

# nocooug

## JOURNAL

Official Publication of the Northern California Oracle Users Group

VOL. 17, No. 2 · MAY 2003

\$15



### PRESIDENT'S MESSAGE

## Keeping Your Oracle Skills Current



Roger Schrag

People frequently ask me how they can advance their Oracle career. Some folks want to know how to keep up to date on new Oracle technology. Others are interested in the non-technical skills required to be a successful Oracle engineer. These questions don't have simple answers, but I am always happy to share what I've learned after 14 years in the trenches. So, let's talk about keeping your Oracle skills current.

Oracle technology is hard to teach in a classroom, for several reasons. Oracle products and best practices change constantly. (Remember SQL\*Forms? CONNECT INTERNAL?) Rapid change can cause obsolescence in courseware prepared only a few years ago. In an attempt to keep up, course materials are constantly being updated. With all of the new information, it is challenging to create consistently high quality curriculum—by the time the material is fine-tuned, some of it is out of date.

*(continued on page 4)*

## Oracle Temporary Tables, Temporary Segments & Temporary Tablespaces—Part I



Hamid Minoui

by Hamid Minoui, UPS-Supply Chain Solutions

### Abstract

Part I of this paper will attempt to describe the usage, characteristics and management of Oracle's temporary tables and various temporary segments and the association between them. The focus of Part II is to address the role and intended usage of tempo-

*(continued on page 6)*

### TABLE OF CONTENTS

President's Message .....	1
Oracle Temporary Tables .....	1
From the Editor .....	2
NoCOUG Board .....	2
User Groups.....	2
Publication and Submission Format.....	2
Advertising Rates.....	2
Board Member Spotlight.....	5
Sponsorship Appreciation.....	11
Treasurer's Report.....	11
Another Example of Interpreting Wait Events to Boost System Performance.....	12
Thank You, Oracle! .....	16
Jonathan Lewis Event at ChevronTexaco .....	16
SQL Server 2000 and Oracle 9i.....	17
Identifying and Analyzing Key Factors Affecting System Performance .....	19
NoCOUG Spring Conference Session Descriptions.....	26
NoCOUG Spring Conference Schedule .....	28
—ADVERTISERS—	
Quest Software.....	3
LECCOTECH.....	6
Mission Critical 24/7.....	6
BMC Software .....	10
Database Specialists .....	14
Embarcadero Technologies.....	14

# From the Editor

NoCOUG exists for the education and representation of the users of Oracle Corporation's database and tools software, and those who contribute to the NoCOUG Journal help achieve that vision.

If you would like to get involved, now is a great time. There are a number of short- and long-term assignments that you can take on, depending on your time and interests. Just get in touch with me at [journal@nocoug.org](mailto:journal@nocoug.org) or call (415) 344-0500, ext. 42. I look forward to hearing from you. ▲

—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](mailto: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](mailto: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](mailto:rschrag@dbspecialists.com)

### Vice President

Darrin Swan, LECCOTECH  
[darrin@leccotech.com](mailto:darrin@leccotech.com)

### Treasurer/Secretary

Judy Lyman, Contra Costa County  
Public Works  
[gooma@california.com](mailto:gooma@california.com)

### Membership and Past President

Joel Rosingana, Independent Consultant  
[joelros@pacbell.net](mailto:joelros@pacbell.net)

### Webmaster

Vadim Barilko, Independent Consultant  
[vabarus@onebox.com](mailto:vabarus@onebox.com)

### Journal Editor

Lisa Loper, Database Specialists, Inc.  
[lloper@dbspecialists.com](mailto:lloper@dbspecialists.com)

### Vendor Relations

Mike DeVito, BankServ  
[mike@devito.com](mailto:mike@devito.com)

### IOUG-A Representative

Hamid Minoui, UPS  
[hamid.minoui@ups-scs.com](mailto:hamid.minoui@ups-scs.com)

### Members, At Large

Ganesh Sankar, Providian Financial  
[bgs2k2@yahoo.com](mailto:bgs2k2@yahoo.com)

Vilin Roufchaie, Cingular Wireless  
[vilin.roufchaie@cingular.com](mailto:vilin.roufchaie@cingular.com)

Eric Buskirk, Mission Critical 24/7  
[ebuskirk@mc247.com](mailto:ebuskirk@mc247.com)

### Conference Agenda Coordinator

Colette Lamm, Independent Consultant  
[colette\\_lamm@yahoo.com](mailto:colette_lamm@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 August issue is July 1, 2003. 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.*

(APPLICATION CONFIDENCE)



"Stride Rite's databases and DBAs run better with Quest Central."

Yusef Akyuz, CIO, Stride Rite Corporation

stride rite

**High Performance. High Availability.** With 45 discrete mission critical databases, Stride Rite cannot afford to lose ground due to poor performance or downtime. Quest Central™ allows a three-person team to manage more than a terabyte of data without running regular database management tasks that used to take days now take hours. And performance issues are identified and resolved before they affect users. Let Quest Software show you how you can confidently manage more databases without breaking stride or your staffing budget. For a free Database Health Check, call

800-306-9329 or visit [www.quest.com/nocouq](http://www.quest.com/nocouq).

FREE  
DATABASE  
HEALTH  
CHECK



© 2003 Quest Software Inc. Issue: 02/03/03 304492510000 For 916-2918898

QUEST CENTRAL™: MANAGE MORE DATABASES TODAY

## President's Message

(continued from page 1)

Also consider the magnitude of features in newer Oracle products. Who has time to try them all, let alone become an expert on each? This leads people to develop expertise in narrow areas, and it becomes challenging to find courses taught by instructors who have a deep under-

## Speaking of NoCOUG conferences, you won't want to miss the Spring Conference coming up Thursday, May 15 at Lockheed Martin in Sunnyvale.

standing of all the topics they are teaching. This reminds me of the instructor at Oracle University who insisted you could switch from rollback segments to automatic undo management without restarting the instance.

**Reading reference books is another popular way to keep your Oracle skills up to date.** But rapid change and feature complexity pose challenges here as well. The English language hasn't changed much in the last few years and the folks at Webster's have plenty of time to keep up with new words as they get introduced.

Thus, you don't question what you read in a dictionary. But if you consider that it takes at least a year from

the time somebody starts writing an Oracle reference book until you can buy it in the store, then it is possible that some material is already on its way to obsolescence.

And like instructors, writers can't possibly keep current on all of the newest product features and best practices. I remember reading in an Oracle 8 DBA guide a discussion of segment storage parameters. The

author recommended sizing the initial extent on all segments large enough to hold a year's worth of data. That practice was standard in Oracle 7, but was found to lead to fragmentation. Today we use the uniform extent model or auto-allocation instead.

**So am I suggesting you skip Oracle classes and reference books altogether?** Heck no! Just choose judiciously. Pick classes taught by technical people who are experts in the subject matter and keep their courseware updated. Read recently published books on focused topics by authors recognized in the industry.

Never take anything somebody tells you as gospel. In the classroom, ask a question if something doesn't seem quite right. When reading a book, jot down notes and research questionable statements.

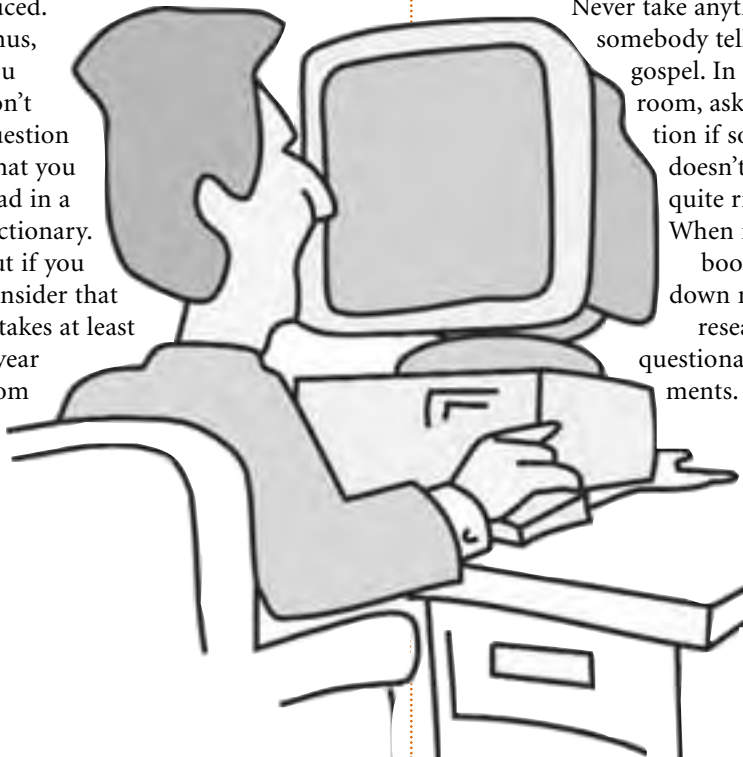
**There are also other sources for learning besides classes and books.**

Take NoCOUG for instance. At a NoCOUG conference you get to attend technical sessions prepared by a mix of renowned experts and typical Oracle users. The material is current and up to date. Room sizes are modest, so you can ask questions either during the session or later in the day. When you aren't attending a session, you get to exchange tips, techniques, and ideas with fellow Oracle users. The Roundtable Discussions provide a forum to formally ask your toughest questions to a large group of fellow Oracle users. It is quite possible that you'll find somebody who has already faced a challenge you are dealing with, and can point you in the right direction.

For \$40 you can attend one NoCOUG conference. For \$70 you can attend a full year of NoCOUG conferences. That's four solid days of Oracle education for about the price of a hefty reference book, or a fraction of a day of instructor-led training from Oracle University.

**Keeping your Oracle skills up to date can be challenging.** Finding courses or books that are truly current can be difficult, not to mention finding the time to study or read. NoCOUG conferences offer a great opportunity to learn from highly regarded experts and fellow Oracle users. The material is current and timely, and most presenters speak from personal experience. What better way to stretch your training dollar in these tough economic times?

Speaking of NoCOUG conferences, you won't want to miss the Spring Conference coming up Thursday, May 15 at Lockheed Martin in Sunnyvale. There will be twelve technical sessions for you to choose from, plus Roundtable Discussions, vendor exhibits, networking opportunities, and more. Thomas Kyte, the Tom behind the famous "Ask Tom" feature on Oracle's website, will be there. Check out the latest conference details at [www.nocoug.org](http://www.nocoug.org) and let us know you are coming. I hope to see you there! ▲





# An Interview with Colette Lamm

Colette is one of the newer members of the NoCOUG board, though she's been an active volunteer for over a year. As Conference Agenda Coordinator, Colette plays a vital role, especially in planning and organizing our quarterly NoCOUG Conferences. If you've attended our conferences in the past year or two, you've most likely seen her friendly face. If you haven't been introduced yet, here's your chance to find out a little more about Colette.

**What interested you in becoming a member, (or who twisted your arm to get you signed up)?**

Two years ago, I had the opportunity to go on a database-oriented GeekCruise. While on the cruise, I asked the experienced individuals which conferences and organizations they recommend and they led me in the direction of NoCOUG. I have been attending ever since due to the quality presentations and connections I have made. The knowledge and connections have been important to me, and I wanted to play a role in bringing that to others.

**Can you tell us a little more about your duties on the board?**

I am in charge of coordinating the presentations for the conferences. I work closely with the track leaders and other board members to attract speakers to NoCOUG, and I correspond with the speakers during the planning phases of each conference.

**What is your technology background?**

I have an M.S. in Database Development and Administration, B.S. in Information Systems, and a Minor in

Management. I have eight years of experience in IT, six of which were as a Microsoft Access Database Developer.

**How did you become interested in Oracle technology?**

Clients asked for guidance on which database technology to use. I wanted to provide them with concrete knowledge about which would be best in helping to perform their work.

**What type of work do you do (or what are you looking for)?**

With my Access background, I had the opportunity to do all phases of development as well as administration and maintenance. I am looking for an Oracle opportunity at this time as a team member either in development or administration. My favorite aspect of working with databases is the initial design process—working with the users by helping to define their needs and mapping out the system. One of my strengths is observing new environments and quickly picking up on the business rules and flow of information. I have worked in a variety of industries such as medical devices, telecom, manufacturing, and retail/e-commerce. I have helped a number of startups get their bearings, as well as departmental applications for larger organizations. I am a process-oriented individual and enjoy using databases as a tool to improve productivity, quality of life, and the bottom line.

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

I grew up in Walnut Creek and have always lived in the Bay Area, except for a short stint in the Seattle area working for Microsoft.



Colette Lamm

**What are your favorite hobbies?**

My favorite hobbies have to do with puzzles: jigsaw puzzles, logic puzzles, helping others organizing offices/ rooms, foreign languages, and most of all picture collages . . . funny how I would be organizing data for a living!!!!

**What was your most memorable vacation?**

I have had a few. As a child, two really stand out. One was a trip where we took sixteen airplane flights in three weeks checking out Bermuda, Orlando, Club Med, and various Caribbean islands. The second was an RV trip in tandem with another family to the National Parks of the eleven western states.

**What is your favorite electronic toy?**

I want to say my car, though I realize it is more mechanical. I have put more miles on my car this year than ever before. In addition to my involvement with NoCOUG, I also run a single's organization, and attend lots of networking events. If not in my car, there is one other place that one can guarantee you will find me . . . at my PC. Some think my fingers are permanently attached to my keyboard, I respond to email so quickly. I look forward to hearing from you in the future with your ideas for NoCOUG's future conference speakers and sessions. ▲

## Oracle Temporary Tables

(continued from page 1)

rary table spaces and the differences between, permanent and temporary, locally managed, and dictionary managed tablespaces designated as temporary.

Part II will be included in the next *NoCOUG Journal*.

### Overview

Two types of Oracle tables can exist in an Oracle database. These are the conventional permanent tables and the temporary tables that were introduced as of Oracle8i. There are also temporary segments that are used and managed internally by the Oracle engine as a result of certain operations such as those requiring sorts. Since a table is also referred to as a segment (or data segment), I refer to the segment created and managed by Oracle as “temporary segments” and use the term “temporary table” for an object of type table.

As the name suggests, both type of temporary segments have a short life span and are needed only during specific operations, during a session or even for the duration of a transaction. However, there are some major differences between the two.

### Temporary Tables

Unlike Oracle’s permanent table in which the data persists permanently until deleted (or truncated), data held in temporary tables persists only for the duration of a transaction or session depending on the option used when it

was created. In addition, the definition of a temporary table, created as GLOBAL TEMPORARY, is visible to all sessions, however, the data in the table are only visible to the session that inserted the data and not to any other sessions. Although, the data in a temporary table do not persist across sessions, their definition does. Currently only GLOBAL TEMPORARY tables are supported, which is a temporary table that can be shared by other sessions provided the appropriate privileges are granted.

The persistent level of the data inserted in a temporary table to be session-specific or transaction-specific is indicated by the option used in the creation of the table. The syntax below demonstrates these two options.

#### For session-specific:

```
CREATE GLOBAL TEMPORARY TABLE table_name  
(columns) ON COMMIT PRESERVE ROWS
```

#### For transaction-specific:

```
CREATE GLOBAL TEMPORARY TABLE table_name  
(columns) ON COMMIT DELETE ROWS
```

The default behavior deletes rows once the transaction is committed, otherwise the data in the table remain until the session is terminated, at which time the temporary segments associated with the session data are automatically cleaned up by Oracle.

### Temporary Segments Allocation

Even though the contents of temporary tables are just like those of a permanent table, they differ in the manner segments are allocated for them. First, a temporary table (and its indexes) has no segment allocated for it until the first row of

**LECCO**  
**SQL Expert**  
SQL with a Higher IQ

LECCO SQL Expert automates the task of performance tuning and optimization of database applications by providing developers and DBAs with expert knowledge of SQL. Through its proprietary Feedback Searching engine, it provides optimum SQL statements—guaranteed to provide the correct results. No guesswork or hopeful suggestions, statements are actually rewritten.

- Oracle
- Sybase
- MS SQL Server

Join the ranks of the experts and download a FREE evaluation copy today!

Where the experts go for performance.

**LECCOTECH**  
www.leccotech.com

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

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

\* The Five P's: 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

data are inserted. Note that this behavior is different in the creation of the permanent tables in which the initial extent of the segment is allocated as soon as the table is created. When segment allocation occurs, the temporary table is assigned a temporary segment associated with the session using the global temporary table. Such temporary segments are created in the temporary tablespace of the user to which the session belong, each session having its own private temporary segments. After the termination of the session or transaction, whichever the case might be, Oracle automatically drops the temporary segment associated with that session, and de-allocates the extents of the segment in the tablespace.

### Usage of Temporary Tables

Temporary tables have a number of different applications. They are used when a transaction or session requires a table for storing of intermediate data from which other data belonging to other permanent tables are derived or generated. Once the generated data are stored in the permanent table, the transient data are no longer needed. The duration of the data in the temporary table depends on the way the application uses it. The decisions for which type of table to use is driven by the need of the data manipulation process to the transient data, after each transaction is committed, or after the all transactions are completed. If the transient data are still needed beyond the life of the session, then a permanent table may be more appropriate.

### Operations on Temporary Tables

DML and DDL operations applicable to ordinary tables are also applied to temporary tables with some restrictions. TRUNCATE operations truncate only the current session's incarnation of the table. You cannot perform a DROP or ALTER operation to a temporary table while a session is bound to it. Deleting all the rows, using DELETE, followed by a COMMIT statement does not break the bound between the session and the table created with the ON COMMIT PRESERVE ROWS, only TRUNCATE does that.

### Indexes on Temporary Tables

Like ordinary tables, indexes can be created on temporary tables. However, the index must be created when the table is completely empty (i.e., when there are NO temporary segments for incarnations of the table).

If an attempt is made to create an index on a temporary table when the table is not empty, Oracle generates: **ORA-14452: attempt to create, alter or drop an index on temporary table already in use**

Similarly, an index created on a temporary table cannot be dropped or altered while a session is associated with it. Once all the sessions having data in the global temporary tables are terminated or the table is truncated, an associated index can be created (dropped or altered). Unlike ordinary indexes, an index on a temporary table is implemented as a separate temporary segment in the user's temporary tablespace.

Other objects such as views and triggers can also be created on these tables.

The index has the same session or transaction scope as the data in the temporary table.

### Redo logs

Perhaps one of the most advantageous uses of temporary tables is that redo entries are not being generated for the operations on the table. However, undo records are generated to facilitate transaction recovery, rollback to save points and read consistency. Hence, Oracle still needs to generate redo records for the changes made to the rollback segments. This can greatly reduce the number of redo files that would otherwise be generated for DML on permanent tables. Because of this, the overhead associated with frequent log switching is less compared to similar operations using a permanent table. This also speeds up the recovery after a media failure by not having to apply too many log records. Considering the fact that an INSERT requires a small amount of undo data and a DELETE requires a small amount of redo data, if the temporary table is mostly used for inserts with no delete, the relative saving in log generation can be more than 50% compared to log generation rates for a permanent table.

### Constraints

Constraints can be implemented on the table either at the session or transaction level, depending on the scope of the temporary table, and are not for the entire table even though the constraint is defined for the entire table. For example, a primary key or unique key constraint is applicable only at either session or transaction level. If the scope is the session, two users can enter the same values into the table from different sessions even though you have a primary/unique key in place for the entire table.

In the case of a transaction level temporary table, the same values can be entered from different transactions. Remember that all rows inserted by each transaction are deleted on commit.

You cannot specify any referential integrity (foreign key) constraints on temporary tables.

### Other characteristics

**Export:** Since the data on the table is only visible to the session associated with it, an export of the table only exports the definition of the table, which is visible to all sessions.

**Locks:** No DML lock is acquired on the data of the temporary tables and the LOCK statement has no effect on a temporary table because each session has its own exclusive data.

**CBO statistics:** Analyzing the table has no effect on temporary tables and the cost based optimizer has no statistics to help determine an execution plan.

### Restrictions

The following restrictions are imposed on temporary tables:

- They cannot be partitioned, index-organized, or clustered
- They cannot contain columns of nested table or varray type
- You cannot specify LOGGING or NOLOGGING, MONITORING or NOMONITORING or TABLESPACE, or storage clause
- Parallel DML and parallel queries are not supported
- You cannot specify the *segment\_attributes\_clause*, *nested\_table\_col\_properties*, or *parallel\_clause*

- Distributed transactions are not supported.
- You cannot rebuild an index on a temporary table

### Different Kinds of Temporary Segments

Temporary segments are used as temporary workspaces required by Oracle for intermediate stages of database processing. They are automatically created in the user's temporary tablespace by Oracle for a session requiring them as result of some operations during a session. There are different kinds of temporary segments in the database. These segments, used for sorting and other activities, have `SEGMENT_TYPE=TEMPORARY`. The tablespaces in which these segments are allocated may be of type `TEMPORARY` or `PERMANENT`. Furthermore, a `TEMPORARY` tablespace may be locally managed or dictionary managed. Depending on the type and extent management method of the tablespace in which temporary segments are created, different actions are performed in relation to the management and allocation and de-allocation of temporary created segments. Locally managed temporary tablespaces are the preferred tablespace for sort operations.

Temporary segments can explicitly be created by users for permanent segments creation. These temporary segments are created in the tablespace specified in the create statement or in the user's default tablespace.

### Temporary Large Objects (LOB)

Temporary LOBs are used as an interface to support the creation and deletion of LOBs that act like local variables. These types of temporary segments are not associated with any table and are only accessible by their creator. They are automatically deleted at session termination. A user can create a temporary BLOB, or CLOB, and their corresponding indexes by calling the `DBMS_LOB.CREATETEMPORARY` procedure.

### Temporary segments as Work Area for Sorting

Query processing often requires a temporary workspace for intermediate stages of SQL statement execution. This workspace, called a "sort area," is allocated in real memory with its *per-user-process* maximum size specified by the `SORT_AREA_SIZE` parameter. If this area is not sufficient for sorting, the entire sorted rows are written to disk to free up the sort area so that it can be reused for the remaining sort. For this purpose, Oracle automatically allocates a temporary segment in the user's `TEMPORARY TABLESPACE`.

See below for more information about sorting in memory.

Operations that may require temporary segments for sorting are:

- `CREATE INDEX` statement
- Queries that contain any one of: `ORDER BY`, `DISTINCT`, `GROUP BY`, `UNION`, `INTERSECT`, `MINUS`
- Statistic collection (e.g., `ANALYZE TABLE`)
- Un-indexed joins
- Correlated sub-queries

### Temporary segments for Permanent Segments Creation

Other SQL operations requiring temporary segments are:

- `CREATE PRIMARY/UNIQUE KEY CONSTRAINT`
- `ALTER TABLE ... ENABLE PRIMARY/UNIQUE KEY CONSTRAINT`
- `CREATE TABLE ... STORAGE` with `MINEXTENTS > 1`
- `CREATE TABLE AS SELECT (CTAS)`
- `CREATE PARTITION TABLE`
- `ALTER TABLE ... SPLIT PARTITION`
- `CREATE SNAPSHOT`
- `ALTER INDEX ... REBUILD`
- `DROP TABLE`

### Notes

1) The `CTAS` statement creates a data segment in the target tablespace, marks it as temporary in the dictionary, completes the operation and changes the segment type to `TABLE`. Other temporary segments may also be created if the `SELECT` involves sorting operations. One of the differences between these temporary segments is their target tablespace location; the former segment is allocated in the tablespace specified or the user's `DEFAULT TABLESPACE` whereas the latter is in the user's `TEMPORARY TABLESPACE`. For each parallel `CTAS` statement, each slave creates its own data segment feeding each from the row sources and marking each as `TEMPORARY` in the dictionary.

2) The `CREATE INDEX` statement, after sorting the index values, builds another temporary segment in the target tablespace and changes its segment type to `INDEX` and its name to the name of the index being created, once the index is built.

3) For `ALTER INDEX REBUILD`, Oracle also creates another temporary segment, in addition to the temporary segments required for sorting, and redefines it as the permanent index, dropping the original index once the rebuild is complete. The transient temporary segment is created in the same tablespace that the current index resides, or the tablespace specified with the `ALTER` statement.

4) For `DROP table`, Oracle converts the table into a temporary segment, and starts de-allocating the extents of the now temporary segments. If the drop is interrupted, the temporary segment is cleaned up by `SMON`. Furthermore, if the `SMON` process is interrupted by a shutdown abort, this may cause serious problems and the total time to clean up is increased.

### Sorting in Memory

The `init.ora` parameter `SORT_AREA_SIZE` specifies the maximum number of real memory bytes that can be allocated to each user process that performs sort in memory. `SORT_AREA_RETAINED_SIZE` specifies to what size the area should be reduced to, once sort is completed. This sort area size is allocated in the Program Global Area (PGA) for dedicated server configuration and in the User Global Area (UGA) for multi-threaded server configuration.

An automatic tuning mode referred to as automatic "SQL execution memory management feature" was introduced in Oracle9i. This feature, if enabled, takes the responsibility of tuning by setting the `SORT_AREA_SIZE` parameter away from the DBA. By way of this feature, the size of the PGA tunable portion allocated by an instance depends on the overall



PGA Memory target that is explicitly set by the DBA via the PGA\_AGGREGATE\_TARGET parameter. To enable this feature, the parameter WORKAREA\_SIZE\_POLICY must be set to AUTO. Only when enabled, a size value for PGA\_AGGREGATE\_TARGET is used, otherwise there is no effect.

Unless the server is configured with the shared server option, these parameters enable and set automatic sizing of SQL working area to the specified limit. As a result of this, Oracle dynamically alters the memory profile to assure best possible use of system memory and optimal system performance for memory consuming operations like sort. This leaves the determination of SQL execution memory

requirements to the database engine itself. This should improve the throughput for large number of concurrent users as well as queries response time.

The SORT\_AREA\_SIZE parameter is retained for backward compatibility.

### Space management for Temporary Segments

Oracle's management of temporary segments depends on the type (TEMPORARY or PERMANENT) and the extent management (LOCALLY or DICTIONARY) of the tablespace in which temporary segments are allocated. This is a subject that will be covered in Part II of this paper. ▲



## TECH TIPS

### LTMS Tablespaces: A Tech Tip from Dan Hotka

Oracle8i introduced a new method of managing space within a tablespace. Locally-managed tablespaces (LTMS) use a bit string stored in the beginning of each tablespace file that keeps track of free or assigned extents. Now that there is a new tablespace type, Oracle had to name it for the existing technology . . . now known as Dictionary Managed Tablespaces. LTMS also has a feature I have been pushing administrators to implement for a long time using the same extent size per tablespace. The following syntax is used to create the standard LTMS with a standard extent size of 1M.

```
CREATE TABLESPACE
<tablespace name>
DATAFILE '<file spec>'
SIZE 20M REUSE
EXTENT MANAGEMENT LOCAL;
```

Notice the “extent management local” syntax. Tables assigned to a LTMS defined tablespace will use the tablespace extent management and ignore any initial and next extent parameters, if defined. This LTMS syntax will allow you to create a tablespace with an extent size of your choice.

```
CREATE TEMPORARY
TABLESPACE <tablespace
name>
DATAFILE '<file spec>'
SIZE 20M REUSE
EXTENT MANAGEMENT LOCAL
UNIFORM SIZE 64K;
```

LTMS is great for those applications that are frequently allocating and de-allocating space, such as the temporary tablespace. I recommend using LTMS in an Oracle8i database for the temporary tablespace. Oracle v9.2 has autoextend set upon install. Make sure to change this if you do not want this feature in your tablespaces. Oracle9 v1 tablespaces are created as LTMS except for the SYSTEM tablespace. Oracle9 v2 has ALL of the tablespaces defined as LTMS, including the SYSTEM tablespace. When the SYSTEM tablespace is LTMS, you cannot define any dictionary-managed tablespaces for this instance.

#### What does LTMS do for you?

- No more “coalesce tablespace”
- No more wasted space due to uneven extent size
- Faster allocation/de-allocation for any tablespace, including the temporary tablespace

See my October 2002 Oracle Professional article entitled “Oracle Tablespace Options” for more detail on Oracle Tablespaces. The bottom line: you don't have a choice. LTMS is upon us.

**A full list of Dan Hotka's technical presentations are on [www.DanHotka.com](http://www.DanHotka.com). You can also sign up for his newsletter. ▲**



SPACE EXPERT™ GIVES YOU A SMARTER WAY TO  
*visualize* YOUR DATABASE'S SPACE PROBLEMS.

It's not always easy to see where pesky space problems are lurking, ready to degrade the performance of the database that your business depends on. And it's not always easy to keep your database operating at peak levels without lots of highly trained DBAs.

Until now, that is.

Introducing Space Expert™ for Oracle from BMC Software. It intelligently and automatically visualizes, isolates, analyzes and corrects space-related problems. Enabling even a novice DBA to perform the work of many. Which means the IT department's SLAs are good as gold. IT managers can concentrate on managing. Your database's performance is enhanced. And your company is more competitive than ever.

See for yourself. **Register for a 30-day trial copy of Space Expert for Oracle today at [www.bmc.com/distdata/spaceexpert](http://www.bmc.com/distdata/spaceexpert).** Then you'll discover just how easy we make it for you to help your database, and your business, really perform. Because it takes intelligence, not hocus pocus.





## SPONSORSHIP APPRECIATION

# 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

CHEVRON TEXACO

ORACLE CORP.

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

- BMC Software
- Database Specialists, Inc.
- Embarcadero Technologies
- 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*

January 1, 2003 \$ 52,787.79

### Revenue

Advertising	\$ -
Sponsorships	-
Membership Dues	14,280.00
Meeting Fees	520.00
Vendor Receipts	500.00
Interest	35.18
Vendor Mailing	-
<b>Total Revenue</b>	<b>\$ 22,335.18</b>

### Expenses

Regional Meeting	\$ 1,765.18
Journal	5,782.13
Membership	-
Administration	162.27
Website	75.00
Board Meeting	572.43
Miscellaneous	425.79
IRS	430.00
FTB Tax	405.00
<b>Total Expenses</b>	<b>\$ 9,192.01</b>

### *Ending Balance*

March 31, 2003 \$ 58,673.71

# Another Example of Interpreting Wait Events to Boost System Performance

Roger Schrag and Terry Sutton, Database Specialists, Inc.

This article is an excerpt from a 36-page white paper entitled "More Examples of Interpreting Wait Events to Boost System Performance." You may download the complete version of the white paper from [www.dbspecialists.com](http://www.dbspecialists.com). The complete white paper includes a detailed introduction to the wait event interface, wait event enhancements in Oracle 9i, and five examples of how wait event data was used to boost system performance.

## Example: More Buffer Busy Waits, Plus Latch Contention for Good Measure

In this article, we will look at an example where multiple Oracle processes try to write to different copies of the same data block in the buffer cache at the same time. Since Oracle has row-level locking, multiple users can update different rows in the same table at the same time without locking each other out. Multiple users can even update different rows in the same data block concurrently. But, as we will see in this example, contention and buffer busy waits can sometimes occur.

A genetic research company used an Oracle database to store research data. Periodically, they would load large volumes of raw data into the database. Then they would run a processing application on multiple workstations. Each workstation would fetch raw data from the database, process it, and put the processed data back in the database. The company had planned to add more workstations to speed up the processing, but found that adding more workstations actually seemed to reduce the throughput. The Oracle database was being used merely as a repository, we were told. All Oracle had to do was fetch the raw data and then store the processed data. The actual processing was being performed on the workstations outside of the realm of Oracle. So it would seem that the Oracle database should not be the bottleneck.

We found about 100 active database sessions, each initiated by a separate workstation. We modified the application code running on workstation 30 to activate debug event 10046 for wait event tracing. We did this by inserting the following statement into the code just after connecting to the database:

```
ALTER SESSION SET events '10046 trace name context forever, level 8';
```

We also modified the application to exit after iterating through its main loop 500 times. We ran the modified application on the workstation and retrieved the trace file that Oracle had created in the directory specified by the



Roger Schrag and Terry Sutton

`user_dump_dest` parameter. The application ran for less than one minute before exiting. Since this was an Oracle 9i database, we were able to run TKPROF to generate a report including wait event statistics with the following command:

```
tkprof prodgen_ora_16466.trc report_16466.prf waits=yes
```

An interesting excerpt from the TKPROF report was as follows:

```
UPDATE processing_stations
SET   status = 'ACTIVE',
      status_date = SYSDATE,
      data_set_id_being_processed = :b1
WHERE station_id = 30
```

call	count	cpu	elapsed	disk	query	current	rows
Parse	1	0.01	0.00	0	0	0	0
Execute	500	0.23	10.14	0	3616	1010	500
Fetch	0	0.00	0.00	0	0	0	0
total	501	0.24	10.14	0	3616	1010	500

```
Optimizer goal: CHOOSE
Parsing user id: 40
```

Rows	Row Source Operation
500	UPDATE
500	INDEX UNIQUE SCAN PROCESSING_STATIONS_PK (object id 9372)

```
Elapsed times include waiting on following events:
Event waited on          Times    Max. Wait      Total Waited
-----
buffer busy waits         26          0.71           7.87
latch free                17          0.57           2.08
log file switch completion  3          0.09           0.20
```

The UPDATE statement here seemed quite trivial—changing three column values on one row, given a primary key value. Oracle only needed to do about nine logical reads per execution, and no physical reads were required at all. Yet it still took Oracle over 10 seconds to execute the statement 500 times. If we had not specified the “waits=yes” TKPROF option, we might conclude that the database server was CPU-starved. But a quick look at the wait event portion of the TKPROF report showed that most of the 10 seconds of elapsed execution time was spent waiting on busy buffers and latches.

Some of the buffer busy waits listed in the trace file were as follows:

```
WAIT #2: nam='buffer busy waits' ela= 527727 p1=18 p2=10 p3=220
WAIT #2: nam='buffer busy waits' ela= 498765 p1=18 p2=10 p3=220
WAIT #2: nam='buffer busy waits' ela= 137611 p1=18 p2=10 p3=220
WAIT #2: nam='buffer busy waits' ela= 124165 p1=18 p2=10 p3=220
WAIT #2: nam='buffer busy waits' ela= 5237 p1=18 p2=10 p3=220
WAIT #2: nam='buffer busy waits' ela= 264050 p1=18 p2=10 p3=220
WAIT #2: nam='buffer busy waits' ela= 270177 p1=18 p2=10 p3=220
WAIT #2: nam='buffer busy waits' ela= 330912 p1=18 p2=10 p3=220
WAIT #2: nam='buffer busy waits' ela= 156317 p1=18 p2=10 p3=220
WAIT #2: nam='buffer busy waits' ela= 710696 p1=18 p2=10 p3=220
```

Since this trace file was generated by an Oracle 9i database, the elapsed times are shown in units of microseconds. The wait shown at the bottom of the list above had an elapsed time of 710,696 microseconds, or approximately 0.71 seconds. This was the longest buffer busy wait for this statement, as indicated by the “Max. Wait” figure in the TKPROF report.

From the previous example, recall that the parameters for the buffer busy waits event indicate file number, block number, and reason code. All of the buffer busy waits listed above involve block 10 in data file 18, with a reason code of 220. By querying the data dictionary, we found that block 10 in data file 18 was allocated to the PROCESSING\_STATIONS table, the table being updated by the statement that encountered the buffer busy waits. According to Metalink bulletin 34405.1, reason code 220 indicates that the buffer is busy because another session is using a copy of the buffer in an incompatible mode.

From the TKPROF report we can see that the application updated the PROCESSING\_STATIONS table 500 times during the period traced. Each update affected one row. While performing the updates, there were 26 times that the process had to wait on a buffer busy waits event. From looking at the trace file we know that all of the buffer busy waits occurred because data block 10 in data file 18 was already in the buffer cache, but in an incompatible mode. These 26 waits totaled 7.87 seconds.

A query against the data dictionary revealed the following statistics for the PROCESSING\_STATIONS table:

```
SQL> SELECT SYSDATE - last_analyzed, blocks, avg_row_len, avg_space,
2 chain_cnt, num_rows
3 FROM user_tables
4 WHERE table_name = 'PROCESSING_STATIONS';
```

SYSDATE-LAST_ANALYZED	BLOCKS	AVG_ROW_LEN	AVG_SPACE	CHAIN_CNT	NUM_ROWS
2.132118056	1	62	1686	0	100

```
SQL>
```

We can see that optimizer statistics were gathered recently and the table contains 100 very short rows, all stored in one data block. If the application session that we traced updated the PROCESSING\_STATIONS table 500 times in less than a minute, we might suspect that the database sessions from the other 99 workstations are doing roughly the same thing. This adds up to over 50,000 updates to the same data block per minute. Since each session is updating a different row in the table, there is no waiting on row-level locks (which would show up in the wait event interface as enqueue waits). However, only one process at a time may write to a buffer in the buffer cache. Therefore, a lot of time is being wasted contending for this one very hot buffer.

All of this begs the question, “Why is the application updating a row in the PROCESSING\_STATIONS table 500 times in less than one minute?” It turns out that the application uses this table to coordinate efforts between the workstations. There is one row in the table for each workstation. The application running on each workstation updates its row in PROCESSING\_STATIONS to indicate which data set it is processing so that other workstations will not attempt to process the same data set. Also, each workstation periodically updates the status date so that if a workstation crashes, it will be easy to tell when it crashed and what data it was processing during the crash.



## TECH TIPS

### Plotting Data with Orca



Orca is a free tool for plotting data on web pages. Orca reads almost any kind of data from text files and generates HTML pages and image files for viewing the data graphically in a web browser. Orca comes with a module that monitors operating system statistics. You can use Orca and the Orcallator module together to create graphical representations of useful system statistics such as CPU usage, TCP interface bits per second, I/O rate, memory usage, and disk space availability. Orca can be easily extended to plot any data—from weather statistics to database performance reports. Some DBAs use Orca to publish database performance statistics such as datafile I/O, user calls, parses, sorts, and transaction rates.

Developed by Dr. Blair Zajac, you can learn more about Orca and download it for free at <http://www.orcaware.com/orca>. ▲

Processing of one data set could take anywhere from a fraction of a second to twenty minutes. The application code was written to not only update the PROCESSING\_STATIONS table when a new data set was processed, but also to frequently update the timestamp during the processing of one data set. All stakeholders agreed that 500 updates per minute might be excessive. If the application code could be modified to update the table only once per second, then writes to the table would reduce by roughly 90% and contention for the hot buffer in the buffer cache would likely go away.

However, application changes needed to be tested thoroughly before deployment. A short-term fix was desired until the long-term fix of the application could be implemented. Luckily, a temporary fix was available that was simple to implement.

We rebuilt the PROCESSING\_STATIONS table with the PCTFREE storage parameter set to 99 and PCTUSED set to 1. This caused Oracle to leave 99% of the available space in each data block empty for future row expansion. This change caused the 100 rows in the table to be spread over 100 data blocks instead of all rows being neatly packed into one data block. A query against the data dictionary confirmed this:

```
SQL> SELECT SYSDATE - last_analyzed, blocks, avg_row_len, avg_space,
2         chain_cnt, num_rows
3 FROM   user_tables
4 WHERE table_name = 'PROCESSING_STATIONS';
```

```
SYSDATE-LAST_ANALYZED  BLOCKS  AVG_ROW_LEN  AVG_SPACE  CHAIN_CNT  NUM_ROWS
-----
.130868056            100           62          8014         0           100

SQL>
```

Tracing another 500 iterations of the main loop in the application on workstation 30 after rebuilding the PROCESSING\_STATIONS table yielded the following TKPROF output:

```
UPDATE processing_stations
SET   status = 'ACTIVE',
      status_date = SYSDATE,
      data_set_id_being_processed = :b1
WHERE station_id = 30

call      count          cpu          elapsed        disk        query        current    rows
-----
Parse     1          0.00         0.00           0            0            0           0
Execute  500         0.20         2.22           0           500          1009        500
Fetch     0          0.00         0.00           0            0            0           0
-----
total    501         0.20         2.22           0           500          1009        500

Optimizer goal: CHOOSE
Parsing user id: 40

Rows      Row Source Operation
-----
500      UPDATE
500      INDEX UNIQUE SCAN PROCESSING_STATIONS_PK (object id 9379)

Elapsed times include waiting on following events:
Event waited on          Times    Max. Wait      Total Waited
-----
latch free                2          0.35           0.61
```

The application is still updating the PROCESSING\_STATIONS table very rapidly. But now each row in the table is



## Need a Vacation?

If you manage the Oracle databases for a small or mid-sized company, chances are your vacations are few. We provide remote DBA services to free up your time and provide 24/7 support that's not possible with just one person. We can also handle all production support if needed. Count on us for:

- A cost-effective and flexible extension of your IT team
- Proactive database maintenance and quick resolution of problems
- Increased database performance and minimized database downtime
- Constant database monitoring with Database Rx™
- Onsite and offsite flexibility
- Reliable support from a stable team of DBAs familiar with your databases

**Call Us Today!**  
 (415) 344-0500 • (888) 648-0500  
[www.dbspecialists.com](http://www.dbspecialists.com)

ORACLE | CERTIFIED SOLUTION PARTNER



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

Hey, it could be you. You might be surprised at how much fun it can be with them. They actually work really really well. (Database?) For someone who's got through enterprise systems, storage, performance and security management challenges with innovative, game-changing tools. And Embarcadero's growing list of solutions are things that solve existing, but, lesser, known and specific to a wide range of development, management and integration scenarios for all kinds of IT environments. Completely transparent to how they handle to support. Email to [hr@embc.com](mailto:hr@embc.com) or call us at 415-344-0500. Or why not download a free, full-featured evaluation copy of one of our products and see a little for yourself. [www.embc.com/careers](http://www.embc.com/careers)

EMBARCADERO TECHNOLOGIES

stored in a separate data block, so each workstation is working with its own hot buffer in the buffer cache. The sessions are no longer competing with each other for the opportunity to write to the same buffer. Waits on the buffer busy waits event have been eliminated. As a side effect, logical reads have been reduced to one third of what they were originally.

The original TKPROF reports also indicated waits on the latch free wait event, indicating latch contention. A query against v\$event\_name shows that the parameters to the latch free wait event are as follows:

```
SQL> SELECT * FROM v$event_name WHERE name = 'latch free';
```

EVENT#	NAME	PARAMETER1	PARAMETER2	PARAMETER3
3	latch free	address	number	tries

```
SQL>
```

Some of the latch free wait events in the trace file generated before the PROCESSING\_STATIONS table was rebuilt were as follows:

```
WAIT #2: nam='latch free'  ela= 47004 p1=15113593728 p2=97 p3=0
WAIT #2: nam='latch free'  ela= 14629 p1=15113593728 p2=97 p3=1
WAIT #2: nam='latch free'  ela= 20652 p1=15113593728 p2=97 p3=2
WAIT #2: nam='latch free'  ela= 37737 p1=15113593728 p2=97 p3=3
```

A query against v\$latch or v\$latchname will give us the name of latch number 97:

```
SQL> SELECT name
       2 FROM   v$latch
       3 WHERE  latch# = 97;
```

NAME
cache buffers chains

```
SQL>
```

So we can see that the four wait events listed from the trace file above amount to a wait of approximately 0.12 seconds for the cache buffer chains latch. (Remember that in Oracle 9i trace files, elapsed times are listed in microseconds while in trace files from Oracle 8i and earlier, the elapsed times are in centiseconds.) This is not unexpected, since we had 100 sessions contending for the right to modify one buffer in the buffer cache.

Comparing the TKPROF report before rebuilding the PROCESSING\_STATIONS table with after, we see that rebuilding the table reduced the latch contention but did not eliminate it. It is likely that the long-term solution of modifying the application to reduce the update frequency will eliminate or further reduce the latch contention.

Comparing the TKPROF report before rebuilding the PROCESSING\_STATIONS table with after, we see that rebuilding the table reduced the latch contention but did not eliminate it. It is likely that the long-term solution of modifying the application to reduce the update frequency will eliminate or further reduce the latch contention. ▲

*Roger Schrag, OCP, has been an Oracle DBA and application architect for over twelve years, starting out at Oracle*

*Corporation on the Oracle Financials development team and moving on to production DBA and database architect positions at various companies in the San Francisco Bay Area. In 1995, Roger founded Database Specialists, Inc., a consulting firm specializing in business solutions based on Oracle technology, including on-site and remote database administration services. Roger is a frequent speaker at OracleWorld and the IOUG Live conferences. He is also president of the Northern California Oracle Users Group. You may contact Roger by email at rschrag@dbspecialists.com.*

*Terry Sutton, OCP, has been an Oracle DBA for nine years, and has worked in the information technology area for 16 years. Since 2000, Terry has been a senior staff consultant at Database Specialists, performing assignments ranging from production database administration to emergency troubleshooting with a particular focus on Oracle database performance tuning. You may contact Terry by email at tsutton@dbspecialists.com.*

*Note from the authors: We are always looking for interesting wait event situations to learn from. One can never be exposed to too many examples or techniques. If you are trying to tackle a problem using the wait event interface, feel free to email us the specifics of your situation. We'll do our best to look over what you send and give you our thoughts if you wish. But please understand, there aren't enough hours in the day for any one person (or two of us, actually) to solve the world's Oracle performance problems. We'll try to help you if we can.*



## TECH TIPS

### A Puzzle: When Are Nested Loops Free?

**U**nder certain conditions—especially in data warehouses—the cost of the nested-loop join is grossly underestimated. This leads to faulty selection of the nested-loop join method and severe performance degradation.



You'll find the full article at [www.oraclemagician.com](http://www.oraclemagician.com). Enjoy this Tech Tip from NoCOUG member and author, Chris Lawson. ▲



## TECH TIPS

### Show open cursors per user/process

```

rem
*****
rem * Name : OPENCURSOR.SQL
rem *
rem * Date: July 15 2001
rem * Purpose: Show open cursors per user/process
rem *
rem * Downloaded from www.oradev.com
rem
*****

select substr(ses.username||' '||ses.osuser,1,30) user_osuser
,
stat.value open_cursor
,
ses.process process
from v$sesstat stat
,
v$session ses
where stat.statistic# = 3
and stat.sid = ses.sid
;

```



## Thank You, Oracle!

A big “Thank you” goes out to Oracle Corporation which sponsored our February 2003 Winter Conference at the Hotel Sofitel. We’d especially like to thank Kate Kerner and Cassie Naval of Oracle Corporation who helped pull everything together for a very successful conference. Thank you!

—The NoCOUG Board of Directors



## Jonathan Lewis Event at ChevronTexaco, San Ramon

When this issue of the *NoCOUG Journal* went to press, the Jonathan Lewis class “**Optimizing Oracle—Performance by Design**” scheduled at ChevronTexaco in San Ramon on August 18-20 had a number of employees signed up.

Jonathan has agreed to accept external (non-ChevronTexaco) registrations after April 1. Seating is limited and there will be a waiting list. Class and sign-up information can be obtained at Jonathan’s website: <http://www.jlcomp.demon.co.uk/seminar.html>



*Jonathan will also be speaking at our NoCOUG meeting on August 21. Make sure you mark your calendar.*



# SQL Server 2000 and Oracle 9i

Peter Paton, Cognicase, Inc.

Ian Abramson, Author of Oracle9i Beginner's Guide

## Editor's Note

The following is an excerpt from an article titled, "Throwing Down the Gauntlet: Oracle9i vs. SQL Server" in which the authors attempted to provide an unbiased comparison of two products: Oracle9i and SQL Server 2000—both Enterprise Editions. There's not enough space to print the entire article, however, you'll find some interesting comparisons in the article covering a number of topics.

## Materialized Views

Although materialized views, also known as snapshots, have been a feature in Oracle for a number of years, SQL Server has only recently adopted them. Materialized views can be used to summarize and precompute data. For a large and complex view, dynamically rebuilding a result set each time it is requested by a query can incur substantial overhead. Materialized views will significantly reduce response time and ease the load on the server. Both Oracle and SQL Server provide an optimizer that automatically recognizes when a materialized view can be used to provide the desired result set from a query, even when the materialized view is not explicitly mentioned in the FROM clause. Both products also offer the ability to automatically reflect changes in the underlying tables in the materialized view.

## Partitioning

Although both vendors claim to support partitioning, Oracle offers a more comprehensive set of partitioning methods that provide a higher degree of transparency to applications or stored procedures. Partitioning improves query performance and can reduce the impact of administrative functions, such as index creation, by performing these functions at the partition level, rather than on the entire table.

Microsoft's solution to partitioning is to create multiple tables, then link these tables via a query. If multiple tables exist on physically separate servers, then the application that is retrieving data must first determine which server contains the majority of the data to be retrieved to optimize performance. It should then run the query on that server to minimize distributed processing. These data routing rules are not automated by the database engine; instead, they must be coded into the business services tier. Microsoft's partitioning solution is similar to the Oracle 7 Partitioned Views solution, which was abandoned due to performance problems.

Conversely, Oracle partitioning can be implemented without requiring any modification to the SELECT state-

ments or DML statements called by an application; creation of a separate view is not required. In addition, Oracle offers four partitioning methods: Range, List, Hash and Composite Partitioning. Range Partitioning allows data to be partitioned automatically by the database engine based on ranges of the partition key value. List Partitioning partitions data based on a specific list of values. Hash Partitioning is used when the distribution of data is not known, and Composite Partitioning is a combination of Range and Hash Partitioning.

Oracle also offers index partitioning, and partition pruning. Partition pruning automatically eliminates unnecessary partitions or subpartitions from SQL Statements if Oracle determines that the relevant data is not required. This frees server resources and speeds up performance. Oracle's partitioning solution is easier to implement and maintain than Microsoft's solution. In addition, Oracle provides a number of performance enhancing features that SQL Server lacks. Although SQL Server offers partitioning at a basic level, Oracle has delivered the more complete solution.

## Indexing

Effective indexing techniques are essential for maximizing database performance. Oracle and SQL Server both perform basic indexing functions in a similar manner. However, some of the more advanced functionality is found only in the Oracle product. SQL Server triggers are generally classified into two types:

### Clustered

- Determines the physical order of data in a table
- Effective for columns with a large number of distinct values and queries that return large result sets
- Only one clustered index may exist for a table

### Non-Clustered

- Appropriate for queries that do not return large result sets
- Multiple non-clustered indexes may be created in a given table

Function-based indexes are also supported.

It is important to note that when the fill factor for either type of index is modified, then a rebuild of the index is required. A rebuild may also be required to reduce defragmentation. However, rebuilding an index in SQL Server 2000 is quite efficient as the new (rebuilt) index is created from the old index rather than by scanning through the base table. An impressive feature is SQL Server's ability to

rebuild indexes online, minimizing downtime. Oracle's indexing solution is more comprehensive. Oracle provides the following five types of indexing, though some other variations of these methods also exist:

- B-tree indexes: the default, and most common type, of index
- Hash cluster indexes: physically store rows of a table in a hash cluster; reducing I/O for reads
- Reverse-key indexes: useful in cases of data distribution
- Bitmap indexes: used for columns with a small number of distinct values
- Function-based indexes: contain the precomputed value of a function or expression

Oracle also provides users with different options for defragmenting an index. Rebuilding an index is fast, but can require a significant amount of disk space. Coalescing an index, however, does not require more disk space. As with Microsoft, a rebuilt index may be created online and is created from the older version of the index, rather than by scanning the base table.

Clearly, Oracle has a wider range of options for users who wish to improve performance using advanced indexing techniques.

### Triggers

Although constraints should be used to enforce database integrity whenever possible, there are occasions when triggers are required to automatically enforce an organization's business rules. Oracle and Microsoft implement triggers somewhat differently, with Oracle offering extended functionality.

Oracle provides a number of trigger options including:

- ROW Triggers: executed once for every row that is affected by the triggering statement
- STATEMENT Triggers: executed once, regardless of the number of rows affected
- INSTEAD OF Triggers: executes instead of the triggering action; used primarily for views
- Event/System Triggers: can fire whenever one of the following types of operations occurs:
  - DML Statements on a schema object
  - DDL statements issued within a schema or database
  - User logon/logoff events
  - Server errors
  - Database startup
  - Instance shutdown

This range of triggers provides database developers and DBAs with a great deal of control over the database.

SQL Server 2000 supports STATEMENT triggers and INSTEAD OF triggers. ROW triggers do not exist, and execution of tasks based on system events is much more limited in SQL Server, though these may be handled through administrative utilities in SQL Server.

### Clustering

Microsoft and Oracle approach the issue of server clustering quite differently. Microsoft uses a federated data-

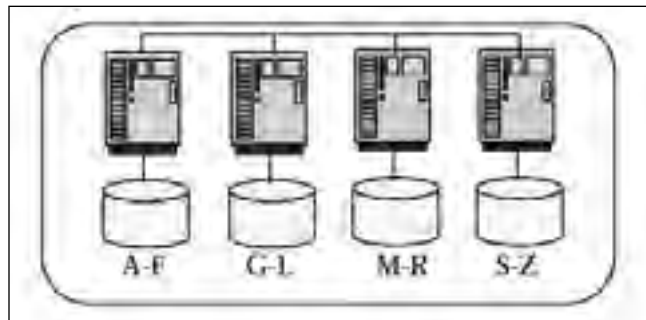


Figure 1 – SQL Server Federated Database

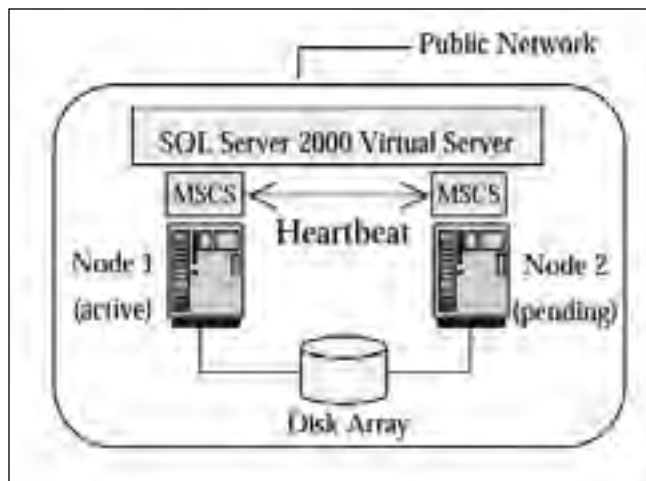


Figure 2 – SQL Server Failover Clustering

base approach to improve performance by adding additional computers. Using this approach, data is partitioned into separate databases that are controlled by different SQL Server instances. In the diagram below, data is partitioned into four separate databases accessed by four instances of SQL Server (Figure 1). Referential integrity and other database constraints cannot be implemented across instances of SQL server. Instead, triggers or data replication must be used.

Although the servers cooperate in managing the partitioned data, adding a new computer for scalability requires that data be repartitioned. Significant application code changes may also be required to reflect changed data routing rules. As an alternative, Microsoft offers failover clustering based on the Microsoft Clustering Service (MSCS) to protect against server failure. MSCS consists of multiple server computers that may access "common" resources such as a disk array (see Figure 2).

These multiple nodes appear as a single SQL Server 2000 machine. Failover clusters are not intended for load balancing, as only one node in the cluster may access SQL Server data at a given time. In the diagram, only Node 1 is actively responding to client requests for a particular database. Cluster nodes periodically send each other heartbeat messages. If a server fails to send a heartbeat message, then MSCS automatically transfers the application resources of that node to the other nodes in the network. Real Application Clustering (RAC) divides the workload evenly among all machines in the cluster. In doing this, Oracle

(continued on page 26)

# Identifying and Analyzing Key Factors Affecting System Performance

by Reyer Barel, Independent Consultant for Easiserve Inc.

## Abstract

Understanding how to identify and analyze system performance will greatly assist administrators in circumventing limiting performance factors. The tone and content of this presentation is designed to be streamlined for intermediate administrators who need an approach to determine factors which may be affecting overall system performance on their Midrange Unix Servers.

This presentation will explain various aspects on how to identify and analyze the load on system resources and what actions might be taken to alleviate this load. When specific syntax is provided it will be done so for the Solaris UNIX operating system.

## Introduction

Understanding how system resources are being used is a key step in maintaining control on performance. Often, incremental requirements will place too harsh a burden for system resources to handle. When a server becomes overloaded to the point where resources are insufficient to match the load, performance will suffer.

Administrators will benefit from understanding how to analyze aspects of system performance, as it will assist them in accurately identifying resources that are being depleted. In times of system performance crisis, the administration team will want to be efficient in detecting the root cause of the problem with minimal time-loss chasing non-contributing factors.

Early detection of performance degradation and proper knowledge of how system load is distributed are key aspects to effective proactive responses from administrators. With a good understanding of how system resources are allocated during normal operation, deviations in performance can be quickly detected by comparing resource usage to the healthy baseline statistics.

Ongoing regular system performance analysis can also greatly assist efforts for capacity planning for a given system. Understanding how a load is growing over time will provide the basis for rational hardware upgrade or replacement decisions. Early warning signs will allow for proper planning arrangements for upgrading an existing hardware platform.

## CPU Performance

### CPU Configuration Check

Identifying the number and speed of processors currently available on your server will be useful information to have. It is important to understand hardware limitations for CPU capacity for a given server platform. There are numerous models of Midrange UNIX Servers available for purchase from various Vendors. These models will have their own potential for CPU scalability.

Knowing your server model type will be sufficient information to determine its maximum CPU potential. Contacting the vendor directly, checking the server manuals, or searching the Vendor's website are typical ways of determining server hardware specifications for a given model type.

It is important to understand how your server is currently configured relative to its maximum processor configuration. This will allow you to calculate how much additional workload can be accommodated on the server with faster and additional processors before needing to consider a server replacement or server addition. Purchasing a more scalable hardware platform can be cost inhibitive so foreknowledge about a server's potential for upgrade is useful information. On Solaris platforms the "psrinfo" command can be used to tell you which CPUs are currently in use and when they were last enabled or disabled. Look at the following output run on a Solaris UNIX server. This data is for a four CPU server configuration with all CPUs active.

```
UNIXPROMPT> psrinfo
0 on-line since 12/14/01 11:18:56
1 on-line since 12/14/01 11:19:07
4 on-line since 12/14/01 11:19:07
5 on-line since 12/14/01 11:19:07
```

Specifying the psrinfo with verbose option and the CPU id number retrieved from the psrinfo command, will allow a determination of the processor's clock speed.

```
UNIXPROMPT> psrinfo -v 5
```

```
Status of processor 5 as of: 01/09/02 16:47:12
Processor has been on-line since 12/14/01 11:19:07.
The sparc processor operates at 464 MHz, and has a
sparc floating point processor.
```

## CPU Data Collection

By collecting CPU workload statistics on your servers you will be able to see the history of CPU usage. This is very useful for building trends and identifying load patterns during a day. Collected CPU statistics will allow you to relate user complaints of poor performance and other observations of performance back to the recorded load pattern of CPU usage.

After deciding to collect CPU data for workload analysis you need to consider when these activities should take place. A rational starting approach is to collect CPU data in broad daily intervals. For example, capturing performance metrics hourly throughout the day will provide a general sense of when your servers are experiencing peak loads. Later, when the broad daily load fluctuations have been collected, target specific times for data collection to increase the data resolution.

The goals of what you are trying to accomplish will set the tone for the frequency of data collection. If the goal is to establish an average workload for a given day, then regular and more broad time intervals should be used. If the goal is to break down the specifics of CPU usage during a particularly busy hour, then obviously you will sample the data collection more frequently.

The standard way to collect CPU workload data averaged across participating CPUs is to use the VMSTAT or SAR commands. These commands will provide you with various CPU metrics gathered from the UNIX kernel for a given time interval. They can be scheduled to run at preset times with the output being saved to specific files for later analysis.

It is important to note that whether you decide to use VMSTAT or SAR to gather the CPU statistics, be consistent with that method. There will be differences in how statistics are presented and calculated for these commands, so pick one and stick with it. For the purposes of discussion in this paper we will use the SAR command. To enable the SAR data collection method, a few administrative tasks need to be performed. You should have your system administrator set up SAR by running the `/usr/lib/sa/sa1` script and ensure that there is free space in the `/var/adm/sa` directory where the SAR files will be saved. The amount of space required will depend on the sampling times for the SAR collection. The easiest way to set up SAR to collect every hour would be the following entry in the crontab for root or sys.

```
0 * * * * /usr/lib/sa/sa1
```

Each day the SAR file will save collected data in a file specified by `/var/adm/sa/sadd`. The “dd” refers to the day of the month that the data was collected. Later we will explore some of the SAR commands used to retrieve this data for specific system resource usage.

### Run Queue Length/Run Queue Occupancy

The Run Queue Length is a term which refers to the number of processes which are runnable but are waiting for some CPU processing time. This is one of the key measurements in determining CPU deficiencies on the system. As the Run Queue Length grows, there are more processes waiting for CPU time. Thus, large Run Queues for pro-

longed periods of time indicate a deficiency of CPU power. Short durations of large Run Queue Length values are less significant than for prolonged periods of time.

Different systems will have different tolerances to performance slowdown. On one system poor performance for 10 minutes might have detrimental consequences and on another system an hour of poor performance might be OK. So take the tolerance factor into consideration when using guidelines for system performance analysis. As a general principle, pay attention to the Run Queue if it is sustaining values of more than several times the number of CPUs. So on a 4 CPU system, run queue lengths of more than 12 for “sustained periods of time” warrant extra attention.

This could be symptomatic of a CPU resource deficiency. Observe whether high values of the Run Queue Length correlate to perceived system slowdown as experienced by users, or the extra time that it is taking for certain jobs to finish.

The Run Queue Occupancy is the percentage of time that the Run Queue is occupied. Thus a low percentage for this parameter indicates a lot of system idle time on the server. A large value for both Run Queue and Run Occupancy indicates heavier loads. Using the “`sar -q`” command will yield the Run Queue Length and Run Queue Occupancy information. As we can see in the following output, CPU resources can’t keep up with the demand of processes during the 9am to 11am time frame on this 4 CPU system.

```
UNIXPROMPT> sar -q
00:00:02  runq-sz  %runocc
08:00:00  2.2      38
09:00:00  12.2     97
10:00:00  14.2     99
11:00:00  18.0     98
12:00:00  5.2      50
13:00:00  4.4      32
14:00:00  2.1      35
```

## CPU Utilization

CPU Utilization is the amount of time that the active CPUs on the system are working on running processes. CPU Utilization statistics presented by the “`sar -u`” command is displayed as a composite of the %System, %User and %Idle times, where the addition of all three parameters will equate to 100%. A lower %Idle time indicates a higher workload.

System and User statistics represent the proportion of time the CPUs are working on system-related activities or user-based programs, respectively. Characterization of a systems performance in terms of CPU Utilization is a widely used approach. In a normal workload, %System should not consume more than 20% of CPU utilization. CPU Utilization information is made more meaningful when combined with the Run Queue Length and Run Queue Occupancy statistics. For example, if you notice that your server is running close to the 95% utilization level for most of the day, does this indicate an immediate CPU deficiency on the server? The answer is: we don’t know yet. If this is a single CPU system and the Run Queue is consistently at a value of 1 then the answer is: the system is adequately provisioned with CPU power.

If the Run Queue Length consistently exceeds the number of CPUs on the system and the Run Queue Occupancy

is consistently high together with a high CPU utilization rate, then this would indicate a CPU deficiency.

An ideal situation would be for your server to consistently run under the 90% utilization level with a stable workload. In this situation, your investment in CPU power is justified since the utilization rate is high but not maxed out. The reality of most servers is that workload fluctuates and rarely is stable throughout an entire day. The measurement of CPU utilization can help to identify shortfalls in CPU processing power. This is especially true during peak periods where the demand of CPU resources can exceed those available and cause system-wide performance degradation. The `sar -u` command will generate the CPU utilization statistics. The following output shows that the system actually has some processes running but these statistics do not indicate CPU over-utilization.

```
UNIXPROMPT> sar -u
08:00:00 %usr %sys %wio %idle
08:00:00 30 10 2 58
09:00:00 28 9 2 62
10:00:00 15 5 1 78
11:00:00 19 5 1 75
12:00:00 14 4 1 81
13:00:00 18 6 1 74
```

Another useful command to get CPU load information is to run the “uptime” command. This command will provide the following information:

- The current time
- The number of days since the last reboot of the server
- The current number of users on the system
- The average number of jobs in the run queue averaged for the last 1, 5 and 15 minutes

If the 1- and 5-minute statistics show a heavier load than the 15 minute interval, this indicates an increasing load. If the 1- and 5-minute statistics show a smaller load value than the 15-minute interval, it means that the CPU load is being reduced. This command can be useful for spot-checking CPU performance and to check the current trend in usage. Here is an example of an increasing CPU load.

```
UNIXPROMPT> uptime
8:25pm up 26 day(s), 9:07, 10 users, load
average 14.1, 10.2, 7.5
```

### Using UNIX to Identify CPU Top Sessions

When you need to know how the workload is distributed amongst individual processes, you need to either run the UNIX “ps” command or enable system accounting at the server level. There are several ways to identify CPU top sessions on a system. One could use a wide variety of UNIX system tools, many of which are hardware specific, or use the basic UNIX ps command. The options used with the ps command will also be specific to the type of UNIX being run on your server. In general you will want to look at accumulated CPU time and %CPU time that the processes are consuming expressed as a percentage of total CPU resources.

Processes which are accumulating large amounts of CPU time and %CPU resources are your obvious targets. Analyze the regularity of CPU consumption for these processes. Are these processes consistently the top CPU sessions or just peri-

odic culprits? These questions can help identify processes which require tuning or rescheduling. In the following example, the `bbpkr` command, (which happens to be a backup process from Veritas Netbackup) has consumed 40 minutes of CPU time and 14% of the available CPU resources. This is quite an expensive process and the administrators will want to limit the times that this process runs so as not to affect the applications running during regular hours.

```
UNIXPROMPT> ps -ef -o ppid,time,%cpu,comm
| sort -r -k 3 | more

PGID    TIME    %CPU    COMMAND
576     40:17   14.0    bpbkar
15140   11:37   1.6     /export/svcbrkr/bin/sbetssl
620     29:38   0.9     /opt/JavaWebServer2.0/
jre/bin/native_threads/jre
17295   22:41   0.7     /export/svcbrkr/bin/sbnesubdb1
17295   38:28   0.7     /export/svcbrkr/bin/sbeaispp1
19910   0:16    0.6     /export/svcbrkr/bin/sbsorb1
```

The ps command run in this way allows you to observe the relative CPU consumption of UNIX processes. This command shows which processes are significant contributors to performance slowdowns that are caused by CPU resource shortages. In this syntax the “sort -r -k 3” is used to sort the ps command in terms of heaviest %CPU processes in descending order.

### Looking at CPU Utilization Within Oracle

On a database server, running the ps command can yield program names which are fairly non-descriptive. Usually you will only be able to determine the Oracle instance that the process is running against from the UNIX shell level. To relate UNIX processes with their Oracle instance sessions, determine the UNIX process id and then query inside the Oracle instance V\$SESSION table where the V\$SESSION.PROCESS column equals the UNIX process id. The Oracle SID and SERIAL# values can be selected from the V\$SESSION table and used for further querying inside the database to see what SQL statements are being run on behalf of the UNIX applications. This is useful when you want to determine what a heavy UNIX process is doing when it is running against an Oracle database. One could query inside the Oracle instance and sort for the top Oracle CPU consuming sessions independent of investigating activity at the UNIX level. In order to determine which users are using the most CPU resources you could run the following query inside an Oracle instance:

```
SQL> SELECT c.username, a.sid, a.value/600
"CPU(min)"
FROM v$sesstat a, v$statname b, v$session c
WHERE a.statistic# = b.statistic#
AND c.sid=a.sid
AND b.statistic#=12
AND a.value > 60000
ORDER BY a.value DESC;
```

USERNAME	SID	CPU(min)
SVCAPP	264	1700.8
THIC	270	102.0
JSMITH	530	63.8
JBOD_DAEMON	547	31.2
CMC	263	14.9

From the V\$STATNAME table, STATISTIC# equal to 12 represents the amount of accumulated CPU time in 10's of milliseconds. From this example we see that the SVCAPP userid is consuming an order of magnitude more accumulated CPU resources than the other users.

### Putting It All Together

Using the SAR command you can track CPU Utilization, the Run Queue and Run Occupancy statistics. This data will allow you to demonstrate trends in CPU usage during the course of a day. Independent analysis of which processes are contributing to the overall CPU load on a server can be accomplished by checking the results from a UNIX ps command. Applications which are running SQL statements can be identified inside the database and the details of what they are doing can be determined. Poor system performance should be rationalized in terms of the resources being depleted. After this is demonstrated, specific processes with an overly voracious appetite for those resources can be identified and investigated.

### Disk I/O

Disk i/o is a key factor in a system's ability to perform well. Due to the large difference in speed between an i/o request retrieved from physical disk versus memory, it is important to balance and minimize physical disk i/o. The ability of a disk to service i/o requests is another system resource that needs to be monitored and analyzed periodically to maintain overall health of the system.

### Disk Systems

Within the last decade or so, disk manufacturers have surfaced selling sophisticated disk frames with intelligent microcode and huge cache layers in an attempt to optimize disk i/o operations. In an effort to provide statistics to allow further optimization, these manufacturers have also grown successful by selling specialized software which allows detailed analysis of where disk i/o operations are occurring. The success of these companies demonstrates the importance of a disk to service its stored data efficiently.

### Disk Configurations

Disk devices which are accessible on your host server might be made up of one or more physical devices. The native UNIX analysis which you can run on your server to analyze disk i/o deals with the device level. Therefore, it is important to understand the configuration of the devices seen from the server so that good decisions will be made for where to store different data files. To that note, if RAID has been enabled, different devices may represent various levels of striping, mirroring or parity. This type of information is useful so that proper matching of RAID levels and the type of data is made to balance requirements for data protection and performance of data access.

### Analyzing Disk I/O

If your disk manufacturer has provided you with specialized software to analyze disk i/o, use it! Typically this software will allow you to track where the i/o is occurring and

how much is being serviced from physical disk and from the memory cache layer. The goal in analyzing disk i/o is to gain an understanding of how data is currently being retrieved from your disk storage so that you can minimize physical i/o and balance the requests across participating devices. The amount of data serviced from memory is a function of how much cache your disks have been configured with and the type of data retrieval algorithms being used in the disk microcode.

One data collection method to use on Solaris Servers is the "iostat" command. Another option is to use the sar -d command. The iostat command is more generic than the SAR for i/o reporting as it interactively reports statistics for disks, tapes, and terminals. Different options can be specified and various statistics are reported. Whichever data collection method is used, the kilobytes read per second (kr/s), kilobytes written per second (kw/s), and percentage of time that the disk is busy (%b) are useful statistics to track. A good approach is to schedule the iostat command for a period of time to sample disk i/o activity during a day and save the results to a file for later analysis.

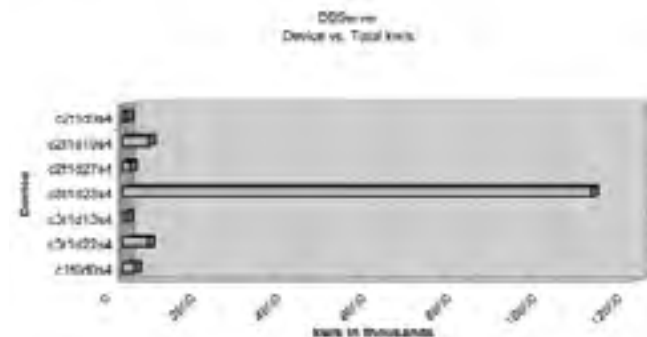
You can run the iostat with a time parameter to sample disk activity in set intervals. The data will be presented per device for each interval, so some scripting will be required to aggregate the data. The data collected can add up quickly, so be careful of where you are saving the data. In order to summarize the results over your sampling period some UNIX scripting will be required. Awk or perl is very useful for this task. Essentially, the exercise is to summarize the data and then compare the disk activity per device.

```
UNIXPROMPT> iostat -nx
r/s  w/s  kr/s  kw/s  wait  actv  wsvc_t  asvc_t  %w  %b  device
1.6  2.2  17.0  42.8  0.0  0.2    5.9    57.3  0  2  c1t0d0
0.0  0.0  0.0  0.0  0.0  0.0    0.0    0.0  0  0  c2t1d9
1.6  2.2  17.2  42.8  0.0  0.2    6.8    55.8  0  2  c2t1d19
0.0  0.0  0.0  0.0  0.0  0.0    0.0    2.1  0  0  c2t1d27
0.0  0.0  0.0  0.0  0.0  0.0    0.0    0.0  0  0  c3t1d13
0.0  0.0  0.0  0.0  0.0  0.0    0.0    2.1  0  0  c3t1d22
9.5  5.2  390.7  61.7  0.0  0.1    0.0    6.0  0  6  c3t1d23
```

Graphing the summarized results of kr/s, kw/s and %b per device will give a quick visual comparison of the use of each device. For reference, the kr/s and kw/s statistics do not distinguish whether the disk has serviced the i/o request from cache or not. As far as the server is concerned, the i/o request is now the responsibility of the disk device. This is where specialized disk software from the disk vendor can be useful in determining the hit ratio and whether upgrades in cache would be beneficial. Below is an example of graphed aggregated kw/s data for some active devices on a database server. It is quite evident from the graph that device c3t1d13 is far more active in writing data than any of the other devices. If possible, load balancing the write load across other devices will help improve overall disk i/o performance for this system. The data from your graphed iostat or sar -d command will show if system disk i/o load tends to be uniform or skewed across devices. You are looking for devices which are clearly more or less utilized than others.

Then the goal is to achieve approximate uniform data distribution. Heavily used devices are the ones you will

investigate further for an explanation of the load. Once you understand how a device is being accessed and why, you will be able to make a strategy for balancing the load. Obviously, under-utilized devices will be prime candidates for future data storage, whereas heavily used disks may need to have some data offloaded to other disks.



### Determining Oracle I/O Distribution

Since databases are applications with major data storage needs, it is beneficial to analyze the data distribution from different Oracle instances. The data which an Oracle database retrieves to satisfy a request from a user is tracked within tables. To access this data, one needs to run either the `utlstat/utlstat` scripts or to use `STATSPACK`. Rather than explain how to use these tools, let's look at the how to query directly against the Oracle tables which hold the pertinent information.

```
SQL> DROP VIEW stats$file_view;
SQL> CREATE VIEW stats$file_view AS
  SELECT ts.name ts,
         i.name name,
         x.phyrd$ pyr,
         x.phywrt$ pyw,
         x.readtim prt,
         x.writetim pwt,
         x.phyblkrd$ pbr,
         x.phyblkwr$ pbw,
         round(i.bytes/1000000) megabytes_size
  FROM   v$filestat x,
         ts$ ts,
         v$datafile i,
         file$ f
  WHERE  i.file# = f.file#
         AND ts.ts# = f.ts#
         AND x.file# = f.file#;
```

Creating the `STATS$FILE_VIEW` object is a convenient way to access the accumulated disk i/o statistics by tablespaces and filenames. The approach is to select from `STATS$FILE_VIEW` once at the beginning of a time interval, and put the results into a table like `STATS_FILE_BEGIN`. Do the same thing at the end of the time interval—put the results into a table name like `STATS_FILE_END`. The actual names of the tables do not matter. Run a query that subtracts the contents of the two tables `STATS_FILE_END` and `STATS_FILE_BEGIN` and you will end up with delta i/o statistics in your time interval across tablespaces and data files. This is the same approach that the `STATSPACK` and `BSTAT` utilities do except they capture much more than just these statistics.

### I/O Load Balancing

Summarizing the distribution of Oracle i/o allows you to see which tablespaces are performing the most disk

activity. Since a tablespace is a logical construct, it is more useful for i/o tuning to look at the amount of activity occurring at the datafile level. Working backwards from a datafile you can associate the mount point, and then the devices where this i/o is occurring. By adding new datafiles to mount points of the least utilized devices you can begin to load balance the disk i/o. If optimization of your current i/o distribution is needed, then you would work towards redistributing existing datafiles based on the results of your Oracle i/o statistics.

If data on a heavily used device can be relocated to a less utilized device, this should result in an overall increase in performance assuming the disks are equally matched in hardware specifications. You should be aware of software and hardware RAID levels before relocating data. For example, it is not advisable to move critical data from a RAID 0+1 group to a non-RAID device to balance performance as you would be sacrificing data protection. It is key to understand how devices have been configured prior to making assumptions about how to best balance load.

Your data requirements may have grown to the point where your defined devices are insufficient to satisfy the demands of data protection plus i/o optimization. In these cases it is always useful to have several standby disks which can be configured to meet new requirements. In addition, space on standby disks can be used as temporary storage for performing i/o migration in order to redefine existing devices.

### Network I/O

#### Network Elements

Networks constitute the physical infrastructure to allow data to pass from one server to the next. The actual number of network elements, i.e., switches, routers, hubs, etc., which are involved in designing a network can vary greatly depending on the particular needs and scope of a network's responsibility. Understanding the network elements that exist between your pertinent servers will improve your ability to troubleshoot network performance problems considerably.

Here are some of the typical questions you should ask when trying to understand how your network can affect online performance. Does a routing decision have to be made between your database server and application server or is it switched? Will backups affect online performance of your database server or is there a private network for the backup data? What is the speed of the interface on the router used by your servers? Is the interface on the router a shared vlan configuration or is it dedicated to only your group of servers? These are the types of questions which can assist in troubleshooting network performance problems.

Network elements should be monitored for potential problems on an ongoing basis. Switches and routers will track issues with packet transmissions by incrementing internal counters. These statistics should be reviewed periodically by the appropriate network personnel. Problems with communication between servers and network elements could be symptomatic of faulty cabling, duplex alignment, configuration or just failing interfaces on the network gear. All of these problems will negatively impact network performance.

It is unusual for a DBA to also have responsibility as a

network operator. However, some rudimentary understanding of the pertinent network components used to support your server(s) will allow for effective troubleshooting.

### Server Network Configuration

Each server must be configured to have an identity on a network. The identity will be the MAC address of your server Network Identification Card (NIC). The type of hardware chosen for the server to communicate on the network is also information that could prove useful when looking at network performance. Network cards can operate at different speeds.

The speed of your network card may be a limiting factor in how quickly your server can communicate to and from other sources on the network. The achieved data transfer rates in the past may not be achievable in the future without changing your server network configuration.

A way to increase your transfer rates for a given network workload on your server is to upgrade the network card to a faster model. For Ethernet technology, megabit and gigabit network cards are available for UNIX servers. Also, you can check the duplex factor on your network card and switch to make sure that if full duplexing is available it is utilized. Quite often auto-negotiation is enabled at the server and switch, which will handshake on a duplex setting. Full duplexing will allow synchronous transmit and receive signals to occur and can result in better network performance.

If you have enough i/o slots available on your server, you may want to configure more than one NIC or buy Multi-Network cards such as the Quad Fast Ethernet cards. Channeling software will allow you to configure Multi NIC to combine resources, which effectively adds throughput potential. For example, a Quad 100Mbit Ethernet card with the proper channeling software configured on the server and switch, will theoretically allow for 4x100Mbit transfer rates. The server and switch will load balance network traffic over all participating clustered interfaces. In this configuration your server will have one IP address but have multiple Ethernet interfaces actively transmitting and receiving packets.

In the situation where network performance is slow and analysis leads to suspect a particular network segment, check the NIC interfaces on the servers. Running a "netstat -i" command will show the interface statistics for those configured on your server. This command will show input/output errors, collisions, transmitted inbound and outbound packets. The percentage of IERRS and OERRS should be very low. If the NIC is consistently accumulating IERRS and/or OERRS the network card is likely faulty and causing undo network performance impact. In this following example the IERRS at 4000 is very suggestive of a faulty NIC. The Quad Fast Ethernet card port 0 (qfe0) interface is having the problems and will likely need to be replaced.

```
UNIXPROMPT> netstat -i
Name Address Ipkts Ierrs Opkts Oerrs Collis
lo0 localhost 366208401 41234 366208401 45466 0
qfe0 DB1 1183260016 0 1657329633 0 0
```

### Network Response Times

Most people working with UNIX have at some time or another come across the ping command. Ping uses a type of network protocol called ICMP. The ICMP packets are treated by default as low priority meaning that regular application traffic will be of higher priority for the network to deal with. Keeping this in mind when analyzing ping response times can help with interpreting the results.

Randomly running the ping commands on various servers during network performance sluggishness is not particularly useful. However, knowing the network layout and network elements in place between your various servers, together with using a targeted ping command, can be very productive in determining the problem. If you are experiencing network performance problems, isolating where the slowness is occurring should be your goal. Is the problem occurring between servers communicating through a router, or servers on the same subnet communicating via a switch? If there are multiple network elements involved, each one should be tested individually for performance degradation. This is where a ping command comes in handy.

```
UNIXPROMPT> ping -s 24.141.30.249
64 bytes from DB1 (24.141.50.249): icmp_seq=0. time=99. ms
64 bytes from DB1 (24.141.50.249): icmp_seq=1. time=67. ms
64 bytes from DB1 (24.141.50.249): icmp_seq=2. time=75. ms
64 bytes from DB1 (24.141.50.249): icmp_seq=3. time=80. ms
64 bytes from DB1 (24.141.50.249): icmp_seq=4. time=45. ms
64 bytes from DB1 (24.141.50.249): icmp_seq=5. time=60. ms
```

This ping command shows the results of pinging a remote server with IP address 24.141.30.249. The results seem to be slow with access times between two servers on an intranet. The reasons the ping results are slow is that pre-analysis of response times from the same servers during regular performance is between 0 and 10ms. You should know what the usual values are before trying to make sense of ping response times during periods of network slowdowns. While the ping command comes in handy for troubleshooting, it can also be used as a means of sampling network performance during different times of the day. The data can be collected to a file and later graphed. It is advisable first to check with the network team to see if they already have software being used to determine network loads before adding even more load with the ping UNIX command.

### Reducing Network Load

When the performance of the network has diminished due to increased volume of traffic, it is time to consider increasing the network bandwidth or decreasing the traffic. While this decision typically lies with the network architects of the company, due diligence on the part of administrators can assist in extending network longevity. Traffic flow analysis can be scrutinized on a periodic basis to determine the traffic cross-section. Network performance tools can be employed to determine the type and amount of traffic across different network points. This data can then be analyzed to rationalize the current traffic flow. It might become evident after analyzing traffic flow reports that existing network infrastructure is adequate, but the usage of the network needs to be rescheduled or re-evaluated.



## Balancing Network Load

Administrators can help by balancing the network workload to the best of their knowledge. Jobs bearing considerable network load should be scheduled to non-peak times if possible. Significant network loads that can be spread out during the course of the day would be helpful in avoiding performance slowdowns. With multiple Ethernet interfaces available on a server, you may decide to have them configured with separate IP addresses and try to balance the network load across available interfaces. One such approach is to configure backup traffic to occur on a private segment from application data. This example is chosen specifically because database servers can store huge amounts of data, often translating into applications impacting network traffic during backups.

One of the huge advantages in separating out backup traffic and application traffic is that DBAs can now schedule backup jobs to tape or other media with impunity. If the backup network is truly private then running a backup job should not impact the application traffic. The flexibility in a 24-hour backup window is truly a wonderful thing. Failed backups or modified backup configurations are now possible during regular business hours due to the separate segment carrying the traffic.

## Network Monitoring

The role of monitoring a network is best left for a network operations team as there are several strategic decisions to make on how best to accomplish this task. Some of those decisions involve which network software tools to use, how to collect, analyze, interpret, and present the data. Automation of network monitoring is a very reasonable approach that many companies have adopted in efforts to keep efficient in the task.

Server-based network monitoring is often not covered off by the network operations team and server-specific problems can be left to the administrators to deal with. Software is available which is designed to probe throughput of different data types and peak loads configurable for different times of the day. As a quick example, FTP traffic and SQL traffic can be simulated and the response times measured by the software between different server endpoints. The data is captured automatically at a very low overhead to the server's network bandwidth and history is stored. The results can be published daily to a web page and available to various users. This type of software automation provides a very strong basis for determining when peak loads are occurring and when network performance is suffering. The administrators can use this data to plan out their servers load on the network.

## Network Troubleshooting Example

Here is a network troubleshooting example which can be solved based on the above discussion. Your company's main application performance is suffering at different times during the day without definite regularity. All eyes are looking at the database as being the culprit since application logs show that more time is being spent on the database SQL statements than normal. After analysis inside the database you

conclude that there is nothing fundamentally wrong there. You begin to become suspicious of the network. You know that the application server is running on a different subnet from your database server and there is a router that gets involved everytime the two servers need to exchange data.

Testing the connection to the database local to the database segment shows a very quick response. Connecting to the database server from the segment local to the application server shows a significantly slower response. You are on the right track! You begin to test ping responses specifically between the application server and the database server and notice that the values are consistently above 50ms, whereas during normal performance they are close to 1ms responses.

Having established that there is a definite slower response between the application server and the database server, you want to determine if the common part of the problem is the router. You test network response times between the database server and another server on the same application subnet as your application server. The results are also between 50-100ms response.

The problem only lasts for 20 minutes and then performance returns to normal. Your tests suggest that the common router interface between the application subnet and the database subnet is being periodically overloaded. A check back on the database server shows that tape archiving was the only scheduled job that occurred during the 20 minutes of network slowdown. These facts suggest that the backup stream places too much load across the router interface in addition to normal traffic. This network slowdown is affecting the speed of your application because it depends on SQLNET traffic to get results from the database server. Temporarily you decide to schedule the backups during the night because application traffic is minimal at this time. You will discuss other options with the network team, including the router interface specifications and the possibility of separating out backup traffic from application traffic.

## Concluding Statements

System resources need to be monitored on an ongoing basis to understand how they might be affecting system performance. On UNIX systems there are different ways to accomplish this task. It has been shown how to accomplish system performance monitoring using native UNIX commands. It has also been shown how to query Oracle databases for its load on system resource usage.

The collection and analysis of system resource statistics over time, will build a good understanding for how system resource usage characterizes performance. While the approach of collecting system metrics is similar, the interpretation of those metrics will vary depending on the server being analyzed. Differences in server configurations, hardware specifications and application load, means that interpretation of system performance statistics needs to be tailored to each server individually. ▲

*This article is excerpted and reprinted with permission from the IOUG Live! Conference, April 2002.*

## SQL Server 2000 and Oracle 9i

(continued from page 18)

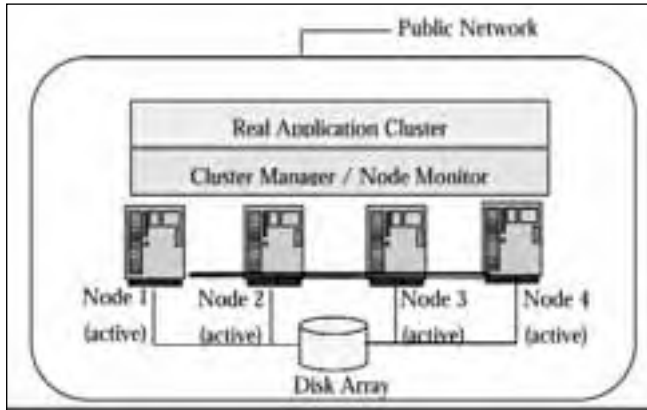


Figure 3 – Oracle Real Application Cluster

has provided a single solution to address both scalability and availability. No data partitioning or additional coding is required (see Figure 3). A single database may be run on a group of servers clustered together, with caching occurring across the cluster.

Failure of a node is detected by Cluster Manager software. This software is usually provided by the O/S or hardware vendor, although Oracle provides its own software for the Windows platform. If one machine fails in RAC, the cluster manager software detects the failure through the node monitor component, and the workload is again evenly distributed among the other servers. The number of nodes in a cluster is highly scalable to over 40, though there is little evidence of clustering being implemented anywhere near this scale. Oracle's clustering solution is significantly easier to administer and more adaptable than Microsoft's solution. Although careful planning in a Microsoft clustered environment will provide performance improvements, modifying the cluster configuration and creating application code can prove difficult.

### References

#### Oracle

- MetaLink (metalink.oracle.com)
- Oracle TechNet (technet.oracle.com)
- Oracle.com
- Oracle 9i – A Beginner's Guide. Michael Abbey, Mike Corey and Ian Abramson, Oracle Press 2002
- Oracle 9i Administrator's Guide (Release 1, 2001)
- Oracle 9i Database Concepts (Release 1, 2001)
- orafaq.com

#### Microsoft

- Microsoft SQL Server Books Online (2000)
- Microsoft TechNet
- Microsoft.com
- MSDN Library
- Microsoft pre-sales technical support ▲

This article is excerpted and reprinted with permission from the IOUG Live! Conference, April 2002.

## NoCOUG Spring Conference

Thursday, May 15, 2003  
Session Descriptions

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

### DBA TRACK

#### Implementing Oracle9i Data Guard for Higher Availability

Daniel Liu, Senior Oracle DBA, First American Real Estate Solutions

This presentation provides an overview of Oracle9i Data Guard technology. The session offers an introduction to the basic concepts and architectures of Data Guard. It discusses the following implementation steps: planning for high availability, creating the standby database environment, setting up the log transport services, managing the log apply services, and administrating the Data Guard environment by using Oracle9i Data Guard Manager and command-line interface. This session covers new features in Oracle9i Data Guard Release 1 and 2. This session also compares Data Guard technology with other disaster recovery methods.

#### Moving to LDAP: A Step-by-Step Setup of Directory Naming Using OID

Daniel Liu, Senior Oracle DBA, First American Real Estate Solutions

Oracle Internet Directory (OID) provides a new mechanism of resolving net service names by storing database-connecting information in a LDAP (Lightweight Directory Access Protocol) Server. LDAP-based name resolution is considered an industrial standard as compared to the more proprietary Oracle Names Server. The OID also replaces the need for storing connecting information on each individual client machine. This presentation discusses the required steps to install, configure, and set up an Oracle Internet Directory Server for name resolution. It offers an introduction to the basic concepts of OID and Oracle Net. It shows how to use Oracle Directory Manager, Net Assistant, and "oidctl" utilities to configure ldap.ora, names.ora, and sqlnet.ora.

#### Oracle Enterprise Manager (EM V4)

Dave LeRoy and Martin Pena, Product Directors, Oracle Corporation

Enterprise Manager (EM) is Oracle's single, integrated solution for administering and monitoring applications and systems that are based on the Oracle technology stack. EM V4 is the latest generation of the Enterprise Manager that includes significant enhancements in the areas of holistic system monitoring, centralized application server management, web application performance management, extensive host monitoring, and enterprise configuration management.

## **Implications of Setting Oracle9i's Statistics Collection Level**

James Koopmann, *Co-Founder and Chief Architect, dbDoctor*

Statistical analysis is at the root of all performance efforts. Oracle has introduced a new method of turning on statistical collection and expanding our ability to look deeper into areas not available before. This presentation will take a look at the implications of setting up this new method of statistical collection and its impact on the system.

## **APPLICATION DEVELOPER TRACK**

### **How to Figure Out What's Wrong**

Thomas Kyte, *Vice President, Core Technologies, Oracle Corporation*

Thomas Kyte frequently gets asked, "My program is slow, what parameter do I need to set to make it go faster"? Many people are in search of the fast=true init.ora parameter—there is no such parameter. This presentation will describe how the author sets about diagnosing application performance-related issues, the tools used, and the techniques you should employ during development to avoid having to triage your system after the fact.

### **National Language Support**

Brian Hitchcock, *DBA, Sun Microsystems*

Of all the DBA joints in all the corporate world, she walked into mine. Christine said little. "Brian, the Kanji data are missing." I knew I was in trouble. Up to that point, no one had mentioned the minor detail about the multi-byte Japanese Kanji in this innocent looking single-byte database. The "expert" DBAs at corporate analyzed the situation in the predictable way. "Brian, a single-byte database can't support multi-byte Kanji." And that was that. I was stuck, in the middle of nowhere, between Christine's need for her Kanji, and the experts' declaration that the Kanji had never existed. This left only one small detail. How to get the Kanji back?

## **GRAB BAG TRACK**

### **Managing What Matters Most—A Holistic Approach to Monitoring Oracle Applications**

Steve Coleman, *Senior Engineer, Candle Corporation, Consulting & Services*

Problem isolation and resolution are hard enough in the simplistic client/server environments. We need to take a holistic approach to managing the Oracle applications infrastructure by bringing in event, performance, and availability data from clients, networks, web servers, middle tiers, and databases and anything else that can be considered part of the infrastructure, and present it a clear, concise fashion so that critical business applications based on Oracle remain up and responsive.

### **Managing OLAP Catalog Metadata in Oracle9i R2 Database**

Shinji Matsumoto, *Vice President of Marketing, IAF Software, Inc.*

The OLAP option of the Oracle9i Release 2 Database uses metadata in the database's OLAP catalog to describe multi-dimensional models and mappings to data sources. This presentation discusses the OLAP option's multi-dimensional data model, data sources (relational and multi-dimensional data types), run-time uses of the OLAP catalog (e.g., the OLAP API and administrative processes), the OLAP Catalog API and BI Beans applications that can be used to manage the OLAP Catalog metadata.

### **Materialized Views**

Willie Albino, *Senior DBA, Lockheed Martin*

This entry-level presentation on Materialized Views (Oracle8i version), will cover what materialized views are, how they are created and used, and what setup steps must be performed to be used effectively. In addition, to relevant examples, some of the "behind-the-curtain" mechanics relating to materialized views will also be covered.

### **Achieving Your Project's ROI: Managing the Human Aspects of Project Implementation**

Dr. Frances Bruce, *Independent Consultant, Bruce Partners Consulting*

Organizations today face even more challenges than before, particularly on how to obtain collaboration, how to achieve ROI, and how to get projects implemented faster with full take-up of what the project delivers. This session provides the latest figures on project "wastage," based on current research into Fortune 500 companies. Analysis of the data indicates the root causes of implementation problems, backed up by case studies of how the symptoms of these causes played out in several organizations. A solution is outlined which addresses these causes through implementing projects as value delivery systems. ▲

**Mark Your Calendars!**

**Our Summer Conference  
will take place on  
Thursday, August 21, 2003 at  
ChevronTexaco in San Ramon.  
You won't want to miss it!**

# NoCOUG Spring Conference

Thursday, May 15, 2003

**Location:** The NoCOUG Spring Conference will be held on Thursday, May 15, 2003 at Lockheed Martin. Building 157 is located at the intersection of Third Avenue and J Street in Sunnyvale. See complete directions to the meeting location at [www.nocoug.org](http://www.nocoug.org). *Note: U.S. citizens must bring government issued identification. Green card holders bring green card. Foreign nationals special pre-registration required. See website for details.*

- 8:00 Registration and welcome. Refreshments served.
- 9:00–9:30 General Session.
- 9:30–9:45 Break
- 9:45–10:45 Parallel Sessions #1  
DBA Track: Implementing Oracle9i Data Guard for Higher Availability  
Application Developer: How to Figure Out What's Wrong, Part 1  
Grab Bag: Managing What Matters Most—A Holistic Approach To Monitoring Oracle Applications
- 10:45–11:30 Break
- 11:30–12:30 Parallel Sessions #2  
DBA Track: Moving to LDAP: A Step-by-Step Setup of Directory Naming Using OID  
Application Developer: How to Figure Out What's Wrong, Part 2  
Grab Bag: Managing OLAP Catalog Metadata in Oracle9i R2 Database
- 12:30–1:30 Lunch Break
- 1:30–1:55 Roundtable Discussions for DBA and Developer Tracks
- 2:00–3:00 Parallel Sessions #3  
DBA Track: Oracle Enterprise Manager (EM V4)  
Application Developer: National Language Support  
Grab Bag: Materialized Views
- 3:00–3:30 Afternoon Break and Raffle
- 3:30–4:30 Parallel Sessions #4  
DBA Track: Implications of Setting Oracle9i's Statistics Collection Level  
Application Developer: TBD  
Grab Bag: Achieving Your Project's ROI: Managing the Human Aspects of Project Implementation
- 4:45–?? NoCOUG Networking and Happy Hour at Faz Restaurant and Bar,  
1108 N. Mathilda Avenue in Sunnyvale
- Cost: \$40 admission fee for non-members. Members free. Includes lunch voucher.

View session  
descriptions on  
page 26.

Keep  
checking the  
NoCOUG website  
for the most  
updated Spring  
Conference  
schedule!

**RSVP online at <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

Thank you to Lockheed Martin,  
our meeting sponsor.