_BLOCK_LRU_LATCHES to have multiple LRU latches will reduce contention. Generally, non-SMP (symmetric multi processor) machines only need a single LRU latch. Oracle automatically sets this to ½ the number of CPUs on SMP machines. You must have at least on LRU latch for each database writer, make sure that you increase this if you add database writers.

**Latch Problem**

<table>
<thead>
<tr>
<th>Library Cache</th>
<th>Potential Fix</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use bind variables; adjust the shared_pool_size</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Shared Pool</th>
<th>Potential Fix</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use bind variables; adjust the shared_pool_size</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Redo allocation</th>
<th>Potential Fix</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimize redo generation and avoid unnecessary commits</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Redo copy</th>
<th>Potential Fix</th>
</tr>
</thead>
<tbody>
<tr>
<td>Increase the _log_simultaneous_copies</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Row cache objects</th>
<th>Potential Fix</th>
</tr>
</thead>
<tbody>
<tr>
<td>Increase the Shared Pool</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Cache buffers chain</th>
<th>Potential Fix</th>
</tr>
</thead>
<tbody>
<tr>
<td>_DB_BLOCK_HASH_BUCKETS needs to be increased or prime</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Cache buffers lru chain</th>
<th>Potential Fix</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set DB_BLOCK_LRU_LATCHES or use multiple buffer pools</td>
<td></td>
</tr>
</tbody>
</table>

Any of the latches that have a hit ratio below 99% should be investigated. Some of the more common latches on the problem list were detailed in this article and include the cache buffers chains, redo copy, library cache, and the cache buffers lru chain.

**BASIC TUNING REVIEW:**

First, you want to review the basics and try to tune everything without resorting to doing ANYTHING at the block level. This includes the usually suspects (summarizing the wait section about into a shorter list)

- Tune the SQL, of course... especially reads of full indexes tables and others.
- Find the hot blocks... hot blocks can cause latching issues. Bad SQL or bad indexes causes hot blocks (scanning through the same large index).
- Do you have enough freelists or can you use ASSM.
- Do you have initrans (ITL slots) set high enough for multiple DML to the same block (pctfree not high enough to auto-generate more ITL slots). Or do you have too many (each ITL costs 24 bytes!).
- Do you have a slow I/O subsystem or poor disk caching or not enough paths and readers/writers are colliding.
- Are you on latest version so can use all the great new features!

**DIVING DEEP TO YOUR LAST RESORT - BLOCK DUMPS**

If you’ve now exhausted the basics and even the intermediate steps of tuning, you’re now ready to go to the next step of looking at things at the block level. The first part to tuning at the block level is understanding how to do block dumps and what’s inside them.

Consider the following example (A copy of the famous EMP table):

```sql
SQL> desc emp1
Name Null? Type
EMPNO NUMBER(4) 
ENAME VARCHAR2(10) 
JOB VARCHAR2(9) 
MGR NUMBER(4) 
HIREDATE DATE 
SAL NUMBER(7,2) 
COMM NUMBER(7,2) 
DEPTNO NUMBER(2) 

select * from emp1 where ename = 'MILLER';
EMPNO ENAME JOB MGR HIREDATE SAL COMM DEPTNO
7934 MILLER CLERK 7782 23-JAN-82 1300 100

Find the blocks for this table:
select file_id, block_id, blocks
from dba_extents
where segment_name = 'EMP'
and owner = 'SCOTT';

FILE_ID BLOCK_ID BLOCKS
1 50465 3

Now dump the three blocks:
ALTER SYSTEM DUMP DATAFILE 5 BLOCK 50465
/ 
ALTER SYSTEM DUMP DATAFILE 5 BLOCK 50466
```
Now go get the block number for EVERY record of information:
select rowid, empno, dbms_rowid.rowid_relative_fno(rowid) fileno,
dbms_rowid.rowid_block_number(rowid) blockno, 
dbms_rowid.rowid_row_number(rowid) rowno, rownum, 
   rpad(to_char(dbms_rowid.rowid_block_number(rowid), 'FM0xxxxxxx') || '.' ||
   to_char(dbms_rowid.rowid_row_number(rowid), 'FM0xxx'), 18) myrid from emp1;

Most of the information found in block dump can be found in the 
data dictionary, or can be accessed using a built-in package such 
as dbms_space. However, there are certain scenarios in which 
knowing how to read a block dump might benefit you such as: 
To determine exactly why a transaction is blocked. You will probably 
used other tools prior to dumping a block, like utllockt.sql or EM 
(Enterprise Manager), but if you want to see exactly what is holding 
lock on a row in a block, and how many rows are blocked, the 
block dump output can be quite useful. You may also want to look 
at row chaining or to look at the space utilization in the block for 
each row or simply to look at the block because a block is corrupted 
and you want to take a closer look at it.

Items to note within the block dump:
One of the key sections of a block dump is the interested 
transaction list (ITL). The ITL is shown below and is in the early part of 
the dump. This one shows TWO ITL slots (two is the MINIMUM number of 
ITL slots for both Tables and Index (if you don’t believe what you 
read – you can dump it yourself to make sure). The XID is the 
Transaction ID. The UBA is the Undo Block Address, the Flag I’ll 
discuss in a moment, the lock shows the number of records 
locked (4 records are locked in the first ITL slot since I deleted 4 
rows for this example) and the SCN/FSC is either the SCN for 
committed information (Flag is a C) or FSC (Free Space Credit) is 
the amount of bytes that will be recovered within the block if the 
transaction is committed. This number is in hex. For our example 
it is 9d, which is 157 bytes recovered if the 4 deleted records 
transaction is committed.

--- The transaction is active, or committed pending block 

--- The transaction has been committed and the row locks 
have been cleaned out

-B--- The Undo Block Address contains undo for this block

--U- The transaction is committed (the SCN is the upper 
bound), but block cleanout has not occurred.(Fast commit)

---T This transaction was still active at when the SCN for 
block cleanout was recorded.

-C-U- The block was cleaned by delayed block cleanout, and 
the rollback segment information has been overwritten. The 
scn will show the lowest scn that could be regenerated by 
the rollback segment.

Reading the DUMP OUTPUT - EMPNO:

Hex to Decimal: Co10 = EMPNO = 7934
50 (Hex) = 80 (Decimal) – 1 = 79
23 (Hex) = 35 (Decimal) – 1 = 34
c2: Number in the thousands (c2 is exponent)
Col 4: [7] 77 b6 01 17 01 01 01
Hex to Decimal: Col 4 = HIREDATE = 23-JAN-82
77 (Hex) = 119 (Decimal) – 100 = 19
B6 (Hex) = 182 (Decimal) – 100 = 82
01(Dec) = 1 (Decimal)<month>
17 (Dec) = 23 (Decimal)<day>
01 01 01 (Hex) = This is the Hour, Minute, Second
(none were entered when the date was entered...default)

Block Dump using SELECT dump():
select dump(ename) from emp1
where ename='MILLER';

DUMP(ENAME)

<table>
<thead>
<tr>
<th>Typ</th>
<th>Len</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>6</td>
<td>77,73,76,76,69,82</td>
</tr>
<tr>
<td>2</td>
<td>6</td>
<td>MILLER</td>
</tr>
</tbody>
</table>

Block Dump using SELECT dump() and convert it to HEX:
select dump(ename.16)
from emp1
where ename='MILLER';

DUMP(ENAME,16)

<table>
<thead>
<tr>
<th>Typ</th>
<th>Len</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>6</td>
<td>4d,49,4c,4c,45,52</td>
</tr>
</tbody>
</table>

Now that we know that all EMP1 data is in Block#: 56650 -
let's dump all the rows:
select rowid,empno,
    dbms_rowid.rowid_relative_fno(rowid) fileno,
    dbms_rowid.rowid_block_number(rowid) blockno,
    dbms_rowid.rowid_row_number(rowid) rowno, rownum,
    rpad(to_char(dbms_rowid.rowid_block_number(rowid), 'FM0xxxxxx') || '.
    to_char(dbms_rowid.rowid_row_number
    to_char(dbms_rowid.rowid_relative_fno(rowid), 'FM0xxx'), 18) myrid
from emp1;

<table>
<thead>
<tr>
<th>ROWID</th>
<th>EMPNO</th>
<th>FILENO</th>
<th>BLOCKNO</th>
<th>ROWNO</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Let's query the block and watch the EMP1 buffer header:
(For so far its clean and only 1 copy)
select lbda_seq, state, dbarfil, dbablk, tch, flag, hscn_bas,cr_scn_bas,
    decode(bitand(flag,1), 0, 'N', 'Y') dirty, /* Dirty bit */
    decode(bitand(flag,16), 0, 'N', 'Y') temp, /* temporary bit */
    decode(bitand(flag,1536), 0, 'N', 'Y') ping, /* ping to shared or
    null bit */
    decode(bitand(flag,16384), 0, 'N', 'Y') stale, /* stale bit */
    decode(bitand(flag,65536), 0, 'N', 'Y') direct, /* direct access bit */
    decode(bitand(flag,1048576), 0, 'N', 'Y') new /* new bit */
from x$bh
where dbablk = 56650
order by dbablk;

Let's watch the EMP1 buffer header when we delete a row
(two copies of the block):
delete from emp1
where comm = 0;
one row deleted.
select lbda_seq, state, dbarfil, dbablk, tch, flag, hscn_bas,cr_scn_bas,
    decode(bitand(flag,1), 0, 'N', 'Y') dirty, /* Dirty bit */
    decode(bitand(flag,16), 0, 'N', 'Y') temp, /* temporary bit */
    decode(bitand(flag,1536), 0, 'N', 'Y') ping, /* ping to shared or
    null bit */
    decode(bitand(flag,16384), 0, 'N', 'Y') stale, /* stale bit */
    decode(bitand(flag,65536), 0, 'N', 'Y') direct, /* direct access bit */
    decode(bitand(flag,1048576), 0, 'N', 'Y') new /* new bit */
from x$bh
where dbablk = 56650
order by dbablk;

Note that V$Transaction now has our record:
(created when transactions have undo)
SELECT t.addr, t.xidusn USN, t.xidslot SLOT, t.xidsqn SQL, t.status,
    t.used_ublk UBLK, t.used_urec UREC, t.log_io LOG,
    t.phy_io PHY, t.cr_get, t.cr_change CR_CHA
FROM v$transaction t, v$session s
WHERE t.addr = s.taddr;
- USN is the Undo Segment Number (rollback segment ID)
- SLOT is the slot number in the rollback segment's transaction table.
- SQN (Wrap) is the sequence number for the transaction.
- USN+SLOT+SQN are the three values that uniquely identifies a transaction XID
- UBAFIL is the file for last undo entry
- UBLK is block for last undo entry (find out how many undo blocks).
- UBASEQ is the sequence no of the last entry.
- UREC is the record number of the block (shows how many table and index entries the transaction has inserted, updated or deleted.

If you are doing an INSERT or DELETE, then you will see that UREC is set to <number of indexes for this table> + how many rows you inserts/deletes. If you UPDATE a column then UREC will be set to <number of indexes that his column belongs to> * 2 + number of updated rows (so if the column belongs to no index, then UREC is set to the number of rows that was updated). If USED_UBLK and USED_UREC are decreasing each time you query, then the transaction is rolling back. When USED_UREC zero, the rollback is finished.

If you Dump the block at this time (you see the locked record in the ITL section):

```
KSPPINM
KSPPSTVL
KSPPSTDF
KSPPDESC
_db_block_max_cr_dba
6
TRUE
```

Maximum Allowed Number of CR buffers per dba

What happens after we roll everything back – x$bh Still an LRBA (Lowest RBA):

```
KSPPINM
KSPPSTVL
KSPPSTDF
KSPPDESC
_db_block_max_cr_dba
6
TRUE
```

Now let's track some UNDO:

```
Create EMP2 ('MILLER'/ 'ALLEN'):
create table emp2
as select * from emp1
where ename in ('MILLER', 'ALLEN');
select empno, ename, job
from emp2;
```
**EMPNO  ENAME  JOB**
---
7499  ALLEN  SALESMAN
7934  MILLER  CLERK

**Get the Blockno for EMP2:**

```sql
select rowid,empno,
    dbms_rowid.rowid_relative_fno(rowid) fileno,
    dbms_rowid.rowid_block_number(rowid) blockno,
    dbms_rowid.rowid_row_number(rowid) rowno,
    rownum
from emp2;
```

**Dump the EMP2 block (Partial output only):**
Alter system dump datafile 2 block 57810; System Altered.

**Update 'MILLER' to 'SMALL':**

```sql
update emp2
set ename = 'SMALL'
where ename = 'MILLER';
```

**Dump the EMP2 block again (Partial Output):**
Alter system dump datafile 2 block 57810; System Altered.

**Find the Segment & Location of Undo Block:**

```sql
SELECT DBMS_UTILITY.DATA_BLOCK_ADDRESS_FILE(
    TO_NUMBER('00800353','XXXXXXXX')) UFILE
FROM DUAL;
```

```
UFILE
----------
2
```

```
SELECT DBMS_UTILITY.DATA_BLOCK_ADDRESS_BLOCK(
    TO_NUMBER('00800353','XXXXXXXX')) BLOCK
FROM DUAL;
```

```
BLOCK
----------
851
```

After system dump datafile 2 block 851; System altered.
Dump the UNDO Block (Partial Output):

UNDO BLK:

xid: 0x0004.02a.000012ff seq: 0xa9e cnt: 0x7 icl: 0x0 flg: 0x000

Rec #0x7 slt: 0x2a objn: 53366(0x0000d076) objd: 53366 tblspc: ...
uba: 0x00800353.0a9e.04 ctl max scn: 0x0000.0432656 prv tx
scn: 0x0000.0432656
txn start scn: 0x0000.0432731 logon user: 0 prev brb: 8389454 prev bcl: 0
KDO undo record:
KTB Redo
op: 0x03 ver: 0x01 op: Z
KDO Op code: URP row dependencies Disabled
xtype: XA flags: 0x00000000 bdba: 0x00040e1d2 hdba: 0x0040e1d1
itli: 2 ispac: 0 maxfr: 4863
tabn: 0 slot: 10x1 flag: 0x2c lock: 0 cki: 0
nco: 8 nnew: 1 size: 1
col 1: [ 6] 4d 49 4c 4c 45 52 Here's the UNDO: M I L L E R

Let's check V$TRANSACTION after 2 updates & 14 rows deleted - Match ITL (no need to dump):

<table>
<thead>
<tr>
<th>XIDUSN</th>
<th>XIDSLOT</th>
<th>XIDSQN</th>
<th>UBAFIL</th>
<th>UBABLK</th>
<th>UBASQN</th>
<th>UBAREC</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>42</td>
<td>4863</td>
<td>2</td>
<td>851</td>
<td>2718</td>
<td>8</td>
</tr>
<tr>
<td>5</td>
<td>14</td>
<td>667</td>
<td>2</td>
<td>1458</td>
<td>713</td>
<td>25</td>
</tr>
</tbody>
</table>

4.42.4863 = 4.2a.12ff 2.851.2718.8 = 800353.a9e.8
5.14.667 = 5.e.29b 2.1458.713.25 = 8005b2.2c9.19

XIDUSN XIDSLOT XIDSQN UBAFIL UBABLK UBASQN UBAREC
-------- -------- -------- -------- -------- -------- --------
| 8       | 4        | 42      | 4863   | 2      | 851    | 2718   | 8      |
| 5       | 14      | 667    | 2      | 1458   | 713    | 25     |

4.42.4863 = 4.2a.12ff 2.851.2718.8 = 800353.a9e.8
5.14.667 = 5.e.29b 2.1458.713.25 = 8005b2.2c9.19

Why no Change (show uncommitted)?? Delayed Block Cleanout!
(Usually fast commit)

Delayed block cleanout...
Select * from emp2;
(delayed block cleanout is how redo can be generated from a select)
Alter system dump datafile 1 block 57810;
System altered.

<table>
<thead>
<tr>
<th>Tbln</th>
<th>Xid</th>
<th>Uba</th>
<th>Flag</th>
<th>Lock</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x01</td>
<td>0xffff.000.00000000</td>
<td>0x00000000.0000.00</td>
<td>c---</td>
<td>0 fsc</td>
</tr>
<tr>
<td>0x02</td>
<td>0x0004.02a.000012ff</td>
<td>0x00000000.0000.00</td>
<td>c---</td>
<td>0 fsc</td>
</tr>
<tr>
<td>0x03</td>
<td>0x0005.00e.000002ea</td>
<td>0x00000000.0000.00</td>
<td>c---</td>
<td>0 fsc</td>
</tr>
</tbody>
</table>

All records now show as committed (Note Flag is C).

Delayed block cleanout:
If a dirty block has already been written to disk (could also be due to locking or when many blocks are changed), then the next process to visit the block will automatically check the transaction entry in the undo segment header and find the changes made to the block have been committed. The process gets the SCN of the commit from the undo header transaction entry and writes it to the data block header to record the change as committed.

A deeper dive into block tuning is beyond the scope of this book. But, this section should show you some queries that will allow you to investigate what is going on at the block level in the VERY rare case that you need to see it. The best reason to do block dumps is to see what's going on inside of Oracle.

SPECIAL THANKS TO:
Janet Bacon, Brad Brown, Joe Trezzo, Randy Swanson, Burk Sherva, Tony Catalano and the TUSC Team who have all made contributions to this document.

REFERENCES:
Oracle10g Performance Tuning Tips & Techniques, Rich Niemiec
The Machinations of Oracle, Scott Marin, Terlingua software
All about Oracle's touch count algorithm, Craig Schallhammer
Kevin Gilpin, Mark Bobak, Jonathon Lewis, Metalink Notes EM Grid Control 10g; otn.oracle.com, Oracle Corporation
Oracle Enterprise Manager 10g: Making the Grid a Reality; Jay Rossiter, Oracle Corporation
Oracle 10g & Oracle 9i documentation
www.tusc.com
www.oracle.com
www.ixora.com
www.laoug.org
www.ioug.org
www.technet.oracle.com
Scaling Oracle8i, James Morle
Reading a block dump, Janet Bacon, TUSC
www.tusc.com
www.oracle.com
www.ixora.com.au
www.jlcomp.demon.co.uk/
www.laoug.org, www.ioug.org,
www.technet.oracle.com
ABOUT THE AUTHOR:
Rich Niemiec is the CEO of The Ultimate Software Consultants (TUSC), a Lombard, Illinois based Oracle centric consulting company. He is the author of the Oracle Press book Oracle Database 10g Performance Tuning Tips and Techniques. TUSC is an Inc. 500 company that specializes in the full cycle of database development including Business Modeling, Design, Development, Oracle Applications, Implementation, Database Administration and DBA Remote Services. Rich has been giving lectures and presentations on Oracle for the past 20 years and is the past President of the IOUG and the current President Midwest Oracle Users Group (MOUG). Rich can be reached at TUSC at (630) 960-2909, rich@tusc.com or visit our web site for more papers at www.tusc.com. Please report errors in this article to rich@tusc.com. Neither TUSC nor the author warrants that this document is error-free. TUSC © 2007. This document cannot be reproduced without expressed written consent from an officer of TUSC except RMOUG may make copies and make this paper available as needed for the RMOUG Newsletter.
ABSTRACT
In June 1970, IBM’s Ted Codd published the 11 page paper “A Relational Model of Data for Large Shared Data Banks.” This article would lead to the relational databases that would run the world. With 64-bit processing and using grid control, it is theoretically possible to store all information currently in every Fortune 500 database into a single Oracle10g database (Oracle10g allows an 8E database) and you could load all of it in memory (64 bit allows 16E), if you had a LOT of hardware. The advances of the last decade will be dwarfed by those of the next 10 years. Welcome to the 21st century DBA and Generation-64! In 11g, there are many self tuning features as well as Oracle tools to help you tune things to access less data. Utilizing the tools Oracle gives you to tune things is paramount to managing greater amounts of data in a more efficient manner. In this article, I want to focus on one simple and quick new index called the invisible index that is new in 11g.

The Invisible Index
An invisible index is an index that continues to be maintained as if it was visible, yet to someone who runs a query, the index is not considered by the optimizer unless specifically hinted. Some of the ways you can set an index:

- ALTER INDEX idx INVISIBLE;
- ALTER INDEX idx VISIBLE;
- CREATE INDEX... INVISIBLE;

This is a great testing tool to turn off indexes for a while when you think they’re not being used, but BEFORE you drop them. You can use the INDEX hint to override invisibility, or the NO_INDEX hint to override the use of the index when it is visible. A very important point to remember about an invisible index:

- The invisible index IS MAINTAINED during DML

Creating the Invisible Index
Here’s an example of how to create an invisible index called dept_rich_inv_idx on the deptno column of the dept_rich table:

create index dept_rich_inv_idx on dept_rich(deptno) invisible;
Index created.

The Index is not used since it’s Invisible
Now let’s try a query that would USUALLY use the index if it was visible and we see that the query does a full table scan:

select count(*)
from dept_rich
where deptno = 30; (the index is not used since it’s invisible)

```
COUNT(*)
---------
512
```

Execution Plan
```
<table>
<thead>
<tr>
<th>Id</th>
<th>Operation</th>
<th>Name</th>
<th>Rows</th>
<th>Bytes</th>
<th>Cost (%CPU)</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>SELECT STATEMENT</td>
<td></td>
<td>1</td>
<td>2</td>
<td>4 (0)</td>
<td>0:00:01</td>
</tr>
<tr>
<td>1</td>
<td>SORT AGGREGATE</td>
<td></td>
<td>1</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>* 2</td>
<td>TABLE ACCESS FULL</td>
<td>DEPT_RICH</td>
<td>512</td>
<td>1024</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```
The Index is used without the need of a hint when it is VISIBLE:
If we then alter the index to be visible, we no longer have to use a hint. The optimizer uses the index:

```
alter index dept_rich_inv_idx visible;
Index altered.
```

```
select count(*)
from   dept_rich
where  deptno = 30;
```

```
512
```

.Execution Plan
```
| Id  | Operation         | Name              | Rows  | Bytes | Cost
|-----|-------------------|-------------------|-------|-------|------
|     | SELECT STATEMENT  |                   | 1     | 2     | 1    |
|     | SORT AGGREGATE    |                   | 1     | 2     |      |
|*   | INDEX RANGE SCAN  | DEPT_RICH_INV_IDX | 512   |       |      |
```

Now that the INDEX is VISIBLE, the index is used by the optimizer as seen in the explain plan above.

The Visibility of an Index can checked in the database:
We can query DBA_INDEXES to check an index.

```
alter index dept_rich_inv_idx invisible;
Index altered.
```

```
select  index_name, visibility
from dba_indexes  (or go to USER_INDEXES)
where   index_name = 'DEPT_RICH_INV_IDX';
```

```
INDEX_NAME             VISIBILITY
---------------------- ------------------
DEPT_RICH_INV_IDX      INVISIBLE
```

The invisible index is a great tuning and testing tool. Make sure you try it out when testing 11g.

References:
Collaborate 2007, Oracle11g Best New Features, Rich Niemiec, 2007
Oracle Database 10g Performance Tuning Tips and Techniques; Niemiec, 2007

ABOUT THE AUTHOR:
Rich Niemiec is the author of the Oracle Press book Oracle10g Performance Tuning Tips and Techniques. He is also the CEO of TUSC, the Chicago based Oracle Expert database integrator (www.tusc.com). Rich can be reached at rich@tusc.com. Neither TUSC nor the author warrants that this document is error-free. TUSC © 2007. RMOUG is authorized to reproduce this text in any form.
Oracle Identity Management

Oracle—“database management conceived in the 20th century for use in the 21st” continuously and consistently grows more powerful with every new version. As the marketplace demands more efficient and effective ways of managing data, Oracle creates and upgrades its Application Server (processes, middleware, etc) to include new functionalities which respond to more needs for more types of organizations. What is even more important in the Post 9/11 world is that in 2005 Oracle obtained a powerful middleware called Oracle Identity Management. This fusion software facilitates:

- Stable and secure data processing
- Documenting regulatory compliance to ever increasing legislation governing how data may be used, i.e. the Sarbanes-Oxley Act
- Flexibility and environmental responsiveness
- An Increase in system integration
- Reducing the costs of managing data

Almost anyone who has ever used Oracle is aware of Oracle’s constant market responsiveness. While Oracle responds and upgrades, Oracle consumers (managers, DBAs, Developers, and other IT personnel) don’t always upgrade their understanding of the additional power tools offered. There can be a huge gap between what is available in recent versions of Oracle for example 10g and what a consumer who first learned Oracle 8i knows about the newer version’s capabilities. Many DBAs have a tendency to continue using what they already know. They fail to learn new parts of Oracle unless they have a “need to know” in order to accomplish a new task or solve a new kind of problem.

People who work with databases know that there is an ever widening gap between the available Oracle database solutions and the number of Oracle functions any one person can actually master and use effectively. Even some Oracle pundits will admit, no one can be an expert or even needs to use everything Oracle has to offer.

Sadly, too many organizations are missing out on Oracle capabilities they really need. Middleware such as Oracle Identity Management (IdM) may even be available on their system, yet this useful security management tool lies dormant. This happens for many reasons, the most common are:

- Lack of information or knowledge about IdM
- Belief that the Oracle version in use does everything needed (no matter how old it is)
- IT personnel don’t have the time or inclination to learn about new features
- IT personnel resist changes that reduce the need for specialized skills, i.e. coding
- Fear that additional features will greatly increase costs and/or down-time

The saddest case, however, is that decision makers in small, medium and even large companies are not as computer savvy as they need to be. They are not aware of what Oracle “tools” their system has. Worse they do not understand what these versatile tools can do for their organizations.

Oracle adds many advanced technological upgrades to each new version. Consumers, especially decision makers not directly involved with the day to day functioning of the computer system, may find out about only a few advances and have their IT people incorporate these in their system. Database managers, systems analysts, or developers may learn about new features at conferences such as RMOUG’s Training Days. Many other upgrades are ignored. Unfortunately, IT personnel may be hesitant to suggest technical upgrades to non-technical managers, especially those who refuse to try new ideas they don’t understand.

In reality, tools (upgrades, technological advances, fusion ware, etc.) don’t work! The people who use these tools are what work or do not work. How well technical personnel understand Oracle’s possibilities and how capable they are of utilizing Oracle Enterprise Suite (OES) will determine what results can be expected.

For example, a hammer is a very useful tool for building structures. However, if no one in a construction company knows how to properly use a hammer – it is doubtful that any:

- Sound and secure
- Code/regulation compliant
- Useful over a long time period
- Well designed and integrated
- Cost effective

Structure will result from the use of the company’s hammers.
To continue the metaphor of construction, what if the company also owned nail guns. These “quicker, better, faster” tools may not get used for several reasons:

- Management may refuse to pay for training or provide the time needed for employees to learn to use the new more efficient and effective tools correctly.

- Management does provide training, etc. and employees still refuse to use the new tools. There are a number of reasons this can happen, whether the tool is a nail gun or computer middleware.

- Even though employees were “trained,” they still don’t know how to load the nail gun or configure the middleware.

- There are people who are happy with the status quo i.e., using a nail gun is for “sissies” or learning how to use Oracle’s built-in flexible architecture with readily available code choices would make database and code writing experience less necessary and open the field by competition to cheaper employees.

Another resistance to changes that result from doing things quicker, better, faster by machine (nail gun) or flexible architecture is that this can reduce the number of people needed to build structures or develop and maintain databases.

Worse of all reasons for not using new ideas such as Oracle Identity Management (IdM), and worth repeating here, is no one or at least non-IT decision makers knows they already have this wonderful security and documentation tool. Ignorance is not bliss. Ignorance about the powerful fusion middleware OES contains causes procurement officers and decision makers to look for solutions to their database problems (security, regulatory compliance, efficiency, effectiveness, flexibility, and/or cost reduction) from sources other than Oracle.

Many DBA’s can tell you horror stories about trying to load “prefabricated—off the shelf” non-Oracle solutions onto their Oracle Applications Server. Only after installing these “foreign” problem solvers, do companies find out these non-Oracle add-ons have trouble communicating with the other parts of the computer system with which they need to interact. Although Oracle Application Server is renowned for its ability to play well with others, “foreign add-ons” are “bolted on,” whereas IdM, to paraphrase Oracle’s Data Sheet, is application-centric. IdM weaves integrated solutions into applications as opposed to “bolting on” solutions. This “fusion” or weaving promises “unprecedented efficiencies and ROI.” “Bolted on” solutions are often as useful to company profitability as the Edsel’s added chrome was to Ford’s profitability in the 1950s.

Oracle middleware is deceptively undemanding. The IdM portion of this fusion ware can be installed in a relatively short time as long as management access policies have been translated into definable and rational ideas prior to installation. In very simplistic terms, IdM allows organizational decision makers to decide who gets access to what, and where they can “go” in the computer system. It also permits decision makers to create elaborate, yet elegant restraints on how data, information, or any other part of the system can be interacted with by any and all users in ways that are unique to each user’s specific needs.

The restraint functions within IdM allow management to more effectively implement internal security policies which eliminate potential threats from expired and unauthorized users. Allowing decision makers the power to decide who, what, when, where and how the computer system can be accessed provides a twofold benefit for management. First, it enables access to only authorized users. Second, it acts as a guardian of the organization’s own proprietary information against unauthorized use by both insiders and outsiders.

Many companies purchase elaborate security software and hardware to keep intruders out of there system. Some ready-made security software will actually take over the computer system with measures so extreme that even bona fide users can no longer work efficiently. IdM proves to be a much more elegant and useful security tool than what many organizations are now using. At the very least, it compliments and strengthens any present security measures.

Moreover, IdM is a cost-effective way to enforce requirements associated with identifying the users in an organization who should have access to privileged and or sensitive information. Laws such as the Sarbanes-Oxley Act of 2002, PCI Security Standards and the Gramm-Leach-Bailey Act are all designed to protect personal and private
information from fraud or misuse by others. The Gramm-Leach-Bailey Act creates regulations to safeguard specific consumer data i.e.; name, address, social security number, bank account numbers, credit card numbers, etc. The Act’s regulations are enforced by the Federal Trade Commission. IdM when integrated into Oracle Application Server has the power to permit only trusted, bonded, or specifically designated users access to this or any other private or proprietary information. It can also be configured to record or document who accessed what information about whom, when information was accessed, why it was accessed, and how it was used with minimal system resources.

Documenting that an organization is complying with regulations concerning the use of information is the major method legislative regulations employ. Organizations waste a lot of time and money proving compliance if their system isn’t recording the information regulators demand automatically. IdM will also document who accessed the company’s own proprietary information in the same way, thus safeguarding the company’s privileged information, i.e.; strategic planning, upcoming bids, salary information, etc.

Quick responsiveness is an important key to success. When users have immediate access to the applications and system tools they need, internal users can be more productive and external users can get quicker, more effective responses to their needs. Once IdM is in place, Single Sign-On (SSO) becomes available to all users.

Single Sign-On (SSO) means that a system user only needs to identify themselves (log-on) one time in order to access all the system’s features or applications for which their pre-set characteristics allow them permission.

SSO can be maintained and allows users in the organizations to improve their IT operations by automating many user administrative tasks. By providing administrative access as close to the user as possible, an enterprise has tighter controls and better security, while productivity is increased. IdM has the capability for users to manage their own passwords. If a user forgets his/her password, IdM will present a number of user preset questions to enable self service verification by the user to retrieve a password. This reduces the need for help desk assistance.

SSO has historically been available to intranet users; organizational employees, members, or insiders whose task definitions and application needs are well defined within the organization’s intranet. Single Sign-On capabilities for internet users which allow outsiders to define and maintain their own access profile information such as “user name” and/or “password” over the internet is relatively new.

More and more organizations use web based systems to communicate with internal and external users (customers, supplies, regulators, etc.). SSO has become an instant efficiency and effectiveness tool saving millions of man hours monthly. It is estimated that up to ninety percent of help desk inquiries concern the user’s failure to remember their own sign on information. The resulting man hour reduction from employing SSO not only translates into increased efficiency, but also cost reduction and increases the ROI of installing IdM.

When management hears the IT department wants to upgrade their system, panic is their usual response. Upgrades can require their systems to be down for days with the corresponding loss in productivity. IdM is an integrated solution and can be deployed quickly for most common operating systems like UNIX, LINUX, and Windows. In the Windows environment, Identity Management integrates easily with Active Directory. In the UNIX and LINUX environment, Oracle Internet Directory and Security Manager will integrate into the system easily and realize gains immediately. Common security infrastructure is a feature within IdM which provides a library of pre-configured connections to get a head start on integrating most popular applications into the system through user repositories and technologies. Oracle’s flexible architecture facilitates fusion ware such as IdM to integrate into the system quickly thereby reducing the cost of deploying and maintaining these secure, efficient, cost reducing solutions.

IdM includes an Adapter Factory technology that eliminates the complexity associated with integrating and maintaining the connectors to most standard software. This will assist in rapid integration into both custom and commercial systems. DBAs or Developers can create new or modify existing integrations using the Adapter Factory’s graphical user interface, without programming or scripting. When connectors are created, their definitions are maintained in the repository of IdM thus creating a map or view—documentation of how integration was accomplished. Views make upgrading, extending and maintaining connections to common software a manageable and straightforward process as they allow integrations to be replicated by others without having to “quiz” the original DBA or Developer on how they achieved integration. Views make further upgrading and maintaining connections to common software a straightforward and manageable process.

**Identity Database**

OID and Identity constant communication
User logs on to database
Admin sets what access users have
Admin Work Station
Oracle Internet Directory
User can perform his/her duties user has access to
User Wk station
Operational Database
ws

**Database checks user access**

Identity database gives user permission to certain fields in the operational database
The diagram above illustrates a typical configuration of IdM on UNIX and or LINUX operating system. This configuration automates most usual connections thereby using very little system down time.

The key to achieving
- Stable and secure data processing
- Documenting regulatory compliance to ever increasing legislation governing how data may be used
- Flexibility and environmental responsiveness
- An Increase in system integration
- Reducing the costs of managing data

is to carefully translate access policies into clear, unique, and definable categories prior to installation. Two sets of these clearly unique and definable user access categories will be needed: one, for intranet users and the second for internet users as they are stored separately to eliminate any cross-over between access categories.

When IT personnel whether DBA's, software developers, or system administrators do a good job of access definition as stated above IdM's fusion middleware and Oracle Flexible Architecture along with the built-in connectors contained in the Oracle's Application Software should make gaining the five advantages of IdM for a computer system. This is true even if the system contains a massive database. IdM is the way to make sure the system is secure, documents regulatory compliance, uses fewer man hours to access and complete both intranet and internet administrative tasks thereby reducing costs and increasing the organization's ROI.

ABOUT THE AUTHOR:
Thomas Green holds a B.S. in Computer Science for Regis University and has been an RMOUG member since 2000 and a volunteer at Training Days since 2002. Tom has worked with computers for many years. For the past eight years he has been employed in a high security environment creating, maintaining and upgrading database as large as 18 terabytes.

In his day job Tom continues to gain valuable experience as an Oracle Problem Researcher. In his spare time he challenging himself through Oracle OTR and Metalink to research and learn more about Oracle Security Middle Ware. He can be reached after 3:00 PM most weekdays at 303-517-5169.
It is important when tuning SQL to have the most information available when reviewing explain plans. These tips will hopefully help you with the basics of viewing explain plans. I will also include a couple of tips on where to start your tuning efforts.

The most simple way to get an explain plan is to trip on AUTOTRACE from your SQL*Plus environment. There are a few options to this command. I usually issue SET AUTOTRACE ON EXPLAIN. This command will give you the output of your SQL as well as an explain plan. Example 1 will show our sample SQL for this paper (from my Tuning workshops) and Example 2 will show the output from this AUTOTRACE command.

Example 1: Syntax and Sample SQL

```sql
SQL> SET AUTOTRACE ON EXPLAIN
SQL> 1 SELECT count(*)
2    FROM B, C, A
3    WHERE A.STATUS = B.STATUS
4        AND A.B_ID = B.ID
5    AND B.STATUS = 'OPEN'
6    AND B.ID = C.B_ID
7*    AND C.STATUS = 'OPEN'
SQL>
```

Example 2: Output from SQL and AUTOTRACE

Tip: Issue SET TIMING ON will add a clock timing to the output!

Sometimes (usually) you want to get an explain plan for a long-running SQL without running the SQL. This too is easily accomplished. Use the EXPLAIN PLAN FOR syntax as indicated in Example 3 and this will only produce an explain plan for you. This method requires the use of a script (I use my own SHOW_PLAN.sql script that is downloadable from my website).

```
rem
rem  Dan Hotka Pinnacle Professional article
rem  SHOW_PLAN.sql   - used to print contents from Plan_Table
rem
```
rem Install utlxplan.sql for each user, then put this line in front of sql statement
rem --> explain plan set statement_id = '<some name>' for <SQL statement>
rem --> run this script --> sqlplus userid/pwd @show_plan <some name>
rem
set pagesize 20
set linesize 80
column id format 999 heading 'ID'
column parent_id format 999 heading 'P_ID'
column cost format 999 heading 'Cost'
column access_plan format a30 heading 'Access|Plan'
column access_path format a15 heading 'Access|Path'
column object_name format a15 heading 'Object|Name'
select cost,id, parent_id, lpad (' ', 2 * level) || operation Access_Plan, options Access_Path, object_name from plan_table where statement_id = '&1'
connect by prior id = parent_id
start with id = 0;
delete from plan_table where statement_id = '&1';

Example 3: SHOW_PLAN.sql Script

Issue the command 'explain plan set statement_id = '<some name>' for <SQL statement>; then run this SHOW_PLAN script. Notice the script will ask for the STATEMENT ID given in the explain plan command. This allows you to save explain plans from multiple runs.

Example 4: Getting just an explain plan

This method gives a nicer looking explain plan and also allows you to easily include other information in the output from the PLAN_TABLE.

Oracle8i has included a new method of getting explain plans. It uses the same EXPLAIN PLAN FOR syntax but the output is considerably different. Capture the explain plan information as in Example 4. Remember to delete from PLAN_TABLE when done! SHOW_PLAN.sql does this cleanup for you. Example 5 shows the EXPLAIN PLAN syntax for our example SQL.

Example 5: Explain Plan Capture

Notice below the new syntax! The DBMS_XPLAN.DISPLAY function gives some nice output, showing the WHERE clause predicates that goes with the explain plan. This information answers the question of what part of the SQL caused this step to happen. Code like this takes all the guesswork out of tuning SQL statements.

Example 6: New Explain Plan Output

The last couple of examples are from my JSTuner tool (again, downloadable from my website). This tool combines all of the information discussed so far into a simple interface. Notice that Example 6 does not work for any version of Oracle8, however, JSTuner will capture this same information for Oracle8 databases as well!

Example 7 shows the AUTOTRACE output nicely associated in the tabs OUTPUT, Explan Plan, and Statistics. These panes are populated from the ‘Run’ button. Example 8
shows the WHERE clause predicates inline with the SQL! This is a version of my SHOW_PLAN.sql script combined with the additional WHERE clause syntax! This pane is populated from either the ‘Run’ button (runs the SQL) or the ‘Explain’ button (just captures the explain plan without running the SQL). JSTuner also captures all the history, has table, index, and hits available and allows you to read in and save SQL to/from operating system files. Again, this tool can be downloaded from my website: www.DanHotka.com.

Example 7: JSTuner AutoTrace

Example 8: JSTuner Enhanced Explain Plan

As you can see, JSTuner combines the power of these command line explain plans and produces an easier-to-read explain plan to aid you in your SQL tuning needs.

I hope these explain plan tips help you in your day-to-day work with the Oracle RDBMS.

ABOUT THE AUTHOR:
Dan Hotka is a Training Specialist who has over 30 years in the computer industry and over 25 years of experience with Oracle products. He is an internationally recognized Oracle expert with Oracle experience dating back to the Oracle V4.0 days. Dan’s latest book is the SQL Developer Handbook by Oracle Press. He is also the author of Oracle9i Development By Example and Oracle8i from Scratch by Que and has co-authored 7 other popular books including the Database Oracle10g Linux Administration by Oracle Press. He is frequently published in Oracle trade journals, and regularly speaks at Oracle conferences and user groups around the world. Visit his website at www.DanHotka.com. Dan can be reached at dhotka@earthlink.net.

Dan Hotka - Author/Instructor/Expert
www.DanHotka.com
Dan.Hotka@DanHotka.com

Onsite Oracle Training

Single course fee for up to 15 attendees

Fee includes portable computer lab Course Guides priced per student

Oracle Courses in:
SQL Performance Tuning
Advanced PL/SQL
TOAD Tips and Techniques
- Oracle BI Courses
- Discoverer for End Users
- Analytics for End Users
Discoverer for Administrators
Oracle Intro and Unix courses too

Contact Dan at:
www.DanHotka.com
DHotka@Earthlink.net
(515) 279-3361
## Upcoming RMOUG Events Calendar

<table>
<thead>
<tr>
<th>Month</th>
<th>Event Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>June</td>
<td>ODTUG Kalidescope, see <a href="http://www.odtugkaleidoscope.com/">http://www.odtugkaleidoscope.com/</a> for details</td>
</tr>
<tr>
<td>July</td>
<td>Board Positions to be assigned</td>
</tr>
<tr>
<td>August</td>
<td>Board Meeting</td>
</tr>
<tr>
<td>August</td>
<td>QEW (tentative date, may be rescheduled to August 22)</td>
</tr>
<tr>
<td>September</td>
<td>Board Meeting</td>
</tr>
<tr>
<td>October</td>
<td>Board Meeting</td>
</tr>
<tr>
<td>November</td>
<td>QEW</td>
</tr>
</tbody>
</table>

### Ongoing Calendar Items:
COAUG meets the 3rd Tuesday of each month check website for details [http://www.coaug.org/](http://www.coaug.org/)

---

**RMOUG** would like to offer the opportunity for members to advertise Oracle technology related services, consulting, products and training in the upcoming issue of RMOUG’s SQL>Update. For only $35.00 you can purchase a 2½ X 3 inch black and white business card size space for a single issue of SQL>Update.

If interested, you will need to submit a pre-formatted/pre-edited copy of your business card ad or announcement no later than **July 31, 2008** for the November issue. Please feel free to contact me directly if I can help out in any way.

Allison Leech  
Newsletter Director  
RMOUG  
303.868.5835  
allison_leech@yahoo.com
Membership Application

Type of Membership and Annual Dues (please check one). Membership is good for one year from date payment is processed. The following rates are in effect for the 2008 membership year (until further notice):

- Individual...$75
- Student (must have Student ID) $35
- Corporate:
  - 1-5 Members ................................................................. $300
  - 6-8 Members ................................................................. $450
  - 9-12 Members ............................................................. $595
  Additional members (over 12) $50 each

Corporate Members: (Use a separate sheet of paper for additional names)

Please list corporate contact first.

Corporate Contact Name ___________________ Phone _________________
Address _____________________________________________
City, State, Zip _________________________________________
e-mail _______________________________________________

Name _________________________ Phone _________________
Address _____________________________________________
City, State, Zip _________________________________________
e-mail _______________________________________________

Name _________________________ Phone _________________
Address _____________________________________________
City, State, Zip _________________________________________
e-mail _______________________________________________

Please mail the completed form and payment to:
Rocky Mountain Oracle Users Group
P.O. Box 621942
Littleton, CO 80162
Fax: (801) 697-4366

- The Rocky Mountain Oracle Users Group (RMOUG) is over 20 years old and currently consists of over 1200 members throughout the Rocky Mountain Region. RMOUG is incorporated in Colorado.
- RMOUG has three general membership meetings a year and one major training event. General meetings are scheduled for a full day of presentations by users, vendors, and Oracle Corporation, preceded by general business. The training event is a two-day, multi-track, training extravaganza, provided at an unbelievably low cost.
- RMOUG has a major publication, this newsletter, with a circulation of over 1200.
- Most of the work, including the newsletter, presentations, and training, is done by volunteers from the board, membership, advertisers, vendors, and Oracle Corporation.
Advertising Rates

Publishing Schedule for RMOUG SQL>UPDATE

All rates are per issue and for space reservation and placement only.

**Black & White or 2 Color**
- Full page .................................. $500
- 1/2 page .................................. $300

**Full Color**
- (Full page only)
- Back Cover .............................. $1,250
- Inside Covers .......................... $1,000
- Inside ...................................... $750

*For special pricing on packages, please contact Allison Leech at newsletterdir@rmoug.org.*

All advertisements and graphics must be submitted at a **300dpi resolution** for proper printing requirements.

*The deadline for all articles and ads, respectively, for the next issue of RMOUG SQL>UPDATE is July 31, 2008.*

Publishing Schedule for RMOUG SQL>UPDATE

SQL>UPDATE_RMOUG is published quarterly. Please e-mail the director of newsletters prior to submitting/writing articles. All ads and articles are due on the following dates:

<table>
<thead>
<tr>
<th>Issue Date</th>
<th>Article Deadline</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spring</td>
<td>April 2009</td>
</tr>
<tr>
<td>Summer</td>
<td>June 2008</td>
</tr>
<tr>
<td>Fall</td>
<td>September 2008</td>
</tr>
<tr>
<td>Winter</td>
<td>November 2008</td>
</tr>
</tbody>
</table>

SQL>UPDATE_RMOUG is a publication of the Rocky Mountain Oracle Users Group P.O. Box 621942, Littleton, CO 80162. (303) 948-1786 Fax (801) 697-4366. All reproduction is expressly prohibited without specific written permission of the editor. The Rocky Mountain Oracle Users Group contributors and editors do not assume any liability for damages arising from the publication or non-publication of any advertisement, article, or other material in the newsletter.
Your membership benefit includes access to the RMOUG list servers. All the list servers except for the Announcements are voluntary. You can sign up for the list servers by going to:

http://lists.rmoug.org/mailman/listinfo

The techies list server is for technical questions only. We have a vast amount of technical expertise in our membership, so this is a great place to ask questions and learn new information.

The misc list server is for non-technical information like job postings, job searches, product endorsements. While only members can post, there is often forwarded information from recruiters, companies, etc. As a member, you can also forward job information that you think will be beneficial to the membership.

The ISProjVol list server is for the IS Project Volunteers who manage our website, so if you would like to help, join them!

The Announcements list server is for RMOUG-only announcements. Membership and posting to this list is managed by the RMOUG Administrator, so if you need a new e-mail address, you will need to e-mail it to admin@rmoug.org. This will also keep your membership information updated.

If you have any questions, feel free to e-mail Heidi Kuhn at heidikuhn@rmoug.org, or call (303) 948-1786.

A new benefit to RMOUG members is an Associate Membership to IOUG (Independent Oracle Users Group), if you are interested in more information about this benefit or wish to sign up; please visit their website www.ioug.org.
For those times when you need to contact.

Board of Directors

Bill Schwartzkopf       President@rmoug.org
Peggy King             VicePresident@rmoug.org
Linda Seley            Treasurer@rmoug.org
Jon Arnold             Secretary@rmoug.org
Ron Bich               MembershipDir@rmoug.org
John Peterson          EducationDir@rmoug.org
Jon Arnold             ProgramsDir@rmoug.org
Peggy King             TrainingDaysDir@rmoug.org
Allison Leech          NewsletterDir@rmoug.org
Barbara A. Lewis       ISDir@rmoug.org
Allison Leech          VendorsDir@rmoug.org
Chip Briggs            SIGSDir@rmoug.org
Vanessa Vedral         OracleLiaison1@rmoug.org
Troy Contestable       OracleLiaison2@rmoug.org
Ann Horton             OracleLiaison3@rmoug.org
Heidi Kuhn             HeidiKuhn@rmoug.org
Regis University's School of Computer & Information Sciences offers a 12-credit (4-course) Oracle database management systems certificate. In addition to enhancing Oracle skills, students can apply credits earned in this program toward our Master of Science in Database Technologies. This program is offered online and at seven Front Range campus locations through 8-week classes.

**Oracle Certificate**
- Database Architecture
- Database Concepts
- Choose two courses from the following:
  - Database Administration
  - Database Backup & Recovery
  - Database Performance Tuning
  - Advanced Database Technologies using OLAP

**Master of Science in Database Technologies**
The Regis University MS in Database Technologies emphasizes relational and object-relational database theory and implementation in concert with a study of the Oracle database management system. Graduates of this 36-credit (12-course) degree program are well equipped to work as database engineers and administrators in enterprise and Internet-based environments.

**Schedule**
- June 30 – August 24, 2008
- September 2 – October 26, 2008
- October 27 – December 21, 2008

Contact an enrollment counselor for tuition, locations, and times.

www.RegisUniversityOnline.org
1.800.944.7667 | masters@regis.edu

Regis University was founded in 1877 and embraces the Jesuit tradition of academic excellence and developing leaders committed to the service of others. Regis University College for Professional Studies was established to provide working adults with Master's and Bachelor's degrees and certificates while focusing on practical, real-world business solutions. The College for Professional Studies serves thousands of working adults each year who wish to advance their careers and enhance their lives.
**About Method R Corporation**

Method R Corporation is a Cary Millsap company that is committed to genuinely satisfying software performance. The company creates software tools, education courses, and consulting services dedicated to helping software application systems run faster and more efficiently. Method R focuses predominantly on Oracle-based systems, but it has the worldwide network in place today to assist with Microsoft SQL Server performance issues as well.

**Method R Education**

Method R is a performance optimization method based on the premise that you should work first to reduce the biggest response time component of a business’ most important user action. It’s not a method about looking at aggregated data or a blinking dashboard of red and green lights. It’s about focusing on your business and how performance gains can be realized by identifying the most important tasks and concentrating your efforts there. We’re so committed to this method that our company name is Method R! Our commitment to helping you to achieve “fully-informed performance optimization” is the driving focus in every class we teach. We strive to give you the knowledge and tools to make the job of performance optimization easier and faster than ever before.

Our curriculum is centered on how and why Oracle works the way it does; it is not just about features and facts. When you know how and why Oracle works the way it does, it becomes easier to prevent, or find and fix, performance problems quickly and efficiently.

**Oracle Performance Management using Response Time Profiling**

Our 3-day lab-based Profiling course transforms normal database administrators, application architects, and application developers into real performance analysts who can find and fix performance problems you may not have even realized that you had. People who take this course routinely fix system performance problems that have plagued their systems for months or even years, improving user productivity and reducing IT spend.

**Optimizing Oracle SQL**

Our 5-day lab-based SQL optimization course transforms application developers and database administrators into fearless SQL optimizers. Here, students learn to measure performance instead of guessing about it, using an efficient performance testing harness that the students get to take home. The course produces faster students writing faster SQL that will scale efficiently to handle your production workloads.

Visit www.method-r.com for full course descriptions and calendar of events

Method R Corporation  •  632 Silicon Drive Suite 102  •  Southlake, TX 76092 USA
+1.817.251.0938  •  info@method-r.com
Join us for an RMOUG Quarterly Educational Workshop!

RMOUG hosts Quarterly Education Workshops in May, August and November of each year with the fourth and largest educational event being Training Days which is held in February.

Our next meeting will be held Friday, May 16th, 2008:
Hotel Gold Crown
(formerly named Clarion Hotel)
7770 South Peoria Street, Englewood, CO 80112

Schedule:
7:30am Registration Opens
7:30am - 8:30am RMOUG Board Meeting (open to all members)
8:15am - 9:00am Continental Breakfast
9:00am - 9:15am General Session/Group Business/Board Elections
9:00am - 10:30am Oracle, LDAP and AD (Active Directory) Integrating the Three Together, by Debra Addeo
10:30am - 12:00pm Microsoft Reporting Services to report on Oracle E-Business data, by Carole Furnish
11:45am - 1:00pm Lunch (Sponsored by RMOUG)
1:00pm - 2:15pm Oracle Identity Management, by Niklas Iveslatt and Jenny McGurk
2:30pm - 3:45pm Getting Coherence: Introduction to Data Grids, by Randy Stafford

Presentations:
9:00am - 9:15am
Oracle, LDAP and AD (Active Directory) Integrating the Three Together, by Debra Addeo
This presentation will give an overview of how to integrate the 3 technologies for user id and password management along with a quick introduction into LDAP and Active Directory. The Oracle database has packages that allow access to Active Directory, Oracle Internet Directory and any other LDAP enabled directory for validating passwords for user ids. This allows developers to delegate the password management policies to another system without code maintenance. These stored packages are also used by other Oracle tools (i.e. Apex) to integrate applications with an LDAP server. A quick overview of how to integrate LDAP in Apex will also be discussed.

10:45am - 12:00pm
Microsoft Reporting Services to report on Oracle E-Business data, by Carole Furnish
We at Douglas County School District have had Oracle E-Business in place for over 12 years. You would think that after this time, at least most reporting requirements would have been met. Yet users still have difficulty getting the information they need and Apps Development continues to receive a flood of tickets requesting information or modifications to reports. This talk discusses a new approach to the reporting problem, as well as the use of a new tool, Microsoft Reporting Services.
1:00pm - 2:15pm
**Know your User Community with Oracle Identity Management, By Niklas Iveslatt and Jenny McGurk**
This presentation will be a case study of a full-scale Oracle Identity Management Suite Implementation (Oracle Identity Manager, Oracle Access Manager, Oracle Virtual Directory) with a user base of 6000 employee users. Identity Management, Access Control and Security are among the most important concerns of today’s Enterprises. A decade of rapid application development and system deployments have left companies with disjointed user and password stores. It takes an army of people to provision and control access to the many applications, users cannot remember their user logins and passwords, and it takes weeks before new employees have the right access to all necessary applications.

This presentation will be technical in nature and will give an overview of how the Oracle Identity Management Suite solution addresses the above problems. Oracle Virtual Directory, Oracle Access Manager and Oracle Enterprise Single Sign-On will be discussed.

However, the emphasis will be on how Oracle Identity Manager solves real life provisioning problems and how it integrates with ERP solutions, Active Directory and custom applications alike.

2:30pm - 3:45pm
**Getting Coherence: Introduction to Data Grids, by Randy Stafford**
Grid-based infrastructures are being developed, deployed and used to achieve unlimited application scalability and continuous availability across multiple datacenters. Understanding the additional capabilities of these infrastructures and how they can be improved with the use of Data Grid technology to solve increasingly difficult and complex problems ensures that your organization is getting the maximum utility from Grid Computing.

This presentation focuses on how Oracle Coherence Data Grid can easily help you achieve all of these goals and more!

This presentation will address:
- How Oracle Coherence capabilities function, such as coherent in-memory caching, dynamic data partitioning, and parallel query and process execution, and how they are being mapped onto grid infrastructures.
- How Data Grid capabilities function, how organizations are using them to solve complex computing problems and examples of how organizations are leveraging this on a global scale.
- How easy it is to deploy Oracle Coherence, which is generally operational within hours.
- How Oracle Coherence is fully configurable, providing total flexibility to change caching topology without code changes.

This presentation is recommended for developers, development managers, and architects who are interested in data grid technology, providing them with a hands-on, in-depth review of all of Oracle Coherence Data Grid capabilities, a Coherence demo, and clarifying when and how each of the different caching options and features should be used.
Join us for an RMOUG Special Interest Group Meeting!

Our next meeting will be held Friday, May 23rd, 2008:
FedEx, 350 Spectrum Loop Drive, Colorado Springs, CO 80921

**Schedule:**
- 1:00pm - 1:15pm General session/Group Business
- 1:15pm - 1:30pm Break
- 1:30pm - 2:30pm Presentation by Tim Gorman
- 2:30pm - 3:00pm Q&A

**Presentations:**
- **1:30pm - 2:30pm** Custom Reporting from RMAN
  
  Oracle’s “Recovery Manager” (RMAN) provides some limited reporting with the LIST and REPORT commands, but how does one obtain answers to some of the really *REALLY* important questions, such as:
  
  - Will I be able to recover to a specific point-in-time?
  - How long should it take me to restore and recover?
  - How much space are my backups taking?

  This presentation will describe the repositories used by RMAN, in the control file (i.e. NOCATALOG) as well as the “recovery catalog database”. Most importantly, this presentation will demonstrate how to report from them by examining reports written by Tim that answer the important questions shown above.

  The goal of the presentation is to help familiarize everyone with:
  
  - The data in the RMAN repository
  - Some techniques for reporting from it
  - How you can build your own reports
Printing (1-6 Color Offset & Digital)
- Labels, Booklets, Brochures, Large Format, Business Forms, Variable Imaging, Laser Forms & Checks

Advertising Speciality & Apparel Items
- Awards, Client Gifts, Incentive Programs, Event & Trade Show Giveaways, Hats & Uniforms, Polos & T-Shirts

E-Shop
- Online Company Store, International Distribution, E-Commerce support allowing access to real-time information relating to inventory levels, usage and more.

Storage/Distribution & Direct Mail Support
- Our storage and distribution services include inventory reports, tracking and product insurance. Distribution services range from daily pick-pack orders to large customized fulfillments.

Phone: 303.799.4646.
Sales@sunwestprinting.com
www.sunwestprinting.com
Another Awarding Winning Year!
TUSC Receives 2007 Oracle Titan Award

The Oracle PartnerNetwork is a global business network of more than 19,500 companies that deliver innovative software solutions based on Oracle software. However, only a handful (the best) are recognized each year with the Titan Award. TUSC is proud to announce that we’ve been recognized as a 2007 Titan Award winner – the third such award received during the short period of time Oracle has been recognizing partners in this manner.

Award winners must have implemented a solution that requires a strong understanding of customer requirements coupled with specific technical expertise and/or industry knowledge. Winners must also have demonstrated that the solution delivered measurable business results and clear customer satisfaction.

Our client’s comment regarding this year’s award-winning project:
“W e started out simply looking for a solution that would scale and accommodate our anticipated growth, but we achieved so much more. TUSC provided a more reliable solution for lower cost – one that is easy to manage, that will scale easily and that has improved our claims processing ability dramatically. TUSC really took the time to understand our business and our work processes, enabling them to design a first-rate solution.”

Hire TUSC as your implementation partner.
Experience the difference!

Oracle’s E-Business Suite
Implementations
Upgrades
Remote Support

BI / Data Warehousing
Strategy
Implementation

Custom Development
Fusion Middleware
Full Life Cycle Development

Managed Services
Remote Support
DBA and Database
Oracle E-business

Database Services
Health Checks
Troubleshooting
Full DBA Support
Remote Support

Training
Classroom (offsite & onsite)
Mentoring

Oracle Licensing

For more information call 800.755.TUSC — ask for a Solution Sales Account Executive. www.tusc.com