

# nocoug

JOURNAL

Official Publication of the Northern California Oracle Users Group

VOL. 17, No. 4 · NOVEMBER 2003

\$15

## 2004 Will Be a Great Year ... to Be a NoCOUG Member!

**A**nnual memberships for NoCOUG run from January 1 through December 31. This means that your membership will expire at the end of the year. Now is the time to renew your membership!

On page 26 you will find the Membership Application for 2004. Fill it out and mail it in with a check. Or, if you prefer, you can visit our website at <http://www.nocoug.org> and renew online.

Don't miss any of the benefits of NoCOUG in 2004, including our upcoming Winter Conference on February 19 at Oracle in Redwood Shores. \$70 a year gets you four great conferences, four issues of the NoCOUG Journal, our member directory, networking, volunteer opportunities, and much more. Renew your membership today! ▲



### Renew Today!

### Don't Miss Our Fall Conference!

The NoCOUG Board has planned a great conference that you won't want to miss. It's taking place on Thursday, November 13 at PG&E Headquarters in San Francisco. Check out the back page for the schedule!

Registration begins at 8:00 a.m. For more details, visit our website at [www.nocoug.org](http://www.nocoug.org). ▲

#### TABLE OF CONTENTS

2004 Will Be a Great Year.....	1
Editor's Note.....	2
NoCOUG Board.....	2
Publication and Submission Format.....	2
Advertising Rates.....	2
President's Message.....	3
Successful DBA Team Management.....	4
Meet NoCOUG's Director of Marketing.....	5
The Mystery of the "Null" Wait Event.....	6
PL/SQL Built for Speed.....	9
Converting MySQL to Oracle SQL.....	13
Book Reviews.....	15
How Cost-Based Optimization Works.....	16
Calling All Authors.....	19
Sponsorship Appreciation.....	20
Treasurer's Report.....	20
The Fourth "R"—Rman.....	21
NoCOUG Fall Conference Schedule.....	27

#### —ADVERTISERS—

Quest Software.....	8
Database Specialists.....	14
Embarcadero Technologies.....	14
Mission Critical 24/7.....	18
LECCOTECH.....	18

## Editor's Note

Another year as the Journal Editor for NoCOUG has gone quickly by. It's been a great year, and I'd like to thank everyone who contributed to the *Journal* in 2003. I'd also like to thank the members of the NoCOUG Board for their ongoing encouragement, support, and appreciation.

A couple of months ago, a very enthusiastic volunteer named **Laurie Robbins** got in touch with me and offered to help with this issue of the *NoCOUG Journal*. Laurie is an Oracle DBA relatively new to the Bay Area who wanted to get involved with our chapter of the IOUG. She has been a tremendous help in planning and organizing this issue. Having a partner in putting this 28-page journal together has made it a lot more fun! So, many thanks to Laurie. I hope we can continue working together and look forward to her ideas and contributions.



### Corrections

A few minor errors were made to Hamid Minoui's article, *Oracle Temporary Tables, Temporary Segments & Temporary Tablespace, Part II* (published August 2003) during the editing process. We apologize for any misunderstanding or inconvenience this may have caused.  
—Lisa Loper, *Journal Editor*

# Other User Groups

### Local

NorCalOAUG – Northern California Oracle Applications Users Group

- **Contact:** Michael Capelle  
(650) 562-1167
- **Email:** capelle@tru-course.com
- **Website:** [www.norcaloaug.org](http://www.norcaloaug.org)

### Sacramento

SacOUG – The Sacramento Oracle User Group

- **Contact:** Ravi Verma  
(916) 705-3261
- **Email:** ravi.verma@telcommand.com
- **Website:** [www.sacoug.org](http://www.sacoug.org)

### International

IOUG – International Oracle Users Group

- **Website:** [www.ioug.org](http://www.ioug.org)

### U.S. Domestic

OAUG – Oracle Applications Users Group

- **Website:** [www.oaug.org](http://www.oaug.org)

ODTUG – Oracle Development Tools User Group

- **Website:** [www.odtug.com](http://www.odtug.com)

Canvassing calls by employment recruiters to local chapter contacts is strongly discouraged.

## NOCOUG BOARD

### President

Roger Schrag, Database Specialists, Inc.  
rschrag@dbspecialists.com

### Vice President

Darrin Swan, LECCOTECH  
darrin@leccotech.com

### Treasurer/Secretary

Judy Lyman, Contra Costa County  
Public Works  
gooma@california.com

### Membership and Past President

Joel Rosingana, Independent Consultant  
joelros@pacbell.net

### Webmaster

Eric Hutchinson, Independent Consultant  
erichutchinson@comcast.net

### Journal Editor

Lisa Loper, Database Specialists, Inc.  
lloper@dbspecialists.com

### Vendor Relations

Mike DeVito, BankServ  
mike@devito.com

### IOUG-A Representative

Hamid Minoui, UPS  
hamid.minoui@ups-scs.com

### Members, At Large

Les Kopari, Corner Pine Consulting  
(650) 802-0563

Vilin Roufchaie, Cingular Wireless  
vilin.roufchaie@cingular.com

Eric Buskirk, Mission Critical 24/7  
ebuskirk@mc247.com

### Conference Agenda Coordinator

Colette Lamm, Independent Consultant  
colette\_lamm@yahoo.com

### Director of Marketing

Jen Hong, Cisco Systems  
hong\_jen@yahoo.com

## Publication and Submission Format

The *NoCOUG Journal* is published four times a year by the Northern California Oracle Users Group approximately two weeks prior to the quarterly regional meetings. Please send your questions, feedback, and submissions to: Lisa Loper, NoCOUG Journal Editor, at [journal@nocoug.org](mailto:journal@nocoug.org).

The submission deadline for the upcoming February issue is January 1, 2004. Article submissions should be made in electronic format via email if possible. Word documents are preferred.

*NoCOUG does not warrant the NoCOUG Journal to be error-free.*

Copyright ©2003 by the Northern California Oracle Users Group. Permission to reproduce articles from this publication, in whole or in part, is given to other computer user groups for nonprofit use, with appropriate credit to the original author and the *Northern California Oracle Users Group Journal*. All other reproduction is strictly prohibited without written permission of the editor. Two copies of each reprint should be sent to the editor.

## ADVERTISING RATES

### Contact: Nora Rosingana

325 Camaritas Way  
Danville, CA 94526  
Ph: (925) 820-1589

*The NoCOUG Journal is published quarterly.*

Size	Per Issue	Per Year
Quarter Page	\$100	\$320
Half Page	\$200	\$640
Full Page	\$400	\$1,280

*Personnel recruitment ads are not accepted.*



# Keeping Your Feet on the Ground and Reaching for the Stars



*Roger Schrag*

Attending OracleWorld in San Francisco in September, I was reminded of Casey Kasem's parting words each week at the end of American Top 40: "Until next week, keep your feet on the ground and keep reaching for the stars."

At his keynote presentation, Larry Ellison, chairman and CEO of Oracle Corporation, introduced the upcoming release of Oracle 10g and its support for the new paradigm of grid computing. The auditorium at Moscone Center was packed. The back of the room was lined with people willing to stand for an hour in order to hear what Oracle has in store for us.

As we made our way out of the auditorium at the end of the presentation, I asked a colleague what versions of Oracle were running at the Fortune 500 company where he works. He told me that about 90% of his production databases were Oracle 8i, with the other 10% Oracle 9i. I went back to my office and looked at the production databases my company supports. Not much different from my colleague: 80% Oracle 8i, 15% Oracle 9i, and 5% Oracle 8.0. I wonder what the breakdown was like for the other DBAs and developers in the audience at Larry's keynote.

It is easy to be a skeptic about Oracle's vision of the future. There is something ironic and almost comical about Larry Ellison presenting Oracle 10g to a room full of Oracle devotees who haven't even made it to Oracle 9i yet. But on the other hand, you can't stop progress. We'll get to 10g eventually. After all, it wasn't too many years ago that Larry was presenting Oracle 8i to a room full of people who were still running Oracle 7.3 and hadn't even begun to think about Oracle 8.0. Well, how many Oracle 7.3 and 8.0 databases do you still have in production today? As we eventually made our way to Oracle 8i, so we will find our way to 10g.

Oracle Corporation, with a lion's share of the database market and a huge development team, is in a commanding position to shape the future of information management. As Oracle professionals, we need to keep ourselves up to date on where information technology is heading and where the future is taking us. But while we are oohing and ahing about Oracle 10g and enterprise grid computing, we need to stay on board with today's reality—which for

many of us is Oracle 8i.

As Oracle professionals, we need to keep our balance. We need to keep our feet steady in the Oracle 8.0 / 8i / 9i reality of today while reaching for the Oracle 10g stars of tomorrow. We need to realize that upgrading Oracle versions is expensive when you consider the testing, validating, and bug fixing involved. We need to bear in mind that companies usually don't upgrade until there is a business justification for doing so.

While that business justification is forming, we can prepare ourselves for the future by learning what Oracle 10g and the grid paradigm have to offer. While being realistic about where we are today, we can be learning about the direction we are heading in and the new features and capabilities that will become available to us with Oracle 10g.

In the spirit of keeping your feet on the ground and reaching for the stars, I'd like to encourage you to come to the NoCOUG Fall Conference on November 13 in downtown San Francisco. We have a great lineup of technical presentations that will touch on everything from performance tuning and partitioning tables to bitmap indexes and ANSI standard SQL. We've packed a lot of sessions into the schedule so that you can maximize your learning and get the most value from the day. Thank you Pacific Gas and Electric for sponsoring this conference and providing the meeting facilities.

As we come to the end of 2003 and my year as President of NoCOUG, there are many others I would like to thank as well. I wish to thank the members of NoCOUG's board of directors for their hard work throughout the year. I would also like to thank all of the presenters and volunteers who contributed to NoCOUG in 2003—providing top-notch technical training, helping out at conferences, helping write the *NoCOUG Journal* and keeping the website up to date, and keeping things running smoothly behind the scenes.

Last, and certainly not least, I would like to thank all NoCOUG members for your participation in the organization and for making this such a successful user group. I look forward to seeing you at the November conference, and in 2004 as well. ▲

# Successful DBA Team Management

By Susan McClain, Systems Manager Alliance Data Systems

- ▶ A team that can work together, help and learn from each other, and remain loyal to its goals and direction is PRICELESS! Slice out time to work towards this goal...it's well worth the investment. People will notice, all the way up to the CIO/CEO level. You have enough responsibilities as a manager. Take care of your team and they will take care of you. Here are a few recommendations:
- ▶ Be open and honest with non-confidential issues. This will build trust.
- ▶ People can tell if you are sincere . . . be sincere.
- ▶ Take the time to listen. Be in the conversation and don't multi-task when someone is talking to you. What's small to you could be detrimental to someone else.
- ▶ Give credit where credit is due.
- ▶ Be loyal and stand up for your team.
- ▶ Don't criticize in public . . . and always compliment what was done correctly.
- ▶ Tell your success stories to your managers. Let them know when you save money or really impress a client.
- ▶ Be human. Smile and let people know something personal

about you, even if it's your favorite restaurant or type of food.

- ▶ Schedule time to do something fun with your team. Hold off-site events that are just for fun. Try to do something small quarterly. It gives everyone an outing to look forward to. Schedule an afternoon movie, or some type of after work event. Put suggestions in a hat and rotate who selects the outing. Do your best to make sure everyone can attend and plan something on a regular basis.
- ▶ Managing any technical team requires knowing something about the work they perform. Learn as much as you can about what a DBA does so you can offer guidance and understanding. Employ the "right" kind of DBA for your business. Make sure that person understands the responsibilities and can handle the demands of the job. Know the dynamics of your team. Learn what makes them tick individually and as a group. Develop professional and technical skills based on your discoveries.
- ▶ Try to simplify where you can. Define goals and have schedules for routine activities. Build relationships with employees, associates, peers, clients, and vendors. Build a network of

people you can rely on to get the job done. Take time out to have fun, it makes the challenges go a little easier.

- ▶ Most of all walk the talk. Lead by example and expect no less from yourself. Be credible, sincere and strong enough to make the best decisions for the business, even if it's not the popular solution. Cultivating a strong and loyal team takes time, some successes and some mistakes. Be patient and learn from each experience. Ask for feedback from your employees and your manager. ▲

---

*Susan McClain is an IT systems manager for a team of database administrators with Alliance Data Systems and has worked with Oracle versions 7 and up, including PL/SQL for several years. Susan has presented at Oracle Open World, other Oracle User Groups and IOUG-A. She was recently a member of the IOUG conference committee. Susan has worked in the IS industry since 1986. She served as President to the Ohio Oracle Users group for the past two years and has prior experience with the other board positions. Susan can be reached at Alliance Data Systems at (614) 729-4338 or by email at [Smcclain@alldata.net](mailto:Smcclain@alldata.net).*

*This article was excerpted with permission from the IOUG-Live! Conference 2003.*

## Mark Your Calendars!

The NoCOUG Winter Conference will take place on Thursday, February 19, 2004 at the Oracle Conference Center in Redwood Shores. You won't want to miss it!



# Meet NoCOUG's Director of Marketing

Jen Hong joined the NoCOUG board of directors this summer as a Director At Large. After getting the lay of the land, Jen recently jumped into the newly created role of Director of Marketing. He told the board that he is excited to take on the challenges of this new role and develop a marketing plan that will help make our organization more successful. We are looking forward to Jen's continued contributions. His commitment, knowledge and enthusiasm have been a great addition to our NoCOUG team. Let's find out a little bit more about our new Director of Marketing.

**First off, tell us about your Oracle background. How did you originally get involved in Oracle technology?**

With a BS degree in computer science, I always have a fascination on how database management systems works. I started out with Oracle ten years ago while implementing a global ERP system.

**What is your role in your present job?**

In my present job, I am a database administrator managing Oracle databases and working with various project teams.

**In your day-to-day work, what are some of the challenges you most enjoy?**

With the advancements of modern technology there are more tools to enable people to work virtually and in closer collaboration. I enjoy working with people working together to accomplish common goals and objectives and meeting the high demands of business.

**If you had to choose a profession outside of technology, what do you think it would be and why?**

If given an opportunity, I'd choose to be in the medical field. Not working for a living, but to help people to live healthier, happier, and better. I think it is incredibly rewarding to help people that are in pain and suffering both physically and mentally. To help them get healthier and allow them to enjoy their lives better to achieve their potential each day.

**We've heard that you and your wife recently started a family. What has been the biggest surprise about fatherhood?**

I think the biggest surprise for me as a father is the enjoyment of spending time with my son, David. All responsibilities and sleep deprived nights aside, I found much happiness doing nothing, slowing down and just spending time playing with him. Kids are so much fun.

**What are your favorite pastimes/hobbies?**

I have many pastimes. One of the favorite is travel and another is reading. I enjoy traveling to new places. Seeing attractions and destinations for myself and making discoveries of architecture, art, culture, foods, and people. Seeing the beauties and wonders of the place we call home—"Earth." I also enjoy reading self-help books. They help me to understand and enlarge my perspective in life and from other's people's point of view.

**With work and family, how do you still find the time to volunteer with NoCOUG?**

Honestly, I do not have much time for volunteering with NoCOUG with work and a young family. However, it is a commitment that I like to keep. I found myself to be more efficient in the way I do things.

**Why is it important to you?**

I am a believer in cause and NoCOUG belongs to this category. This user group provides an invaluable forum and community for Oracle professionals in the greater San Francisco and Bay Area to learn and network together with peers. I believe that being involved



Jen Hong

is a good way to be part of something good and worthwhile.

**What has been most enjoyable about being a board member?**

I enjoy planning for the quarterly conferences and helping to bring the events together along with a group of talented people.

**How long have you lived in the Bay Area?**

I grew up and was educated in Southern California and have been working up and down the state. I came up to Silicon Valley five years ago.

**Where is the most interesting place you have traveled?**

There are many, but I think one of the most interesting places I have been is hiking on the trails of the Cliffs of Moher in County Clare, Ireland and singing Irish folk songs in an Irish castle with a hundred other strangers.

**And, of course, we're always looking for good resources. What is one of your favorite Oracle-related websites?**

There are many favorite Oracle websites that I like and offer great resources. I would just mention a few:

- Oracle Technology Network—<http://otn.oracle.com/index.html>
- Google—[www.google.com](http://www.google.com)
- DBAZine—[www.dbazine.com/oracle.shtml](http://www.dbazine.com/oracle.shtml)
- Metalink – <http://metalink.oracle.com> ▲

# The Mystery of the “Null” Wait Event

By Chris Lawson

**G**ood Oracle performance tuning specialists are always looking for better ways to diagnose database performance problems. As the technology changes, and as we increase our understanding of database internals, we will want to experiment with new ideas and new approaches to performance tuning. Sometimes this also means admitting that some of our former methods may no longer be appropriate.

For instance, it was popular for many years to focus on a metric usually called the cache “hit ratio.” This statistic indicates how frequently data is being accessed in the database cache, as opposed to having to read it from disk. The idea behind this method is to try to get a hit ratio as close as possible to 1.0. The closer you get to 1.0, the better your database (supposedly) ran.

Of course, in the past several years, many DBAs have discovered that this “hit ratio” method can be very misleading. Among many problems with this method is the unpleasant fact that poorly tuned databases may also have a very “good” hit ratio! As a consequence, few performance experts consider this metric a reliable indicator of proper database operation.<sup>1</sup>

## A Better Way—Oracle’s “Wait Event” Facility

Rather than focusing on one simplistic (and misleading) metric, many DBAs are learning to use a diagnostic methodology called Oracle “Wait Events.” Admittedly, this method is more complicated than the simplistic “hit ratio,” but it has quite a few advances. Best of all, the wait event method focuses on finding the root cause of performance problems.

<sup>1</sup> One noted performance expert actually uses a high hit ratio as a strong indicator of poor database performance!

The essence of the Wait Event approach is simple: we focus on trying to find out why a particular session is waiting. Presumably, if we know why a session is waiting, we can do something about it. For instance, could it be that the database is waiting to read from disk? Is it waiting for me to respond to a Sql\*Plus prompt? Am I being blocked by some other user?

Fortunately for the performance tuner, the Oracle database designers have provided a way to get answers to these questions. Using some special views, it is possible to find out what a particular session is waiting for, and for how long.

The Wait Event approach has proven to be an excellent diagnostic tool; in fact, many DBAs begin their performance analysis by using the Wait Event views.

## Oracle 9i’s “Mystery” Events

Unfortunately, there are some problems in Oracle 9i that make the Wait Event facility a little harder for DBAs to use. To see why, let’s review how the Wait facility is supposed to work.

Ideally, any database process that is holding things up is supposed to be listed by name in a certain V\$Wait view, so that the DBA can easily identify problems and begin troubleshooting. Thus, if the database is waiting on disk I/O, there should be an entry listing this fact, along with statistics such as “Wait Time.” Clearly, this is extremely valuable information for the performance specialist.

In Oracle 9i, however, it is not unusual for an event to be titled “Null Event.” No, this doesn’t mean that Oracle is waiting on “nothing.” Instead, this simply means that the Oracle programmers goofed—they forgot to put a title on some events. So, if your session is waiting on disk i/o, instead of seeing the Wait Event called “Scattered Reads,” you might see one called “Null Event.”

Obviously, this makes it very difficult to diagnose performance problems, since the DBA has to guess on what the event really is. A quick search of the Internet will show several database forums with questions from frustrated DBAs trying to solve a problem that is only known as a “Null Event.”



**Wait Events  
reveal what is  
holding things up.**

**You imbecile!  
What do you  
mean, the data-  
base is waiting  
on “nothing”?**



Of course the DBA's boss will not appreciate the point that the database is not really waiting for anything. The performance tuner needs to get some answers—but where?

### Some Clues for the Oracle Detective

Fortunately for the DBA, we have some extra information that can help unravel the puzzle. Remember that each Wait Event has three parameters (called P1, P2, and P3). Although the “mystery” event is not named properly, the P1–P3 parameters appear to still be correct. The exact value of these parameters can give you clues on what the event probably is.



**Hmmm ... Let's  
look at the  
P1–P3  
parameters**

For instance, the P1 and P2 parameters for a multi-block read (called “Scattered Read” by Oracle) will contain the File# and Block ID that correspond to the database object being read. Therefore, if the P1 and P2 parameters for a given Null Event correlate with an actual file# and block ID of a table that you are reading, it is likely that the Null Event is really a disk-read event. (Use the DBA\_EXTENTS table to look up File# and Block\_Id.)

You can gain further confidence in your preliminary identification of the Null Event as a disk-read if different Null Events continue to point to the same table (different blocks, however). On the other hand, if the parameters do not appear to match the database object being read, then your preliminary identification of the Wait Event as a disk read was probably in error.

### Postscript

Oracle Corporation has been soundly lambasted for the shortcoming of the 9i Wait Event facility. To their credit, they have apparently reduced the problem in Oracle 9.2.0.3, although the author has still seen the occasional Null Event even with 9.2.0.3. Note that the Oracle patch to 9.2.0.3 has nothing to do with actually solving the performance problem—it simply corrects the diagnostic routines so that the Wait Events are correctly named.

Once you are able to identify the Wait Event—either by upgrading to 9.2.0.3, or by clever deductions using the P1–P3 parameters, you can now begin the real work of solving the performance bottleneck. Of course, with the Wait Event correctly identified, you won't have any more excuses—you will have to go about your real job of eliminating those pesky waits.

Of course, if you don't fix the performance problem, your boss may still yell at you, but at least he won't be yelling at you for nothing. With the help of Oracle's Wait Events, you will have the satisfaction of knowing exactly why he is yelling at you. ▲

### About the Author

*Chris Lawson is a DBA consultant. His recent book, The Art & Science of Oracle Performance Tuning, has received numerous five-star reviews, and is available on Amazon.com.*



## TECH TIPS

**Here's a quick tip from Tom Kyte's  
book, *Expert One-On-One Oracle***

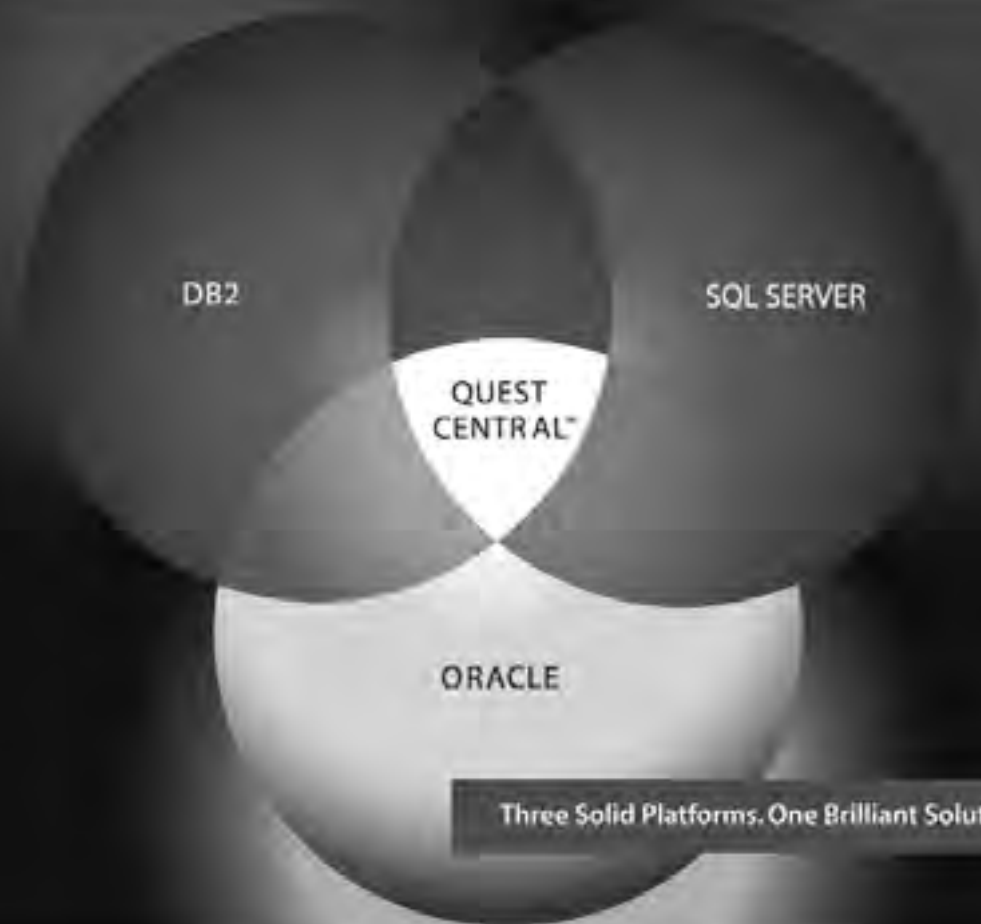
—Submitted by Laurie Robbins

**Y**ou can construct a SQL Query to order the contents of a table, and return the results in sections. Using the EMP table from the SCOTT schema, the example below shows a query that will return rows 5 through 10. You can just imagine a next button on the user interface, allowing the user to continue to view the contents of the table as needed:

```
SELECT ename, hiredate, sal
FROM (select ename, hiredate, sal,
row_number() over (order by ename) rn FROM emp)
WHERE rn between 5 and 10
ORDER BY rn;
```

For more examples and explanations, check out the Analytic Functions Chapter in Tom Kyte's book. ▲

(APPLICATION CONFIDENCE)



Three Solid Platforms. One Brilliant Solution.

**You work in a heterogeneous environment. Your database management software should, too.** With Quest Central you can manage more critical databases with astonishing ease — regardless of platform. It delivers comprehensive production management from administration to diagnostics, analysis, space management, SQL tuning and more — all with built-in expertise and a single intuitive, console-based interface. Try Quest Central. It's not just for Oracle anymore. Download "Surviving in a Multi-database Environment," at [www.quest.com/mocoug](http://www.quest.com/mocoug) and find out how Quest Central changes database management forever.



© 2003 Quest Software Inc., Irvine, CA 92618-1699. All rights reserved.

QUEST CENTRAL™: MANAGE MORE DATABASES TODAY



# PL/SQL Built For Speed

by James F. Koopmann

**E**veryone is looking for an edge. With PL/SQL that edge is compiling the PL/SQL code into native code. This article will show you how to set up and compile your PL/SQL procedures and packages.



The default behavior for creation of PL/SQL code execution is purely interpretive. If you want to skip the interpreted aspect of PL/SQL and jump through to native C code, then you must use a C compiler on the system. When you do this Oracle generates native C code, then compiles it with the C compiler on your system, turns it into shared libraries, and then it is linked into the Oracle process. This means that when these procedures are called they are immediately available for execution. All of this is dynamic and you don't need to restart the database to take advantage of this feature. Just remember, the only speed increase you will achieve is PL/SQL code execution, not SQL execution. So if your PL/SQL code is laced with SQL statements and not logical statements, the switch to C code will not buy you much speed.

## Set Up to Compile

1. Within the \$ORACLE\_HOME/plsql directory, there is a makefile called spnc\_makefile.mk. You will need to change this file for the appropriate paths for the C compiler. A quick look at this file and a few verifications proved un-eventful as Oracle had all the paths set for my Linux installation. Table 1 will show the paths I verified and what command I used to verify them.

**TABLE 1: Verification of path entries in \$ORACLE\_HOME/plsql/spnc\_makefile.mk file**

Entry in file	Unix command to verify	Output to Unix command
PLSQLHOME=\$(ORACLE_HOME)/plsql/	ls -d \$ORACLE_HOME/plsql	/u01/app/oracle/product/9.2/plsql
PLSQLINCLUDE=\$(PLSQLHOME)include/	ls -d \$ORACLE_HOME/plsql/include	/u01/app/oracle/product/9.2/ plsql/include
PLSQLPUBLIC=\$(PLSQLHOME)public/	ls -d \$ORACLE_HOME/plsql/public	/u01/app/oracle/product/9.2/ plsql/public
RM=/bin/rm -f	which rm	/bin/rm
CC=/usr/bin/gcc	which gcc	/usr/bin/gcc
LD=/usr/bin/ld	which ld	/usr/bin/ld

2. The initialization parameter PLSQL\_COMPILER\_FLAGS must be set to NATIVE. The default value for this parameter is INTERPRETED. You can verify your current setting by issuing the "SHOW PARAMETER plsql\_compiler\_flags" command. You can change the setting by issuing an ALTER SESSION or ALTER SYSTEM command. Listing A shows my session to verify my current setting and set this parameter to NATIVE.

**LISTING A: Session to set initialization parameter PLSQL\_COMPILER\_FLAGS**

```
SQL> connect sys/<password> as sysdba
Connected.
SQL> show parameter plsql_compiler_flags

NAME                                 TYPE                                 VALUE
-----                                 -----                                 -----
plsql_compiler_flags                 string                               INTERPRETED

SQL> ALTER SYSTEM SET plsql_compiler_flags = NATIVE SCOPE = both;
System altered.

SQL> show parameter plsql_compiler_flags
NAME                                 TYPE                                 VALUE
-----                                 -----                                 -----
plsql_compiler_flags                 string                               NATIVE
```

3. Listing B gives more parameters that you must set to control the compiling of PL/SQL. Read through them and take appropriate action according to your environment. Each of these parameters may be set through the ALTER SYSTEM or ALTER SESSION commands. Listing C gives my session for setting these parameters.

**LISTING B: Parameters for controlling PL/SQL compiling to Native C**

Parameter	Description	Value
PLSQL_NATIVE_LIBRARY_DIR	This is the name of the directory where compiled objects will be stored	/u01/app/oracle/product/9.2/plsql/lib\
PLSQL_NATIVE_C_COMPILER	Gives the full path of where the C compiler is at. This is an optional parameter and Oracle will pick up the value for the "CC" entry in the spnc_makefile.mk makefile. By setting this parameter you will override the entry in the spnc_makefile.mk makefile.	Use default in spnc_makefile.mk makefile
PLSQL_NATIVE_LIBRARY_SUBDIR_COUNT	Use this parameter to help reduce the performance impact of a large number of compiled objects in the PLSQL_NATIVE_LIBRARY_DIR. If very high number (10000+) compiled objects are done, it is best to segregate these into subdirectories so that the performance impact of file access operations is kept to a minimal. To take advantage of this, set the PLSQL_NATIVE_LIBRARY_SUBDIR_COUNT to a value where each of the total number of compiled objects divided by the number of subdirectories will allow for less than 10000 entries in each of the subdirectories. For example, if you wanted 500 subdirectories, set PLSQL_NATIVE_LIBRARY_SUBDIR_COUNT to 500 and create 500 subdirectories named d0 thru d499. The subdirectories must be prefixed with the letter "d".	I set this to 500 and created subdirectories /u01/app/oracle/product/9.2/plsql/lib/d0 thru /u01/app/oracle/product/9.2/plsql/lib/d499
PLSQL_NATIVE_LINKER	This is the name of the linker to use. This is an optional parameter and Oracle will pick up the value for the "LD" entry in the spnc_makefile.mk makefile. By setting this parameter you will override the entry in the spnc_makefile.mk makefile.	Use default in spnc_makefile.mk makefile
PLSQL_NATIVE_MAKE_FILE_NAME	This is the full path of the spnc_makefile.mk makefile. /u01/app/oracle/product/9.2/plsql/spnc_makefile.mk	/u01/app/oracle/product/9.2/plsql/spnc_makefile.mk
PLSQL_NATIVE_MAKE_UTILITY	This is the full path of a make utility.	/usr/bin/gmake

## LISTINC C: Script for setting additional parameters for Native C compiling

```
SQL> connect sys/<password> as sysdba
Connected.
SQL> show parameter native

NAME                                                    TYPE                                VALUE
-----
-----
-----
plsql_native_c_compiler                                string
plsql_native_library_dir                               string
plsql_native_library_subdir_count                     integer    0
plsql_native_linker                                   string
plsql_native_make_file_name                           string
plsql_native_make_utility                             string

SQL> alter system set PLSQL_NATIVE_LIBRARY_DIR =
                '/u01/app/oracle/product/9.2/plsql/lib'
scope=both;

System altered.

SQL> alter system set PLSQL_NATIVE_LIBRARY_SUBDIR_COUNT = 500 scope=both;

System altered.

SQL> alter system set PLSQL_NATIVE_MAKE_FILE_NAME =
                '/u01/app/oracle/product/9.2/plsql/spnc_makefile.mk'
scope=both;

System altered.

SQL> alter system set PLSQL_NATIVE_MAKE_UTILITY = '/usr/bin/gmake'
scope=both;

System altered.

SQL> show parameter native

NAME                                                    TYPE                                VALUE
-----
-----
-----
plsql_native_c_compiler                                string
plsql_native_library_dir                               string /u01/app/oracle/product/9.2/plsql/lib
plsql_native_library_subdir_count                     integer 0
plsql_native_linker                                   string
plsql_native_make_file_name                           string /u01/app/oracle/product/9.2/plsql/spnc_makefile.mk
plsql_native_make_utility                             string /usr/bin/gmake
```

### Compiling

Now that we have set the initialization parameter for NATIVE compiles, whenever we CREATE a new procedure or package, all procedures and packages will be compiled in C. If packages or procedures already exist and you want to compile them to NATIVE C, all you need to do is issue the ALTER [PROCEDURE | PACKAGE] {procedure\_name|package\_name} COMPILE command. Listing D gives a quick example of what happens for a newly created procedure. As you can see, the first time that it is compiled it gives a message that the object does not exist in the subdirectory. This is fine since it is only trying to create a backup of the source that this created procedure is going to replace. Listing E shows that we can take a look at the newly created DLL in the directory. After subsequent calls to re-create the procedure, you can see in Listing F that Oracle

*As you can see, the first time that it is compiled it gives a message that the object does not exist in the subdirectory.*

will make backup copies of the DLLs. Whenever we re-compile this procedure it will always compile to NATIVE C. It does not matter what the current settings of the init parameters have been changed to. The only way to re-compile into INTERPRETIVE is to drop the procedure, reset the init parameters for INTERPRETIVE, and CREATE the object.

## LISTING D: First creation of a simple procedure

```
SQL> CREATE OR REPLACE PROCEDURE testc
 2 AS
 3 BEGIN
 4     DBMS_OUTPUT.PUT_LINE('Hello world');
 5 END;
 6 /
mv: cannot stat `/u01/app/oracle/product/9.2/plsql/lib/d10/TESTC__SYS__0.so': No such file or directory
Procedure created.
SQL> set serveroutput on
SQL> exec testc
Hello world

PL/SQL procedure successfully completed
```

## LISTING E: Directory listing of the newly created procedure

```
sh-2.05$ ls -l
-rw-r--r-- 1 oracle oinstall 10240 Nov 11 11:14 /u01/app/oracle/product/9.2/plsql/lib/d10/TESTC__SYS__0.so
-rw-r--r-- 1 oracle oinstall 10240 Nov 11 11:14 /u01/app/oracle/product/9.2/plsql/lib/d10/TESTC__SYS__0.so
```

## LISTING F: Directory listing of backup copies made by Oracle

```
sh-2.05$ ls -l /u01/app/oracle/product/9.2/plsql/lib/d10/TESTC__SY
-rw-r--r-- 1 oracle oinstall 10240 Nov 11 11:14 /u01/app/oracle/product/9.2/plsql/lib/d10/TESTC__SYS__0.so
-rw-r--r-- 1 oracle oinstall 10240 Nov 11 11:14 /u01/app/oracle/product/9.2/plsql/lib/d10/TESTC__SYS__0.so.11141
-rw-r--r-- 1 oracle oinstall 10240 Nov 11 11:14 /u01/app/oracle/product/9.2/plsql/lib/d10/TESTC__SYS__0.so.11149
-rw-r--r-- 1 oracle oinstall 10240 Nov 11 11:14 /u01/app/oracle/product/9.2/plsql/lib/d10/TESTC__SYS__0.so.11157
```

## Confirmation of NATIVE Compile

We should now confirm that what we have compiled is truly in NATIVE C. To do this we just need to issue the following SQL to look at the object we have compiled and validate that it has a NATIVE value returned. Listing G shows the simple SQL that will verify a procedure named TESTC was effectively compiled to NATIVE C.

## LISTING G: SQL to verify NATIVE C compile

```
SELECT object_name,param_value
FROM user_stored_settings
WHERE object_name = 'TESTC'
AND param_name = 'plsql_compiler_flags'
```

OBJECT_NAME	PARAM_VALUE
TESTC	NATIVE, NON_DEBUG

## Start Your Engines

If your procedures or packages are laden with logic that seems to run on forever, you should take a look at compiling these to NATIVE C. It will speed up your execution by making code jumps more efficient, function calls quicker, and use of memory more efficient. NATIVE C will give you that boost of energy you have always wanted to give your code. Just remember that DML operations are not effected since table access is a constant and compiling to NATIVE C will not do anything for it. If you want to take the next step, you can also compile the Oracle supplied packages and procedures to give the Oracle engine a boost. If you want to go that route, Oracle supplies a script called utlirp.sql that is located in \$ORACLE\_HOME/rdbms/admin that will invalidate and re-compile all Oracle PL/SQL modules. Go ahead, start your engines. ▲

*James F. Koopmann is the Database Practice Manager at Blue Oasis Technologies, Inc. He can be reached at james\_koopmann@yahoo.com.*

# Converting MySQL to Oracle SQL

By Les Kopari, Corner Pine Computing

**M**any vendors have their own flavor of SQL. Probably one of the first things that comes to mind when considering the task of converting various vendors' SQL are the excellent conversion tools available directly from Oracle for free download: the Migration Workbench.<sup>1</sup> An example is when you're converting MySQL to Oracle SQL.

Possibly the second thing that comes to mind is to wonder how much would be involved in just converting a small bit to use SQL\*Plus directly. Whether you are a developer wanting to just code up a small bit from something already available, or you're a DBA trying to understand what's going on under the hood, as it were, you may suddenly find yourself struck with that wonderful impulse to get your hands on a keyboard for a quick coding session.

Well, actually, there aren't that many major differences. To illustrate, here are a dozen or so typical steps needed to convert SQL, including examples with a few vi commands to spice things up.

These steps were taken to convert SQL in an open-source academic portal, Moodle, for institutions with Oracle already installed.<sup>2</sup>

They should at least provide you with a good start.

## 1. Remove all ` around table names, replace # with rem:

```
1,$s/`//g
1,$s/# /rem /g
```

before:

```
create table `config` ( # Table structure for
table `config`
```

after:

```
rem Table structure for table config
create table config (
```

## 2. Add drop table before create table

```
drop TABLE prefix_config;
CREATE TABLE prefix_config (
```

## 3. Change int to number:

```
1,$s/ int(/ number(/g
1,$s/ tinyint(/ number(/
```

## 4. Change varchar to varchar2:

```
1,$s/ varchar(/ varchar2(/
```

5. Similarly for other datatypes, but check default datatype mappings.<sup>3</sup>

6. Replace auto\_increment with primary key and create sequence & trigger:

before:

```
CREATE TABLE `prefix_config` (
`id` int(10) unsigned NOT NULL auto_increment,
```

after:

```
CREATE TABLE prefix_config (
id number(10) primary key,
etc...
```

```
drop sequence p_config_seq;
create sequence p_config_seq;
```

```
create or replace trigger p_config_trig
before insert on prefix_config
referencing new as new_row
for each row
begin
```

```
select p_config_seq.nextval into :new_row.id
from dual;
end;
```

```
·
/
```

## 7. Move not null from before default to after:

```
1,$s/ NOT NULL default '0'/ default '0' not null/g
1,$s/ NOT NULL default '1'/ default '1' not null/g
1,$s/ NOT NULL default '2'/ default '2' not null/g
```

check for others and make similar changes.

8. Delete primary key lines, converting multicolumn keys into alter table statements:

```
ALTER TABLE prefix_backup_files
ADD CONSTRAINT pbf_pk
PRIMARY KEY (backup_code,file_type,path);
```

### 9. Delete type=MyISAM:

before:

```
) TYPE=MyISAM;
```

after:

```
);
```

### 10. Turn comment clause into comment statement:

before:

```
) TYPE=MyISAM COMMENT='Moodle config variables';
```

after:

```
);  
COMMENT on table prefix_config is 'Moodle  
config variables';
```

### 11. Change text to varchar2 (but CLOB is recommended):

before:

```
`summary` text NOT NULL,
```

after:

```
summary varchar2(1024) not null,
```

### 12. Delete unique key, replace with column constraint

before:

```
`name` varchar(255) NOT NULL default '',
```

after:

```
name varchar2(255) default '' not null  
constraint unq_name unique,
```

### 13. Change KEY to INDEX

before:

```
KEY `courseuserid` (course,userid)
```

after:

```
create index courseuserid on  
prefix_course_display(course,userid);
```

### 14. Remove all unsigned

```
1,$s/ unsigned / /g
```

before:

```
`sortorder` int(10) unsigned NOT NULL default  
'0',
```

after:

```
sortorder number(10) default '0' not null,
```

### 15. Change enum to check constraint:

before:

```
course int(10) unsigned NOT NULL default '0',  
type
```



## There's No Substitute for Experience

**O**ur team represents some of the most knowledgeable and experienced in the industry. We are authors and speakers with long careers as Oracle experts, averaging 12 years. Our specialty is providing remote DBA services and onsite Oracle database consulting.

We offer a free consultation to discuss:

- Increasing uptime and reliability
- Minimizing downtime and data loss
- Optimizing performance
- Reducing cost of database operations

### Call Us Today!

(415) 344-0500 • (888) 648-0500  
www.dbspecialists.com

ORACLE | CERTIFIED SOLUTION PARTNER

  
**Database Specialists**



**What kind of a person thinks building and managing enterprise databases is fun?**

Hey, it could be you. You might be surprised at how smooth it can be with Oracle. This actually works really, really well. (Whisper!) For someone who's got you through enterprise systems, storage, performance and security management challenges with innovative, game-changing ideas. And Embarcadero's growing list of solutions into things that solve existing (and future) issues and provide for a whole range of development, management and integration scenarios for all kinds of IT environments. Completely transparent to avoid any. Please to support. Good to help. Ever was. (And in fact, 94 of the Fortune 100 are now using Embarcadero software. Talk to us at 415-884-0125. Or why not download a free, full-featured evaluation copy of one of our products and save a buck for yourself. [www.embarcadero.com](http://www.embarcadero.com))

**EMBARCADERO  
TECHNOLOGIES**

```
enum('single','news','general','social','eachuser',
,'teacher')
NOT NULL default 'general',
```

after:

```
course number(10) default '0' NOT NULL,
type varchar2(64) default 'general' not null,
constraint type_check CHECK (type IN (
'single','news','general','social','eachuser','
teacher')),
```

**16. Adapt names from 64 MySQL max to 30 Oracle max; one way is by simple truncation, but the name may lose uniqueness, or cause interface problems.**

before:

```
# Table structure for table
glossary_entries_category
CREATE TABLE prefix_glossary_entries_category (
```

after:

```
rem Table structure for table
glossary_entries_category
drop TABLE prefix_glossary_entries_catego;
CREATE TABLE prefix_glossary_entries_catego (
```

Keep in mind that choosing datatypes sometimes involves a judgment call; for instance the TEXT datatype should be converted to CLOB, but in the example was just set to 1024 VARCHAR2. Be sure to check the default datatype conversions. (3)

By the way, deleting lines may leave trailing commas, leading to a distracting error:

```
SQL> create table test(
2  n number(10),
3  );
)
*
ERROR at line 3:
ORA-00904: : invalid identifier
```

But they are easy to recognize and fix.

### Conclusion

Making the move from MySQL is not that difficult with the excellent tools provided by our vendor, but there may be instances when you wonder what's going on under the hood.

In these cases, a simple conversion using these steps may result in something useful. Moodle is being converted this way to work on current releases of Oracle.

### References

1. Oracle Migration Workbench Reference Guide for MySQL 3.22, 3.23 Migrations, Part No. A97249: [http://download-west.oracle.com/docs/cd/B10501\\_01/win.920/a97249.pdf](http://download-west.oracle.com/docs/cd/B10501_01/win.920/a97249.pdf)

2. <http://moodle.org>

3. A97249, page 25: Default Data Type Mappings

### About the Author

*Les Kopari is an independent consultant in Redwood Shores providing programming and database administration. His conversions have ranged from mag-card readers to mainframes, OCL to MPE, MAPICS to MM, and MySQL to Oracle. He has a BS degree from Old Dominion University, and his certifications include Oracle's OCP, CompTIA's A+ and I-Net+. He enjoys playing Bach when he can find a piano nearby. His con-*

## Interested in Writing a Book Review?

Review books are available from O'Reilly books.

Copies of O'Reilly books are available for NoCOUG members to review. O'Reilly loves to have feedback on all of their titles. Book and product reviews make great website and newsletter content. If you need some suggestions, here are a few from the handy-dandy guide from O'Reilly:

- Include the book's title, author(s), publisher, and price.
- Provide the catalog page URL for your book from our book list <http://www.oreilly.com/catalog>
- Include information from the catalog page for each book, such as a link to the sample chapter or author articles.
- Create a rating system with your members. Be creative. For example, the Alaskan Apple Users Group uses Moose to rate their reviews. Five Moose means "Terrific."
- Be honest.
- The length is up to you. Reviews can be short and sweet or long and detailed. Do what works best for you.
- If you like something in a book, say what it is and why you like it. If you don't like something in the book, say what it is and say why you don't like it. Be as specific as you can—this information is important to other readers.
- If something was omitted in the book say what it is and why it is important.
- End the review with an overall summary.
- Have fun!

If you are interested in writing a review, send an email to [journal@nocoug.org](mailto:journal@nocoug.org). Please include the book title and order number on your request. ▲

# How Cost-Based Optimization Works

By Jonathan Lewis, JL Computer Consultancy (UK)

## An Experiment

It is always a little risky to dictate to the reader how their database will behave with a given set of data—there are so many parameter settings and feature differences that may make two “identical” situations behave completely differently.

However, if you create the following two tables in a database based on a block size of 8K (this particular example is set up to work well with the typical default install of an Oracle 8.1 database), index them on the column N1 and then *analyze table . . . compute statistics*, you will (probably) have built the perfect platform for starting your exploration of the *Cost-Based Optimizer*.

```
create table t1 as
select
    trunc((rownum-1)/15)    n1,
    rpad('x',215)          v1
from all_objects
where rownum <= 3000;

create table t2 as
select
    mod(rownum,200)        n1,
    rpad('x',215)          v1
from all_objects
where rownum <= 3000;
```

Once you have built the data and indexes you might like to convince yourself that the two sets of data are identical by running the following pair of queries:

```
select n1, count(*) from t1
group by n1 order by n1;          - 200 rows
returned

select n1, count(*) from t2
group by n1 order by n1;          - 200 rows
returned
```

Now execute the following pair of queries, and examine the execution paths:

```
select * from t1 where n1 = 45;

SELECT STATEMENT Optimizer=CHOOSE (Cost=2
Card=15)
  TABLE ACCESS (BY INDEX ROWID) OF 'T1'
(Cost=2 Card=15)
    INDEX (RANGE SCAN) OF 'T_I1' (NON-
UNIQUE) (Cost=1 Card=15)

select * from t2 where n1 = 45;

SELECT STATEMENT Optimizer=CHOOSE (Cost=15
Card=15)
```

```
TABLE ACCESS (FULL) OF 'T2' (Cost=15
Card=15)
```

You will (probably) find that although the two queries are identical and return the same number of rows (fifteen), the execution paths are different. The query against table *T1* will use the index but the query against table *T2* will use a *table scan* despite the fact that the two tables contain exactly the same data sets. Why is this?

The answer comes from the way the data is scattered. Table *T1* defined using the *trunc()* function will have all fifteen targeted rows located in the same block, but table *T2* defined using the *mod()* function will have its fifteen targeted rows scattered across fifteen different blocks.

Note that there are no skews to the *frequency of use* of the different values (in both tables the values 1 to 200 each appear exactly 15 times), nor are there any skews in the ranges of data (the data distribution is both smooth and gap-free from 1 to 200). Because of this, Oracle has actually been able to produce a perfect estimate of how many different blocks will have to be *visited* to find the data using an index.

```
Path          Blocks
Table T1 by index 1 index block + 1 table block: total 2 block visits
Table T2 by index 1 index block + 15 table blocks: total 16 block visits
```

And in this case, the relevant figures can actually be seen in view *user\_indexes*:

```
select
    table_name          tab,
    num_rows            num_rows,
    avg_data_blocks_per_key d_blocks,
    avg_leaf_blocks_per_key l_blocks,
    clustering_factor    cl_fac
from user_indexes;
```

Tab	Num_rows	D_blocks	L_blocks	Cl_fac
T1	3000	1	1	96
T2	3000	15	1	3000

The starting point for Oracle’s costing algorithm is simply an estimate of how many read requests it will take to acquire the data—and for indexed accesses Oracle makes the very simpleminded assumption that every block visit will require a real physical read request.

However, my tables happen to fill 96 blocks, and my *db\_file\_multiblock\_read\_count* is set to 8. So Oracle calculates that it can scan the entire table by sending out approximately twelve (*ceil(96/8)*) read requests. In fact, this simple “divide high water mark by multi-block read count” is several years out of date—in more recent versions of Oracle, the *db\_file\_multiblock\_read\_count* is adjusted for the purposes of costing, so when you set the parameter to 8, Oracle will



use an adjusted value close to 6.589. which is why my version 8.1.7.4 of Oracle calculated the cost of this scan at 15 read requests ( $96 / 6.589 = 14.57$ ).

By writing a simple pl/sql loop which uses *execute immediate* to change the *db\_file\_multiblock\_read\_count* and explain the execution path for a tablescan of a very large table, you can very easily produce a chart of actual value compared to adjusted value. A few commonly used values are listed below:

Actual	Adjusted
4	4.175
8	6.589
16	10.398
32	16.409
64	25.895
128	40.865

The second, simpleminded assumption made by the optimizer (at least until version 9) is that a single-block read has exactly the same cost as a multi-block read—both in terms of time-required, and consequent CPU usage.

As a result of its two assumptions, Oracle costs the index access path into table *T1* at 2, the index access path into table *T2* as 16, and the tablescan cost of tables *T1* or *T2* as 15. So Oracle uses the index on *T1* (since 2 is less than 15) and the tablescan on table *T2* (since 16 is greater than 15).

Of course, this example was chosen for clarity—and I have suggested as a consequence that there is a simple piece of arithmetic that Oracle uses to determine the cost of the index access. In fact it is likely that there is a decision tree that is used to dictate which of several formulae to use depending on circumstances. After all, the example I have given is extremely simple—it is a single table query, there is just one single column index, the predicate is an equality, the index is a non-compressed B-tree. (The algorithms for bitmap indexes are remarkably different).

But taking this as an important starting point, we can see that the way in which indexed data is scattered or clustered has a most significant impact on Oracle's decision to use an index.

For details of how Oracle's two critical assumptions (block visits = physical read requests, and single-block read = multi-block read) can be subverted by the parameters *optimizer\_index\_cost\_adj* and *optimizer\_index\_caching*, see Tim Gorman's paper referenced below.

### The Join Algorithms

With this outline of how Oracle costs an indexed access path, we can begin to investigate why we might see a sudden increase in table scans as we upgrade Oracle, or change from rule-based to cost-based optimization.

### Nested Loop

The *nested loop* join operates in a way intuitively familiar to the average programmer. To join *tablex* to *tabley*, we select rows from *tablex* then, for each row in *tablex*, we search for rows from *tabley*. Consequently our total cost of the join will be:

- Cost of finding rows in *tablex* plus
- Number of rows found in *tablex* multiplied by cost of one search of *tabley*.

Taking the statistics from the *T1* and *T2* tables above as an example: we have demonstrated that the cost of indexing into table *T1* is 2, and the cost of a tablescan of table *T2* is 15. Assume we write a query that joins *T2* to *T1*, and that we supply some predicate against table *T2* that returns 12 rows. Assume also that (as often happens) Oracle has estimated correctly that our query on *T2* will select 12 rows, then the calculated cost of the *nested loop* join will be:

- 15 (cost of tablescan of *T2*) plus
- 12 (cardinality of select on *T2*) x 2 (cost of one search of *T1* by index)
- Total Cost  $15 + 12 * 2 = 39$

The critical point to note in the costing is that Oracle **does not** make any assumptions about block caching—every one of the 12 searches into *T1* will be treated as if every “logical I/O” will turn into a physical I/O. This is quite likely to be a severe overestimate.

### Hash

If we then consider the *hash* join mechanism, we discover a remarkably different story. In a *hash* join Oracle identifies the table that will return the smaller data set based on the starting predicates. Oracle acquires the data from this table and scatters (hashes) it using a hash function on the join columns into (ideally) an in-memory hash table with a very generous number of hash-buckets that should be sufficient to leave the hash table with no more than one row per bucket.

***We can see that the way in which indexed data is scattered or clustered has a most significant impact on Oracle's decision to use an index.***

Once this is done, Oracle can start to acquire data from the second table (again based on the starting predicates) produce the *hash* value for each row, and check (probe) to see if the corresponding hash-bucket in the hash table contains a matching row from the first table.

Although the two data acquisition steps in a *hash* join could in theory, and completely independently, use an indexed access path, it is frequently the case that both are actually table scans. We will not consider what happens when the data sets are too large for a full in-memory process, but in the ideal situation the cost for such a join is:

- Cost of scanning first table plus
- Cost (usually very small) of building in-memory hash plus
- Cost of scanning second table

Applying this strategy to our *T1* and *T2* tables, and making the same assumptions about the amount of data returned, we would then find the cost of the join to be something like:

- 15 (Cost of scanning T2) +
- 1 (Cost of building a very small hash table) +
- 15 (Cost of scanning T1)
- Total cost = 31

Clearly, for our example, Oracle can produce a smaller cost for the strategy of doing a hash join with table scans. Moreover, this cost is probably a fairly realistic one in some absolute sense, as table scans usually really do convert into physical I/Os, with little benefit from caching.

Many systems that convert from RBO to CBO do so without changing their `db_file_multiblock_read_count`—but this is a critical parameter that can affect the balance of power between *nested loop* and *hash* joins quite dramatically.

### Merge

The *merge* join is often dismissed as irrelevant and inefficient, and indeed it seems to make an appearance fairly rarely in typical systems. However, it should not be overlooked as an option. Remember that hash joins may only be used for equality conditions, not for range-based joins, so there are some cases where merge (or sort/merge) joins are a necessity. If we force our test case into a *merge* join with a `use_merge` hint, we will see a plan like:

```
MERGE JOIN (Cost=41)
  SORT (JOIN) (Cost=17)
    TABLE ACCESS (FULL) OF T2 (Cost=15)
  SORT (JOIN) (Cost=24)
    TABLE ACCESS (FULL) OF T1 (Cost=15)
```

From this we can see that the cost of 41 (rather larger than the default cost of 31 for the *hash* join) can be broken down into

- 15 Cost of scanning first table +
- 2 Cost of sorting first table +
- 15 Cost of scanning second table +
- 9 Cost of sorting second table +
- 1 Small cost (in this case) of merging two sorted results.
- Total cost = 41

In this example, we have to sort both tables before doing the *merge*—in fact if the first data set can be acquired in the correct order—typically because we acquired it through an indexed access path—we wouldn't need to sort it. (Note: the manuals persistently claim that the same would apply to the second data set, but this is not true, the second data set always has to go through a sort—possibly because of the generic need for Oracle to be able to reposition within the second set when doing a range-based join).

Of course, the “quick-fix” parameters will have less impact here. You may be able to reduce the cost of acquiring the data, but possibly the cost of sorting will be the most significant part of the cost—even with this very small data set, 25% of my cost came from sorting. If you need to do a *merge* join, then check where your costs are—if most

*That's right...*  
**no more sleepless nights!**

- Monitoring and Support
- Customized Solutions
- Free monitoring up to 24/7\*

\* The Fine Print: With minimum professional services commitment. See website for details.

**MISSION CRITICAL 24/7**

ORACLE PartnerNetwork CERTIFIED PARTNER

510.352.7300 • 866.352.7300  
 info@mc247.com • www.mc247.com

**LECCO DB Expert**

**Anyone Can Identify Performance Bottlenecks...**

**But Can They OPTIMIZE Them?**

**Download Your FREE Evaluation!**

LECCOTECH  
 Solutions for Oracle, Sybase, SQL Server and DB2 UDB  
 www.leccotech.com

of the cost comes from sorting, your primary optimization method may have to be an increase in your `sort_area_size`.

The `sort_area_size` (like so many other parameters) is, of course, a two-edged sword. How often have you heard the problem: “I’m doing lots of sorts to discs, what should I do?” getting the response: “Increase the `sort_area_size`.”

This is a rule-based “fix” only. Remember that under CBO, this change will reduce the cost of *merge* joins without affecting the cost of *nested loop* or *hash* joins (unless the `hash_area_size` is left to its default of double the `sort_area_size`); so increasing the `sort_area_size` encourages Oracle to do *merge* joins—which may cause more sorting, and make the problem even worse.

As with `db_file_multiblock_read_count`—many sites that switch from RBO to CBO fail to consider the need to re-evaluate this very important parameter.

### What’s New In Oracle 9

Remember that Oracle makes several naïve assumptions about data acquisition. First of all, there is an inherent assumption that a multi-block read will cost (take exactly the same time as) a single-block read. This is probably a little unrealistic. Secondly there is a hard-coded assumption about the probable size of a multi-block read. If you set the system’s `db_file_multiblock_read_count` to 64, Oracle will assume (for the purposes of the calculation) that it will scan a table with a series of multi-block reads of (approximately) 26 blocks each. Again this is likely to be unrealistic, although the actual achievable size of the multi-block read may vary with time.

On top of this, Oracle does not differentiate between the CPU cost of scanning every row in a table and the CPU cost of finding a handful of rows by checking a small number of index entries, and doesn’t allow for the fact that a date comparison performance 1,000,000 times might be more CPU intensive than a numeric comparison performed 1,000,000 times.

Oracle 8.1 started to address these types of issues by introducing a couple of `init.ora` parameters, but Oracle 9 goes further by allowing Oracle to “learn” the physical characteristics of the system. Most significantly, you can instruct Oracle through the `dbms_stats` package to record interval-based performance metrics for:

- The apparent CPU speed
- The typical time for a single-block read request to be serviced
- The typical time for a multi-block read request to be serviced
- The typical size of the multi-block read requests issued by Oracle processes

Representative statistics can be collected for important, characteristic, time periods—such as Monday morning, typical overnight batch, end of month, Friday afternoon. Oracle can then be instructed to use the appropriate set of statistics to model future behavior—factoring in the “correct” multi-block read when calculating the cost of table scans, but attempting to use the specified `db_file_multiblock_read_count` when doing the actual scan—re-arrang-

ing the order of applying predicates if that will reduce the CPU cost of testing data—even allowing the CPU costs of tablescanning and testing millions of rows to swing the balance in favor of an indexed access path.

### Conclusion

Under Oracle 9 DBAs are now able to tell the cost-based optimizer more of the truth about how their system operates. Moreover, the mechanism to do this is very simple and is based on a humanly recognizable aspect of the system. It is inevitably a big migration step to switch optimization strategies on a production system, but an investigation of this new optimization feature should appear high on any DBA’s to-do list.

### Acknowledgements

The material in this white paper and the presentation was originally presented to the UKOUG annual conference in December 2001, and subsequently at OpenWorld in Copenhagen 2002.

### Further Reading

Steve Adams: *Ixora Notes—December 2000: sort\_area\_size*. [www.ixora.com.au](http://www.ixora.com.au)

Tim Gorman: *The search for intelligent life in the Cost-Based Optimizer*. [www.evdbt.com](http://www.evdbt.com)

Wolfgang Breitling: *A look under the hood of the CBO: event 10053*. [www.centrexcc.com](http://www.centrexcc.com)

Jonathan Lewis: *All about bitmap indexes*. [www.jlcomp.demon.co.uk](http://www.jlcomp.demon.co.uk). ▲

*This article is reprinted with permission from the IOUG Live! Conference, April 2003.*



## CALLING ALL AUTHORS

**T**he NoCOUG Journal is always seeking volunteer writers to submit articles for publication.

If it’s something you’ve thought about, now is the time to make it happen. From a one-paragraph Tech Tip to in-depth articles, there’s plenty of room for your talents. Here are a few topics we are particularly interested in:

- Performance tuning
- Real-life problem solving stories
- Oracle developer topics
- Migrating to 9i
- Book reviews (see page 15)
- Java and Oracle
- Oracle and Linux
- Tech tips on miscellaneous Oracle topics

If you have a little bit of time to contribute to the NoCOUG Journal, contact Lisa Loper at [journal@nocoug.org](mailto:journal@nocoug.org) or call (415) 344-0500 ext. 42. ▲



# Many Thanks to Our Sponsors

NoCOUG would like to acknowledge and thank our generous sponsors for their contributions. Without this sponsorship, it would not be possible to present regular events while offering low-cost membership dues. If your company is able to offer sponsorship at any level, please contact NoCOUG President Roger Schrag at [rschrag@dbspecialists.com](mailto:rschrag@dbspecialists.com). ▲

*Long-term event sponsorship:*

- LOCKHEED MARTIN
- CHEVRONTEXACO
- ORACLE CORP.
- PG&E

## Thank you! Year 2003 Gold Level Support Vendors:

- Database Specialists, Inc.
- Embarcadero Technologies
- Golden Gate
- LECCOTECH
- Mission Critical 24/7
- Quest Software, Inc.

For information about our Gold Level Vendor Program, contact the NoCOUG vendor coordinator via email at: [vendor\\_coordinator@nocoug.org](mailto:vendor_coordinator@nocoug.org)



## TREASURER'S REPORT

Judy Lyman, Treasurer

### Beginning Balance

July 1, 2003 \$ 58,958.71

### Revenue

Membership Dues	3,482.05
Meeting Fees	1,180.00
Vendor Receipts	4,000.00
Advertising	-
Interest	22.67
<b>Total Revenue</b>	<b>\$ 8,684.72</b>

### Expenses

Regional Meeting	5,952.34
Journal	5,329.68
Membership	440.85
Administration	980.00
Website	75.00
Board Meeting	552.50
Miscellaneous	-
IRS	130.00
FTB Tax	-
<b>Total Expenses</b>	<b>\$ 13,460.37</b>

### Ending Balance

September 30, 2003 \$ 54,183.06

# The Fourth “R”—Rman

By Michael S. Abbey, The Pythian Group

That’s what I said, the “Fourth R”—reading, (w)riting, (a)rithmetic, and now “rman.” Many of us finally broke down and had a look at rman when our repositories got so big they were impossible to back up. We would try to run a traditional hot backup every 24 hours, and it would take longer than 24 hours to run. Once we implemented rman backups, the run time went down from 27.5 hours to 2! Remarkable? I think so. In this paper, we are going to have a look at rman setup and offer some basics about using this remarkable tool.

## Setup the Rman Catalog

There are two ways to run rman backups.

1. *Catalog-based rman* where metadata for the backups is held in a catalog in a separate instance setup for this specific purpose.

2. *Nocatalog rman* where all this metadata is stored in the control file(s).

This is usually the first decision we make when considering the rman solution. The documentation suggests that catalog-based rman backups offer more flexibility and are the only way to go. That’s good enough for me. The catalog maintains metadata about your rman backups. Whichever route you decide to take, the control file also holds similar information. After we look at creating the rman catalog database, we will make a user to hold the catalog.

## The Catalog Instance

Once the catalog route is chosen, you create a separate instance to hold this metadata using code similar to the following, after connecting to the new instance as a user with *sysdba*:

```
1 set echo on
2 startup nomount
3 create database rcat
4 controlfile reuse
5 maxlogfiles 16
6 maxlogmembers 2
7 maxdatafiles 30
8 maxinstances 1
9 maxloghistory 1000
10 character set we8iso8859p1
11 datafile '/u02/oradata/rcat/system01.dbf' size 200m
12 logfile group 1 ('/u01/oradata/rcat/log1_g1.dbf',
13                '/u02/oradata/rcat/log2_g1.dbf') size 20m,
14 group 2
15                ('/u03/oradata/rcat/log1_g2.dbf',
16                '/u04/oradata/rcat/log2_g2.dbf') size 20m,
17 group 3
18                ('/u01/oradata/rcat/log1_g3.dbf',
19                '/u02/oradata/rcat/log2_g3.dbf') size 20m;
19 create rollback segment rb_temp tablespace system
20 storage (initial 50k next 50k minextents 60 maxextents 60);
21
22 set echo off
23
24 @?/rdbs/admin/catalog
25 @?/rdbs/admin/catproc
26 @?/rdbs/admin/catrep
27 @?/rdbs/admin/catexp
28 @?/rdbs/admin/catldr
29 @?/rdbs/admin/dbmsutil
30 @?/rdbs/admin/prvtutil.plb
31 @?/rdbs/admin/dbmsql
32 @?/rdbs/admin/prvtsql.plb
33 @?/rdbs/admin/dbmsdefr
```

```
34 @?/rdbs/admin/prvtdefr.plb
35 @?/rdbs/admin/catqueue
36 @?/rdbs/admin/dbmsaqad
37 @?/rdbs/admin/prvtaqad.plb
38
39 connect system/manager
40
41 @?/rdbs/admin/catdbsyn
42 @?/sqlplus/admin/pupbid.sql
```

A few points of discussion are required before moving on.

## Line(s) Pertinent Points

- 3–17 Create the database, mainly the SYSTEM tablespace, the control files and the specified pool of online redo logs. Notice the naming convention I like to use for the redo logs—member number of group number—where the first and second members of group 1 are called log1\_g1.dbf and log2\_g1.dbf. The instance character set is specified in line 10. We have used character sets as basis as us7ascii and as advanced as utf8.
- 19–20 Create the first rollback segment other than system. The system rollback segment is part of the create database macro. Having at least one non-system rollback segment allows us to perform rollback segment maintenance while the database is being used.
- 24–27 Run the suggested assortment of administration scripts. There are many admin scripts that can be run, but this list I have assembled usually installs the appropriate functionality to support most instance operations.
- 39 Connect as SYSTEM and build the PRODUCT\_USER\_PROFILE table.

## The Catalog Instance Infrastructure

This piece is nothing new to techies who have been creating databases for years. This is where you create the locally managed temporary tablespace, a dedicated tablespace for rollback segments, the first few non-SYSTEM rollback segments, and a tablespace to hold the rman catalog itself.

```
create temporary tablespace loc_temp tempfile
'/u01/oradata/rcat/loc_temp01.dbf' size 100m extent management
local uniform size 200k;

create tablespace rman datafile '/u01/oradata/rcat/rman01.dbf'
size 200m extent management local autoallocate segment space
management auto;

create tablespace rollback datafile
'/u01/oradata/rcat/rollback01.dbf' size 400m default storage
(initial 1m next 1m);

create rollback segment rbs01 tablespace rollback storage (initial
1m next 1m minextents 199 maxextents 199);

create rollback segment rbs02 tablespace rollback storage (initial
1m next 1m minextents 199 maxextents 199);

create user rman identified by XXXXX temporary tablespace loc_temp
default tablespace rman quota unlimited on rman;
```

```
grant recovery_catalog_owner to rman;
grant connect,resource to rman;
```

## Logging into Rman

Most of the time, the method of invocation for rman is made up of two parts, each with their own login credentials. Let's pick apart the command `rman target / catalog rman/XXXXX@rcat`.

1. The keyword `target` refers to the database which will be backed up. Suppose you want to back up your production database, with the ORACLE\_SID of `prod`. Once an OS authenticated user is established, the rman login can be satisfied using the forward slash (/).

2. The keyword `catalog` refers to the Oracle Net defined instance where the rman metadata resides. The `@rcat` connect string is resolved by a local tnsnames.ora alias or a database registration through Oracle Names.

## Creating the Catalog

Just like about everything else in the rman setup, this is accomplished very easily using the create catalog macro as follows:

```
/home/oracle> rman target / catalog rman/XXXXX@rcat

Recovery Manager: Release 9.2.0.2.0 - 64bit Production
Copyright (c) 1995, 2002, Oracle Corporation. All rights reserved.

RMAN-06005: connected to target database: REPORT1 (DBID=150508596)
RMAN-06008: connected to recovery catalog database
RMAN-06428: recovery catalog is not installed

RMAN> create catalog;

RMAN-06431: recovery catalog created

RMAN>
```

## Connecting and Registering a Database

Databases are registered with the rman catalog before they are backed up as follows:

```
RMAN> register database;

RMAN-03022: compiling command: register
RMAN-03023: executing command: register

RMAN-08006: database registered in recovery catalog
RMAN-03023: executing command: full resync
RMAN-08002: starting full resync of recovery catalog

RMAN-08004: full resync complete
RMAN>
```

We are now ready to backup a database using rman.

## Rman Backup Scripts

We can cover the most ground in the next few sections by starting with a look at a simple rman full database backup script then take it from there. The full database is the most common type of rman backup, though we often back up tablespaces and datafiles on their own.

## Full Database

This is always the first type of backup we become familiar with. Unless you have the fortune or mis-fortune of managing multi-terabyte databases, the full database incremental level 0 is the run with which you will cut your rman molar.

```
1 rman target / catalog rman/XXXXX@rcat > $LOG_FILE << EOF
2 sql 'alter system archive log current';
3 run
4 {
5 allocate channel ch1 type disk;
6 allocate channel ch2 type disk;
7 set limit channel ch1 kbytes=200000;
8 set limit channel ch2 kbytes=200000;
9 backup incremental level $(LVL)
10 (database format '${BDIR}/%d_level$(LVL)_$(DATE)_%s_U.bak');
11 sql 'alter system archive log current';
12 copy current controlfile
13 to '${BDIR}/${ORACLE_SID}_level$(LVL)_$(DATE)_control01.bak';
14 backup current controlfile for standby
15 format '${BDIR}/%d_SCF_%U.bak';
16 release channel ch1;
17 release channel ch2;
18 }
19 EOF
```

Now the fun begins—let's pick apart the script to illustrate some concepts about rman.

## Line(s) Pertinent Points

- 1 The whole rman session is sent to a file previously defined as LOG\_FILE.
- 2 This is the stake in the ground that determines what archived redo logs are required for recovery. An rman backup is valid only if the archived redo logs written while it was running are available. This is nothing new—it was the same case for traditional hot backups. Notice how sql commands are enclosed in single quotes and have no terminating semi-colon as they do in SQL\*Plus.
- 3–4 The rman run-block start. Certain commands must be inside a run-block, where the commands are grouped together between an opening [and a closing].
- 5–6 Manual allocation of channels for rman operations is compulsory with `8i` and optional with `9i`. It is shown here to assist the learning process. Notice how the lines are terminated with a semi-colon.
- 7–8 This figure determines the size of each backup set piece in kilobytes. The 200,000 figure will end up writing pieces sized at just under 200Mb.
- 9–10 The heart of the exercise. We are calling for an incremental backup, with the level passed on the command line to the rman script. The BDIR, LVL, and DATE variables would have been set prior to the call to rman. The %d informs rman to start each backup set piece name with the uppercase ORACLE\_SID. The %s and %U instruct rman to generate a unique name for each backup piece.
- 11 The end-stake-in-the-ground.
- 12–13 Common practice whatever method is being used to back up—making a copy of the controlfile.
- 14–15 Another common practice is to make a standby controlfile during a backup. If this run ends up being used to build a standby database, this file, deliberately written at this time, can make the standby build easier.

Now, we take it down a notch, looking at a more granular backup scenario.

## Tablespace and Datafile Backup

You might remember the traditional approach to writing hot backups where we placed a tablespace in backup mode,

copied its one or more files, then took it out of backup mode when done. You do not need to do that with rman, obvious when perusing the following listings.

```

1 RMAN> run
2 2> {
3 3> allocate channel ch1 type disk;
4 4> backup incremental level 1
5 5> tablespace users
6 6> format '/oracle/rman/prd/ah_tbs_USERS_${DATE}.bak';
7 7> release channel ch1;
8 8> }
9
10 allocated channel: ch1
11 channel ch1: sid=78 devtype=DISK
12
13 Starting backup at 17-FEB-03
14 channel ch1: starting incremental level 1 datafile backupset
15 channel ch1: specifying datafile(s) in backupset
16 input datafile fno=00005
name=/u02/oradata/prod/prod_users01.dbf
17 channel ch1: starting piece 1 at 17-FEB-03
18 channel ch1: finished piece 1 at 17-FEB-03
19 piece handle=/oracle/rman/prd/adhoc_tbs_USERS.bak comment=NONE
20 channel ch1: backup set complete, elapsed time: 00:00:03
21 Finished backup at 17-FEB-03
22
23 released channel: ch1
24
25 RMAN> run
26 2> {
27 3> allocate channel ch1 type disk;
28 4> backup incremental level 1
29 5> datafile '/u02/oradata/prod/prod_users01.dbf'
30 6> format '/oracle/rman/prd/ah_dtf_users01_${DATE}.bak';
31 7> release channel ch1;
32 8> }
33
34 allocated channel: ch1
35 channel ch1: sid=78 devtype=DISK
36
37 Starting backup at 17-FEB-03
38 channel ch1: starting incremental level 1 datafile backupset
39 channel ch1: specifying datafile(s) in backupset
40 input datafile fno=00005
name=/u02/oradata/prod/prod_users01.dbf
41 channel ch1: starting piece 1 at 17-FEB-03
42 channel ch1: finished piece 1 at 17-FEB-03
43 piece handle=/oracle/rman/prd/adhoc_datafile_users01.bak
comment=NONE
44 channel ch1: backup set complete, elapsed time: 00:00:03
45 Finished backup at 17-FEB-03
46 released channel: ch1
47
48 RMAN>

```

There are a couple of points worth making about the preceding code before moving onto the next topic.

### Line(s) Pertinent Points

- 4–6 The tablespace backup command. You must have a previous incremental level 0 that has established a starting point upon which this level 1 is overlaid. Notice how, since I am backing up one single container (tablespace), that I have used a meaningful name for the backup piece, imbedding the date of the backup to establish uniqueness by file name.
- 3,7 Even though *9i* allows for stored channel configuration, I have manually allocated and released resources, mandatory with *8i*.
- 16 The *fno* handle is a familiar rman backup entity, matching a file's *file\_id* column value from *dba\_data\_files* or *file#* from *v\$datafile*.
- 28–30 Same story as the tablespace backup; use a meaningful name, with the DATE environment variable to establish uniqueness.

Technical practitioners in the database world sometimes

talk about backup and recovery as if they are equal partners in their routines. Suppose there were 100 units of importance in the backup and recovery arena, how much of those units does one attribute to the former and how many to the latter? In my little world, I assign 20 to backup and 80 to recovery. A non-tested backup strategy is nothing more than mastering the syntax allowing you to write backups. Until those backups are tested, to use a baseball analogy, you aren't even close to rounding first! Let's examine a few recovery scenarios, then I will tease you with some more material that will be covered during the Live! presentation.

### Rman Recovery Scripts

We briefly looked at database, tablespace, and datafile backup. So let's do the same for recovery. The recovery exercise is two-step.

1. The restore phase builds database files from the backup set pieces.
2. The recovery phase brings the restored entities up to current point in time (tablespace and datafile) or past point in time (full database).

The backup pieces that are fed to the recovery script must be in the location to which they were written during the rman backup and must not be compressed. First, however, the complete database recovery.

### Database Recovery

It's unbelievable how simple and straightforward this turns out to be as shown next.

```

startup mount
run
{
  allocate channel ch1 type disk;
  allocate channel ch1 type disk;
  restore database;
  recover database;
  release channel ch1;
  release channel ch2;
}
shutdown;
startup;

```

The feedback from rman as the restore and recovery progresses displays:

1. Information about what is being restored where by channel, for example:

```

RMAN-08089: channel ch1: specifying datafile(s) to restore from
backup set
RMAN-08523: restoring datafile 00001 to
/u01/oradata/prd/system01.dbf
RMAN-08523: restoring datafile 00002 to
/u04/oradata/prd/rollback01.dbf
...
RMAN-08089: channel ch2: specifying datafile(s) to restore from
backup set
RMAN-08523: restoring datafile 00003 to
/u04/oradata/prd/tools01.dbf
RMAN-08523: restoring datafile 00004 to
/u02/oradata/prd/index01.dbf
RMAN-08523: restoring datafile 00005 to
/u03/oradata/prd/notes01.dbf

```

2. The recovery phase and the application of archived redo logs found on disk where expected and automatically applied:

```

RMAN-03022: compiling command: recover(4)
RMAN-06050: archivelog thread 1 sequence 51 is already on disk as
file /arch/prd/arch_1_51.arc

```

```

RMAN-06050: archivelog thread 1 sequence 52 is already on disk as
file /arch/prd/arch_1_52.arc
RMAN-03023: executing command: recover(4)
RMAN-08017: channel chl: starting archivelog restore to default
destination
RMAN-08022: channel chl: restoring archivelog
RMAN-08510: archivelog thread=1 sequence=50

```

### 3. The extraction of archived redo logs from backup pieces, when required for recovery but not in the archived redo log location any more:

```

RMAN-08511: piece
handle=/oracle/rman/prd/PRD_0ge8trq3_1_1_level1_112802.arc
tag=null params=NULL
RMAN-08024: channel chl: restore complete
RMAN-08515: archivelog filename=/arch/prd/arch_1_50.arc thread=1
sequence=50
RMAN-08515: archivelog filename=/arch/prd/arch_1_51.arc thread=1
sequence=51
RMAN-08515: archivelog filename=/arch/prd/arch_1_52.arc thread=1
sequence=52

```

### 4. Details about the completion of the recovery and opening of the database:

```

RMAN-08055: media recovery complete

RMAN-03022: compiling command: recover(4)

RMAN-03022: compiling command: release
RMAN-03023: executing command: release
RMAN-08031: released channel: chl

RMAN-03022: compiling command: release
RMAN-03023: executing command: release
RMAN-08031: released channel: chl

RMAN> shutdown

RMAN-06404: database dismounted
RMAN-06402: Oracle instance shut down

RMAN> startup

RMAN-06193: connected to target database (not started)
RMAN-06196: Oracle instance started
RMAN-06199: database mounted
RMAN-06400: database opened

```

### Tablespace Recovery (Datafile Recovery)

Not much different than the database recovery picture, except for the fact that you are recovering part rather than all of the instance.

```

run
{
  allocate channel chl type disk;
  restore tablespace users;
  recover tablespace users;
  release channel chl;
}

```

It's almost too good to be true—the whole session is managed by the recovery catalog, and the information it stores about what was backed up when and the operating system files that make up the backup pieces. The feedback from rman tracks the progress of the operations, up to the tablespace being back online and ready for use.

```

RMAN-03022: compiling command: allocate
RMAN-03023: executing command: allocate
RMAN-08030: allocated channel: chl
RMAN-08500: channel chl: sid=23 devtype=DISK

RMAN-03022: compiling command: restore

RMAN-03022: compiling command: IRESTORE
RMAN-03023: executing command: IRESTORE
RMAN-08016: channel chl: starting datafile backupset restore
RMAN-08502: set_count=4 set_stamp=479207186 creation_time=12-MAR-
03

```

```

RMAN-08089: channel chl: specifying datafile(s) to restore from
backup set
RMAN-08523: restoring datafile 00005 to
/u02/oradata/prd/users01.dbf
RMAN-08023: channel chl: restored backup piece 1
RMAN-08511: piece handle=/oracle/rman/prd/ah_tbs_USERS_030312.bak
tag=null params=NULL
RMAN-08024: channel chl: restore complete
RMAN-03023: executing command: partial resync
RMAN-08003: starting partial resync of recovery catalog
RMAN-08005: partial resync complete

```

```
RMAN-03022: compiling command: recover
```

```
RMAN-03022: compiling command: recover(1)
```

```
RMAN-03022: compiling command: recover(2)
```

```
RMAN-03022: compiling command: recover(3)
```

```
RMAN-03023: executing command: recover(3)
```

```
RMAN-08054: starting media recovery
```

```
RMAN-08055: media recovery complete
```

```
RMAN-03022: compiling command: recover(4)
```

```
RMAN-03022: compiling command: release
```

```
RMAN-03023: executing command: release
```

```
RMAN-08031: released channel: chl
```

Not surprisingly, rman knows what datafiles belong to what tablespace—another of many bells and whistles that separate it from more traditional and familiar backup approaches. The recovery of one or more datafiles is the same as the tablespace recovery code, except the name of one or more datafiles is specified instead of the tablespace details in the last listing that are underlined. The catalog instance we created towards the top of this paper is simply an Oracle database, holding the rman schema—nothing special about this schema, it's just like any other you are used to managing.

Before calling it a wrap, I am going to spend a bit of time looking at some catalog repository management that you should know about from day one. Cleaning up unwanted information in the rman catalog is initiated by a simple set of commands, passed to rman from its familiar RMAN> prompt.

### Rman Catalog Cleanup

This is a large topic; a discussion of rman, however basic or advanced, cannot ignore catalog cleanup. As smart practitioners, we like to clean up after ourselves. In the rman catalog, it is useless to keep records about backups that are no longer available; the catalog repository grows as do other Oracle schemas. Suppose you have a retention period of 15 days on your backups; it is therefore worthless to let the rman catalog keep a record of any backups older than 15 days. Deleting records from the catalog is a two-step approach:

1. Checking the availability of the backup piece records you want to eventually delete, using the *crosscheck* command.
2. Once pieces are marked as *expired*, wiping their records from the catalog with *delete expired*.

```
RMAN> allocate channel for maintenance type disk;
```

```
RMAN-03022: compiling command: allocate
```

```
RMAN-03023: executing command: allocate
```

```
RMAN-08030: allocated channel: delete
```

```
RMAN-08500: channel delete: sid=12 devtype=DISK
```

```
RMAN> crosscheck backup completed before 'sysdate-18';
```



```

RMAN-03022: compiling command: XCHECK
RMAN-03024: performing implicit full resync of recovery catalog
RMAN-03023: executing command: full resync
RMAN-08002: starting full resync of recovery catalog
RMAN-08004: full resync complete
RMAN-08074: crosschecked backup piece: found to be 'EXPIRED'
RMAN-08517: backup piece handle=/oracle/rman/prd/v7dms26s_3_1.arc
recid=10437 stamp=460196212
RMAN-08074: crosschecked backup piece: found to be 'EXPIRED'
RMAN-08517: backup piece handle=/oracle/rman/prd/v7dms26s_4_1.arc
recid=10438 stamp=460196287
RMAN-08074: crosschecked backup piece: found to be 'EXPIRED'
RMAN-08517: backup piece handle=/oracle/rman/prd/v7dms26s_5_1.arc
recid=10439 stamp=460196362
. . .
RMAN> delete expired backup;

RMAN-03022: compiling command: delete expired
RMAN-03023: executing command: delete expired
RMAN-08073: deleted backup piece
RMAN-08517: backup piece
handle=/u04/hot/rman/REPORT1/REPORT1_03020718_control01.bak
recid=4902 stamp=485373602
RMAN-08073: deleted backup piece
RMAN-08517: backup piece
handle=/u04/hot/rman/REPORT1/5leesdl4_1_1_03020718.arc recid=4903
stamp=485373605
RMAN-08073: deleted backup piece
RMAN-08517: backup piece
handle=/u04/hot/rman/REPORT1/REPORT1_level1_030207_2235_U5reesrj8_
_1.bak recid=4909 stamp=485387880
RMAN-08073: deleted backup piece
. . .
RMAN-03023: executing command: partial resync
RMAN-08003: starting partial resync of recovery catalog
RMAN-08005: partial resync complete

```

## Wrapup

This paper is designed to get you started using rman and to familiarize you with its operations and a few of its nuances. Just about everything I have talked about here is

applicable to both Oracle8i and Oracle9i. What you should do now, is dive head first into rman backup and recovery, using it in parallel with your legacy backup routines. Once you experience the full blown power of rman, you will wonder why you didn't get started earlier. In the presentation at Live!, I will be discussing a few more topics that I have not touched on here, including:

- Building a standby database from an rman backup
- Restarting a failed rman backup (Oracle9i only)
- Recovering a database using different locations for the files than were written to the backup set
- Archived redo log management during the backup and the recovery phase
- Validating an rman backup

## About the Author

*Michael S. Abbey lives and works out of Canada's capital, Ottawa. He has been working with the Oracle software exclusively since the '80s, having first cut his teeth as a DBA on release 6.0.27.9.4. He has co-authored a number of works in the Oracle Press series, among which the most successful was Oracle9i: A Beginner's Guide, as well as beginner's guides on Oracle7, Oracle8, and Oracle8i. Michael has been very active in the user group arena since the early 1990s, especially involved in the IOUG, the sponsor of this conference. His forte is installation, upgrades, backup and recovery, and issues related to the overall management of large information repositories.*

# NoCOUG Fall Conference (continued from pg. 27)

between XML style sheets. Even after changing the SQL statements into the ISO/ANSI format, developers still need to be concerned with the performance needs of the SQL statements running in an Oracle database server. This session will educate the participants so they can determine when and how to start using the new constructs as well as potential performance issues.

## How STATSPACK Was Used to Solve Common Performance Issues

Brian Hitchcock, *Sun Microsystems*

During the last year four specific performance issues came up and STATSPACK data was reviewed in each case. In two of the cases, the STATSPACK output did not directly identify the performance issue. In the two other cases, the STATSPACK output did result in significant performance improvements. Even when the output did not directly lead to the cause of the performance issue, it did verify that many possible performance issues did not exist in the instance. For the two cases where the STATSPACK output did lead to the root cause, the details of how this was done are examined. The method of calculating the total run time and the wait event time are detailed. Once the root cause was identified, how the performance issue was resolved is discussed.

## Working with Partitioned Tables—The Unpleasant Details

Brian Hitchcock, *Sun Microsystems*

Partitioned tables provide several significant performance benefits. They also present several administrative headaches that will be reviewed from actual experience. The challenges for the DBA become apparent as the process to move from one partitioning scheme to another is reviewed. The details of such a move are covered as well as the mistakes made and how they were resolved. Such details include the syntax of the SQL to split existing partitions, fixing the minor mistakes made when the initial partitioning scheme was set up and moving the tablespaces as the table partitioning scheme is changed. All of which came from the experience of supporting an upgrade of a major third-party application. ▲

**Thank you to  
PG&E,  
our meeting sponsor.**

# NoCOUG

Northern California Oracle Users Group

## Membership Application for 2004

Your membership extends from January 1 through December 31, 2004

Code: NC112001-\_\_\_\_\_

Name:		
Title:	Work Phone:	
Company:	Mail Stop:	
Address:		
City:	State:	ZIP:
Email:		
URL:		

Do you want your mailing information made available on a third-party mailing list?  YES  NO

Do you want your email address added to the NoCOUG Only distribution list?  YES  NO

### Oracle Product Profile

Please tell us a little about the systems and products you use most often:

Computer platform: \_\_\_\_\_ RDBMS version: \_\_\_\_\_ OS: \_\_\_\_\_

Please check each product used:

- Oracle Developer®     Oracle Designer®     Oracle Discoverer®     J Developer     BC4J®  
 Oracle Applications:    \*Rel# \_\_\_\_\_     Web-DB/Oracle Portal®     Oracle Application Server

### Membership Dues

Membership Type	Includes	Qty	Rate		Amount
NoCOUG Individual Membership	Quarterly newsletter, roster and one meeting attendee		70.00	=	
NoCOUG Corporate Membership	Six individual memberships, plus four additional newsletters		450.00	=	
	Total Membership Fee Enclosed				

NoCOUG and the International Oracle Users Group (IOUG) are alliance partners. We encourage you to learn more about IOUG by visiting the IOUG website at <http://www.ioug.org>. NoCOUG members get a \$15 discount on IOUG membership for 2004. To get the discount, choose Northern California Oracle Users Group in the list of user groups when filling out your IOUG membership application.

### Payment

**On site** NoCOUG Memberships **must** be paid by check. Online registration is also available at [www.nocoug.org](http://www.nocoug.org). Enclosed check # \_\_\_\_\_. **Make checks payable in U.S. Dollars to: NoCOUG, P.O. Box 3282, Danville, CA 94526**

Signature: \_\_\_\_\_ Date: \_\_\_\_\_ NoCOUG Tax ID: 94-3237851

# NoCOUG Fall Conference

Thursday, November 13, 2003

## Session Descriptions

For more detailed descriptions and up-to-date information, see [www.nocoug.org](http://www.nocoug.org)

### TRACK 1

#### **Managing an Oracle Optimization Project**

Gary Goodman, *Hotsos*

Many business leaders lack the background to effectively manage Oracle performance issues. This presentation describes a proven method for targeting and prioritizing optimization investments based on profit, ROI, and cash flow, not system metrics. We will explore many of the common mis-steps in dealing with Oracle optimization and explain why they often fail while wasting time and money.

#### **Performance Problems from the Field**

Gary Goodman, *Hotsos*

This will be a very interactive presentation where we will review three actual case studies from our field experience at Hotsos. We will examine our client's performance problems and how a user-action-based method was utilized to rapidly determine and repair the root cause issues. We will use these case studies to show the effectiveness of using 10046 level 8 trace data as the basis for Oracle optimization.

#### **What a DBA Needs to Know About Bitmap Indexing in Oracle to Retrieve Data Quickly (Part 1)**

Vilin Roufchaie, *Cingular Wireless*

(This presentation is for Data Warehouse designers, as well as performance DBAs and capacity planners). Bitmapped indexing is a query execution optimization technique predominantly used in Data Warehousing and DSS environments. The details of bitmapped indexing will be discussed and fully illustrated.

#### **What a DBA Needs to Know About Oracle's "Star Transformation Query" Processing in a Star Schema (Part 2)**

Vilin Roufchaie, *Cingular Wireless*

"Star Transformation Query" (STQ) was launched by Oracle Corporation in Oracle 8 as a result of "Star Query's" (SQ) inability to efficiently and cost effectively address the needs of certain Star Schemas due to excessive number of join operations it had to perform between dissimilar columns from dimension and fact tables. In this presentation I will delve into the heart of STQ and demonstrate how it works and what conditions would warrant its utilization to expedite query processing, followed by illustration of a real-world benchmark and explain plan details. Time permitting,

I will detail the pros and cons of STQ and SQ, and demonstrate how these two methods can actually complement each other.

#### **A Quantitative Basis for Measuring Data Quality Using Metadata**

Bonnie O'Neil, *Apex Solutions, Inc.*

This presentation will explore the relationship between metadata, data standardization and data quality. Metadata can act as a source of record to baseline data content and identify specific key points in the information supply chain to audit and validate specific data content. Performing data audits within the information supply chain can result in the rapid identification and management of data irregularities. The development of statistics related to data distribution and domain values along with historical trending of data values offers a mechanism to quantitatively determine data validity and accuracy under various conditions. Data management teams can maintain credibility with their analytical or end user communities by creating certified data loads based on business-defined metadata attributes. Developing automated mechanisms during acquisition and ETL processes for auditing and validating data is crucial to ensure minimal impact on the end-to-end processes supported.

### TRACK 2

#### **Frequently Asked Questions: A smorgasbord of common questions and problems received by World Wide Support—and how to resolve them. (Part 1)**

David Austin, *Oracle Corporation*

The presentation covers a variety of topics that, based on statistics compiled by World Wide Support, cause problems for many database administrators. A best practice or working solution is provided for each problem, in some cases, some of which may be release dependent. The topics are not offered in any particular order and the audience is encouraged to select from available topics and to add their own contributions to the discussion of the problems and possible solutions.

#### **Help! I Got a Request for ANSI SQL – What Is It, and What Do You Do With It?**

Walter Guerrero, *Computer Associates*

If you are starting to work with XSQL or SQLX, this is a presentation you will not want to miss. As companies deploy more XML and exchange information across many different database types, they will need to implement ISO standards within their business models. SQL statements must then move away from proprietary format and fit within the ISO/ANSI standards. These ISO/ANSI standards provide a more readable and portable format that can be used by different database engines. Oracle is making the move for Oracle developers to start using ISO/ANSI SQL constructs in the creation of SQL statements to exchange data

(continued on page 25)

# NoCOUG Fall Conference

Thursday, November 13, 2003

**Location:** The NoCOUG Fall Conference will be held at Pacific Gas & Electric headquarters at 77 Beale Street in downtown San Francisco. See complete directions to the meeting location at [www.nocoug.org](http://www.nocoug.org).

08:00AM Registration and Continental Breakfast

09:00–09:30 General Session and Welcome

09:45–10:45 **Parallel Sessions #1**

**Track 1:** *Managing an Oracle Optimization Project* by Gary Goodman, Hotsos

**Track 2:** *Frequently Asked Questions: A smorgasbord of common questions and problems received by World Wide Support—and How to Resolve Them (Part 1)* by David Austin, Oracle Corporation

10:45–11:15 **Break**

11:15–12:15 **Parallel Sessions #2**

**Track 1:** *Performance Problems from the Field* by Gary Goodman, Hotsos

**Track 2:** *Frequently Asked Questions: A Smorgasbord of Common Questions and Problems Received by World Wide Support – and How to Resolve Them. (Part 2)* by David Austin, Oracle Corporation

12:15–1:15 **Lunch**

1:15–2:15 **Parallel Sessions #3**

**Track 1:** *What a DBA Needs to Know About Bitmap Indexing in Oracle to Retrieve Data Quickly (Part 1)* by Vilin Roufchaie, Cingular Wireless

**Track 2:** *Help! I Got a Request for ANSI SQL—What Is It, and What Do I Do With It?* by Walter Guerrero, Computer Associates

2:15–2:30 **Break**

2:30–3:30 **Parallel Sessions #4**

**Track 1:** *What a DBA Need to Know About Oracle's "Star Transformation Query" Processing in a Star Schema (Part 2)* by Vilin Roufchaie, Cingular Wireless

**Track 2:** *How STATSPACK Was Used to Solve Common Performance Issues* by Brian Hitchcock, Sun Microsystems

3:15–3:45 **Break and Raffle**

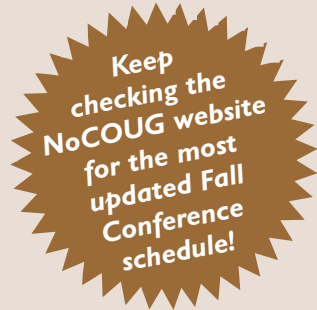
4:00–5:00 **Parallel Sessions #5**

**Track 1:** *A Quantitative Basis for Measuring Data Quality Using Metadata* by Bonnie O'Neil, Apex Solutions, Inc.

**Track 2:** *Working with Partitioned Tables—The Unpleasant Details* by Brian Hitchcock, Sun Microsystems

5:00 -?? NoCOUG Networking and Happy Hour at Beale Street Bar & Grill, 133 Beale Street

Cost: \$40 admission fee for non-members. Members free. Includes lunch voucher.



**View session descriptions on page 27.**

**RSVP online at [www.nocoug.org/rsvp.htm](http://www.nocoug.org/rsvp.htm)**

**NoCOUG**

P.O. Box 3282

Danville, CA 94526

FIRST-CLASS MAIL  
U.S. POSTAGE  
PAID  
SAN FRANCISCO, CA  
PERMIT NO. 11882