

Official Publication of the Northern California Oracle Users Group

NoCOUG

J O U R N A L

Vol. 21, No. 4 • NOVEMBER 2007

\$15

Set Your Sights Higher at NoCOUG

Asking Jeremiah

An interview with our keynote speaker, Jeremiah Wilton.

See page 4.

The Future Is Now!

CA Database Command Center to the rescue.

See page 14.

A Matter of Life or Death!

An unusual perspective on performance tuning. See page 20.

Much more inside . . .

Your Guide to the Changing World of Oracle

Until not so long ago, we got to work at 8 a.m. and left work at 5 p.m. We had no access to the databases from home, all the members of the team worked in one location, most companies only had a few databases, most companies used only one database technology, and the business did not need the databases on nights and weekends.

Those days are gone. The number of databases has increased tremendously. Now, SQL Server databases coexist side by side with Oracle databases; the business needs the databases around the clock, during the night, and on weekends and holidays; database administrators take cell phones and wireless-equipped laptops with them when they go fishing; and distributed teams, outsourcing, and offshoring are the order of the day.

Let the *NoCOUG Journal* be one of your guides to the changing world of Oracle. The highlight of this issue is a thought-provoking interview with Jeremiah Wilton, who oversaw the growth of Amazon.com's technical infrastructure from three databases to 500; through his work there, he became a recognized thought leader in the areas of scalability, high availability, performance, recovery, and disaster planning.

I hope you enjoy this issue and I hope to see you at our fall conference on October 25, where Jeremiah will deliver the keynote address.

—Iggly Fernandez, *NoCOUG Journal* Editor

Table of Contents

President's Message	3
Interview: Asking Jeremiah.....	4
Book Review: Oracle Dataguard	10
Tools of the Trade: The Future Is Now!.....	14
SQL Corner: Oracle Annoyances for Geeks	16
Performance Corner: A Matter of Life or Death!	20
Bag o' Tricks	22
Fall Conference Abstracts	24
Sponsor Appreciation	25
Fall Conference Schedule.....	28

ADVERTISERS

Roundstone Systems.....	7
BEZ Systems	13
Database Specialists, Inc.	13
Quest Software	19
IT Convergence.....	19
Embarcadero Technologies.....	21
Network Appliance.....	21
Princeton Softech	23
Confio Software.....	27

Publication Notices and Submission Format

The *NoCOUG Journal* is published four times a year by the Northern California Oracle Users Group (NoCOUG) approximately two weeks prior to the quarterly educational conferences.

Please send your questions, feedback, and submissions to the *NoCOUG Journal* editor at journal@nocoug.org.

The submission deadline for the upcoming February 2008 issue is November 30, 2007. Article submissions should be made in Microsoft Word format via email.

Copyright © 2007 by the Northern California Oracle Users Group except where otherwise indicated.

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

NoCOUG BOARD

President

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

Vice President

Darrin Swan, Quest Software
darrin.swan@quest.com

Secretary/Treasurer

Jen Hong, Stanford University
hong_jen@yahoo.com

Director of Membership

Joel Rosingana, Independent Consultant
joelros@pacbell.net

Journal Editor

Iggly Fernandez, Verizon Business
iggly_fernandez@hotmail.com

Webmaster

Eric Hutchinson, Independent Consultant
erichutchinson@comcast.net

Vendor Coordinator/IOUG Rep

Diane Lee, Lockheed Martin
diane.c.lee@lmco.com

Director of Conference Programming

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

Director of Marketing

Naren Nagtode, Franklin Templeton
nagtode@yahoo.com

Training Day Coordinator

Hamid Minoui, Optimal Strategies
hminoui@pacbell.net

Track Leader

Randy Samberg, Access Systems Americas, Inc.
rsamberg@sbcglobal.net

Track Leader

Hanan Hit, Skyrider, Inc.
hanan@skyrider.com

NoCOUG Staff

Nora Rosingana

Technical Reviewers

Ted Syrett, Danny Chow, Sumit Aneja

Book Reviewer

Brian Hitchcock

ADVERTISING RATES

The *NoCOUG Journal* is published quarterly.

Size	Per Issue	Per Year
Quarter Page	\$125	\$400
Half Page	\$250	\$800
Full Page	\$500	\$1,600
Inside Cover	\$750	\$2,400

Personnel recruitment ads are not accepted.

journal@nocoug.org

Teamwork

by Lisa Loper

“The strength of the team is each individual member . . . the strength of each member is the team.”

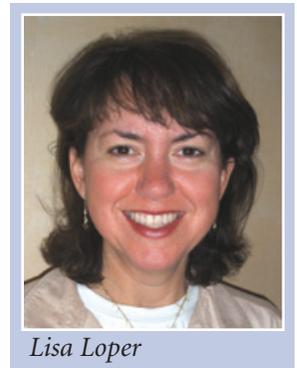
—Coach Phil Jackson, Chicago Bulls

Given that this is the fourth quarterly issue of the NoCOUG Journal for 2007, and therefore my last president’s message of the year, I had to find a way to bring in one last reference to my enjoyment of sports. What better way than to have a Phil Jackson quote regarding teamwork. But even if you’re not involved with sports, teamwork is surely part of your everyday life—where you may be part of a company team, a family team, or even a volunteer team.

As the year comes to a close, I challenge you to look at how you can become a better team member. Whether you’re leading or following, whether it’s work or play, here are some questions that you might ask yourself:

- Are you doing everything you can to support the goals of the team?
- Do you show up prepared?
- Do you follow through with your commitments?
- Can your team count on you to do your best?
- Do you encourage others to contribute?

- Do you take responsibility for your mistakes and come to the table with solutions?
- Are you content to let the team get credit rather than you as an individual?
- Do your teammates know that you value their contributions?
- Do you know when to ask for help when you need it?
- Do you volunteer to help others as often as you are capable—even if it’s outside of your domain?



Lisa Loper

From my vantage point as NoCOUG president for most of the year, I have a new appreciation for how smoothly we operate together as a team. NoCOUG would not be the active and successful organization it is without the teamwork we have and the wonderful people who volunteer their time to help make the team a success.

I want to extend a big thank you to our NoCOUG board members for all of their work this year in serving the NoCOUG membership. I look forward to continuing to be a part of the NoCOUG board team in 2008! ▲

P.S. Here are a few dates to put on your calendar. October 25: NoCOUG Fall Conference at the CarrAmerica Conference Center in Pleasanton (www.nocoug.org/next.html). November 11–15: Oracle OpenWorld. The Independent Oracle Users Group will have a booth at this event, so be sure to stop by and say hello.



(DIANE LEE)

Winners at the Summer Conference; prizes included books, e-books, DVD players, gift cards, and a Sony PlayStation. Alan Williams from Autodesk (not pictured) won a full pass to Oracle OpenWorld 2007, worth more than \$2,000.

Asking Jeremiah

When you speak all these words to them, they will not listen to you either; when you call to them, they will not answer you.—Jeremiah 7:27



Jeremiah Wilton tackles our questions about offshoring, certification, RAC, and Oracle 11g. Come and hear Jeremiah deliver the keynote address at NoCOUG's fall conference on October 25 at the CarrAmerica Conference Center in Pleasanton.



That's a great picture. You must be an Oracle geek then?

Sure, it may be geeky, but having a license plate that says ORA DBA makes all my DBA buddies jealous. It would have been ORA-600, but the DMV said that collided with the namespace of an existing series of plates. I believe in flaunting my geekiness.



Do you have any interesting stories about your time at Amazon? We love stories.

Working at Amazon.com was one exceptional experience after another. There are so many extraordinary stories that I cannot possibly relate them all here, but I'll tell a few yarns . . .

November 1999 brought the World Trade Organization's ministerial meetings to Seattle, as well as up to 100,000 anti-WTO protestors who turned out on the streets of downtown Seattle. Police and protestors clashed, and eventually police used tear gas against protestors in front of our building. The ventilation system pulled the tear gas in from the street and distributed it throughout Amazon's offices. Staff evacuated the building only to find themselves in the middle of the mayhem on the streets.

Then there was the time that same year when the roof

¹ Jeremiah's answer from the "Ask the Oracles" column in the May 2006 issue of the NoCOUG Journal.

drains on our building became clogged during the rainy season and a giant lake of water collected on the roof. The lake remained unnoticed until early one morning when I arrived at work to find water cascading through the ceiling of the server room and my office onto racks of NetApp appliances. We fashioned rain shields from plastic sheeting and protected the critical equipment. The Amazon site experienced no outages that day.

Then there was the time we were in the middle of a critical database recovery, applying archive logs with the website down in an unplanned outage. The fire alarm went off, and the server room's fire suppression system began sending messages to our pagers that it had been activated and that the room doors had been locked down. We left the recovery going, automatically applying logs, and hoped for the best. Out on the street waiting for the fire department, we noticed the server room HVAC contractor walking quickly to his truck. Chased down and questioned by the data facilities manager, the contractor admitted to activating the system by accidentally releasing refrigerant gas. The recovery completed while we were waiting outside and we were able to bring the site back up as soon as we returned.



Career highlights? Times when you felt that it was all worthwhile?

In 2000 at OpenWorld, I was honored and grateful when Oracle granted me an honorary Oracle Certified Master designation, along with seven other lucky luminaries.

Additionally, working at Amazon allowed me to work alongside some of the most talented technical minds in the world. Because of Oracle's commitment to Internet technologies, I was also afforded exceptional access to Oracle's senior staff and internal information.

But above all, watching the little Internet startup with no health insurance and dinky salaries become the Internet's most successful and respected merchant, through hard work and unwavering standards of quality, was the greatest reward of my career.

24x7



Is 24x7 a myth?¹

High availability is no myth. But while you can take many steps to create a stable database, true availability is measured from the end user's perspective, several layers removed from the da-

tabase. High-availability solutions at the database level such as RAC and Data Guard, while potentially important tools in creating stability for a given database, are not the starting point of any larger software-based high-availability solution.

The best availability in the industry comes from application software that is predicated upon a surprising assumption: The databases upon which the software relies will inevitably fail. The better the software's ability to continue operating in such a situation, the higher the overall service's availability will be.

But isn't Oracle unbreakable? At the database level, regardless of the measures taken to improve availability, outages will occur from time to time. An outage may be from a required upgrade or a bug. Knowing this, if you engineer application software to handle this eventuality, then a database outage will have less or no impact on end users.

One way of engineering applications to maximize availability in a down-database situation is through availability testing. Imagine an ordering system with two databases—one for shopping carts and one for orders. If the application software were to time out or encounter a fatal error accessing the database, it would write to a common piece of local state, such as a shared memory segment, with a flag indicating a problem on the target database. Then other threads of the application, before attempting to make a call against the database, would check this state object and forego the database call in order to avoid encountering the same error.

In such an example, failure of the ordering database would not cause hanging or termination of the application thread, and normal use of the shopping cart could continue. Conversely, if the shopping cart database were to become unavailable, then those users already in the order pipeline would be able to complete their orders without the application failing.

Another way to avoid failure-prone database access is to employ caching. An application should not keep going back to the database for the same data over and over again. For instance, a well-engineered application might only query a currency exchange rate once an hour. It would then maintain that data in memory and use it to service many requests, instead of repeatedly and inefficiently querying the database. Beyond improving availability, caching has the added benefit of reducing load on the database.

In summary, there are many ways to improve a single database's availability. But the highest availability comes from thoughtful engineering of the entire application architecture.



because performance did not improve in the manner he expected.

The sales rep took my manager to Four Seasons for lunch and, ever since then, my manager insists that RAC is the answer to all our needs. I've become quite the expert in "hanganalyze" and "systemstate" dumps. My manager thinks there's something wrong with me because

While it is true that RAC has been sold as a silver bullet for both scalability and availability, the deployment of any technology absent a valid engineered reason for deploying it is a

"Many applications are designed in a way that will actually provide worse scaling and availability on RAC. For instance, an application that makes many concurrent changes to a single table row cannot be efficiently scaled across nodes."

recipe for disappointment.

RAC may be a component of an overall scaling strategy for your organization, but many applications are designed in a way that will actually provide worse scaling and availability on RAC. For instance, an application that makes many concurrent changes to a single table row cannot be efficiently scaled across nodes. Along the same lines, an application that is not easily adapted to TAF or FAN, or simply reconnecting on connection failures, will not easily gain improved availability from RAC for the majority of failures. Finally, since RAC provides no redundancy for the database, data-related problems can cause availability issues across all nodes of a RAC cluster.

A good way for your manager to make a valid decision about RAC as a standalone solution for availability is for him to think back over the types of failures that your application has suffered in the past several months or years, and determine whether the problem might have been addressed by RAC or not. Don't forget to include planned outages and upgrades!

Quo Vadimus?



Why Oracle?²

Often the answer is "not Oracle." Oracle is expensive and complex, and not for every application needing a database. In addition to the relatively high license and support costs, Oracle requires the services of database administrators, a very expensive and opinionated class of technical professionals. The open-source community has produced some fine packages that, while currently weaker than Oracle in features and scalability, have great usefulness in a broad variety of applications. Oracle's decades of concerted development have evolved a maturity that makes it the best option, by no means for all, but for certain enterprise-class requirements that cannot be met by other products.

Oracle, if you want a consistent view of data without blocking writers.

Other products have improved their locking and serialization models over the years, but Oracle is still the premier database providing multi-version read consistency (MVRC).

² Jeremiah's answer from the "Ask the Oracles" column in the May 2007 issue of the *NoCOUG Journal*.

Oracle's adherence to this model is long-standing, and it is a fundamental characteristic of the Oracle server. Oracle uses undo to provide users with a view of data consistent with respect to a fixed point in time, even if that data is being changed out from under the reader. This model allows writers to make changes to data seamlessly, even if others are reading it.

Multi-version read consistency becomes important as concurrency and scale of an application increase. Because readers do not block writers, applications can perform operations in many threads simultaneously without the data-base serializing work unnecessarily. In addition, MVRC allows, within limits, for reporting and decision support to take place in the same instance as OLTP operations.

Oracle, if you want the option of scaling across multiple inexpensive nodes.

With Oracle 9i, Oracle became best of breed in shared-everything cluster databases. While most other clustering technologies require or encourage users to segregate their data and

“Your timeline for upgrading to 11g should be driven by business needs and release stability. Unless you urgently need an 11g new feature to support your business, then you will benefit from waiting for at least the first patchset, and preferably the second.”

not actively share data across hosts, Oracle's RAC does away with such concerns to a degree far greater than any competing clustering model. Although scaling on RAC is not linear, and cross-node contention for heavily changed buffers is still possible, RAC opens up scalability to inexpensive units of architecture with greater ease and simplicity than are available on any other platform.

Oracle, if you want to tune without speculation.

Hands down, Oracle has provided among the richest instrumentation features of any enterprise software package currently available. It led the way in time-based performance analysis with the wait-event interface in Oracle 7. With Active Session History (ASH) in Oracle 10g, Oracle provides incredibly rich performance data by inexpensively sampling and tracking wait events, sessions, and cursors over time. Competing database products not only fail to provide a similar feature, but have not even caught up to the basic wait-tracking provided by Oracle in 1992. Performance tuning on many other database platforms is a guessing game.

While third-party wait-tracking products from Precise and Quest have been available for other database platforms for some years, they restrict users to their GUI interface, and refuse to publish their agent API. In Oracle's case, the data is

accessible via tables and views, and can be queried just like any other data. The sampled data is thus available for scripts, monitoring, and other repeatable operations.

By no means are read-consistency, shared-everything clusters, and wait sampling the only features that differentiate Oracle from competing packages. No doubt you have your own favorites. But the above features are so fundamental to providing availability, scalability, and stability to an application that their absence in competing products, especially those from private commercial sources, makes Oracle that much more competitive.



I've been taking short trips on the dark side (SQL Server) and have seen a lot of wonderful things there. My favorites are cumulative patch sets (service packs) and the "shrinkfile" command. Isn't there anything at all about Oracle that you would like to change? Come now, we're all friends here.

Well, that X-windows installer (OUI) is certainly annoying. I wonder how many person-hours have been wasted by people fiddling around trying to get that to work over a distant network link. Those who can recall "orainst" will attest that OUI was a giant leap backwards in ease of use and efficiency.

Oracle's process of applying patchsets and upgrades also needs to be improved. They currently require too much downtime. I understand Oracle is making strides in this area and that improvements are imminent.

Another request I would make is for ASH to collect data on session statistics and segment statistics (v\$sesstat and v\$segment_statistics) over time in addition to waits.

I'll also mention support. Customers who have paid large sums for a support contract should receive service commensurate with the cost. Too often customers receive boilerplate responses or solutions that seek SR closure at the expense of customer time, availability, and stability rather than directly addressing the problem.



The sales rep has made another lunch appointment to take my manager to Four Seasons and I know that when he comes back, my manager is going to be saying that 11g is the answer to all of life's little problems. I'm a little skeptical: what's wrong with me? Is 11g the answer to all our prayers? Should we plan on upgrading soon?

Your timeline for upgrading to 11g should be driven by business needs and release stability. Unless you urgently need an 11g new feature to support your business, then you will benefit from waiting for at least the first patchset, and preferably the second. With each patchset, you will benefit from additional bug fixes without which you could have had a production issue. I beta tested 11g, and it is neither overly buggy nor perfectly stable. Like many Oracle releases since 9.2, it is stable enough for lightweight production environments in the first release. However, 11g's signature feature, Real Application Testing, will not be as useful as one might have hoped

ROUNDSTONE systems



Delivering Information Technology from a Business Perspective

Roundstone Systems works with businesses throughout northern California to optimize their data centers, equip headquarters and branch offices with the right computer hardware, software and ancillary products, and provide the tandem services necessary to assess, deploy and manage the technology investment. Moreover, in our team you will discover a pleasant, helpful attitude that will foster a relationship over time.

As a channel partner, we are an extension of HP to our customers. When you work with Roundstone, you will benefit from the performance of HP's award winning products, and gain pain-free access to the myriad experts within HP. Because we are integrally linked to the manufacturer, we are able to act as customer advocate, identifying the right resources, and delivering a timely response.

Roundstone is a proud sponsor of the Northern California Oracle Users Group.

Gold Business Partner  invent

510-764-2080
www.roundstonesystems.com

in a 10g to 11g upgrade, since the database replay feature has no 10g capture ability.

The Scientific Method



*Help! My database is slow.*³

For this to be any more familiar a refrain to most DBAs, it would need only to be rephrased as “Hey! Your database is slow.”

A shocking number of DBAs approach system slow-ness with a litany of possible causes and solutions. They check for high CPU utilization, high load averages, low free memory, locking, latches, bad explain plans, etc. The list goes on and on. I call this method the “guessing method.”

“I believe that the focus on experience in specific technologies and seniority causes employers to pay more and get less than they could when filling DBA positions.”

To hear the guessing method directly from one of its proponents, simply conduct job interviews with ten DBA candidates. Eight will be active users of the guessing method.

Using the guessing method, how can a DBA possibly check for every one of the many things that could possibly go wrong? Take the “bad explain plans” laundry list item for instance. How will the DBA know which of the thousands of cursors in the SQL area to explain? Once explained, what criteria will the DBA use to declare with certainty that the plan is “bad”? Simply put, the guessing method lacks any ability to empirically determine the cause of a problem, and serves only to make the DBA look busy during a crisis.

In the Oracle world, there is no excuse for employing the guessing method. Oracle is among the best instrumented commercial software products available. Oracle’s wait events allow DBAs to instantly see if there is a problem with response time and trace it to a specific cause. With the Active Session History (ASH) feature in 10g, wait information can be queried and trended over recent time to show when a problem began and ended, and also how it affected the database. ASH allows DBAs to effectively respond to that other familiar refrain, “Hey, your database *was* slow!”

In the majority of cases where the problem in fact lies with the database (and usually with a subset of sessions and not the whole database), simply looking at the current and recent session waits will point to the problem with no additional searching required. “Broken” plans performing full table scans will show up as excessive “db file scattered read” waits. Locking will show up as “enqueue” waits, and so on. For those aficionados of the guessing method, this begs the question: why spend your time poking around when you can go straight to the source?

There are cases of slowness where wait events can be less

³ Jeremiah’s answer from the “Ask the Oracles” column in the November 2006 issue of the *NoCOUG Journal*.

useful in solving the problem. Host resource starvation (CPU, memory) is one variable to rule out while investigating slowness. However, automated monitoring should always alert someone long before resource starvation affects application users. If staff is spending time looking at CPU and memory on the hosts, better-automated monitoring is probably needed.

In fairness, not all application slowness (only most) is Oracle’s fault. To make rapid determinations of time spent on an application, developers should take a page out of Oracle’s book by including Oracle-like instrumentation and timing. Applications and middle tiers that log timing information can be consulted during slowness to see where the application is spending time. If it is spending time querying Oracle, then the DBA can look into Oracle waits. If it is spending time on some other task, then engineers can follow the arrows to the appropriate service.

In cases of tuning slowness, an alarming lack of empiricism plagues a wide swath of Oracle DBAs. By using timing-based analysis, you can answer the simple question that is begged by every instance of slowness: “Where is all the time going?”

Career Planning



My daughter’s piano teacher likes to say that practice makes permanent, not perfect. Just because I’ve been a database administrator a long time doesn’t qualify me as a “senior” database administrator—or does it? Who is a “senior” database administrator? Do I need a college degree? Do I need to be a “syntax junkie?” Do I really need experience with Oracle Streams or ASM to claim the title?

To me, senior means that you have used a lot of Oracle’s features, solved a lot of problems, and experienced a variety of production situations. Do these qualities necessarily mean that I will want to hire you? No.

Of far greater importance than seniority is a DBA’s ability to solve problems in a deductive and logical manner, to synthesize creative solutions to problems, and to forge positive and constructive business relationships with colleagues and clients. For years at Amazon, we simply tried to hire extraordinarily smart people with a strong interest in working with Oracle and others. Some of Amazon’s most senior DBAs started with little or no Oracle experience. I believe that the focus on experience in specific technologies and seniority causes employers to pay more and get less than they could when filling DBA positions.



What about OCP certification exams? They seem to test your knowledge of syntax more than anything else. I know database administrators who have OCP certification but can’t be bothered to change passwords regularly or test database recovery. Also, you can’t get certified without spending a lot of money on Oracle training. Is it money well spent?

None of the clients for whom I recruit care if a candidate has an Oracle certification. Unless I was trying to get a job that required it or my employer required it and was willing to pay, I wouldn't bother. I think you said it better than I could—OCP certification doesn't mean you are a better DBA than a non-OCP; it just means you can pass a questionably formulated standardized test. I have even been at sites where the OCP certification is considered a red flag rather than a qualification.



NoCOUG membership costs only \$80 for four training days and four issues of the NoCOUG Journal but we have a hard time convincing Oracle professionals about the value of membership. Perhaps they'll believe you if you say it!

Referring back to your earlier question, local user group membership and participation is one of the ways that junior DBAs can become senior. Compared to high-priced national conferences, local user groups like NoCOUG provide DBAs

“I think that offshoring may push some of the less-qualified positions out of the country, but that plenty of opportunities will remain for DBAs that are skilled, experienced, and intelligent.”

with affordable access not only to their peers in the local community, but also to well-respected authors and speakers from the global Oracle community.

Local user groups are also a great way to keep up with how other people are solving problems, and what technologies are emerging. It is hard to remain good in a field when you are in a bubble. Local user group membership and attendance gets you out of your company bubble and exposes you to new ideas and trends.



Big companies are sending all the jobs to offshore sweatshops. Are they really saving any money? Has quality suffered? Isn't manageability an issue? Do I need to worry? Will I have a job in five years?

There are a lot of forces at work in the offshore DBA job market. The quality of offshore DBAs in India and China is increasing every year, but so are their salaries. Their salaries are still a fraction of U.S. salaries, but it is not just about money.

Only a small number of companies are willing to use remote DBAs at all. Most still prefer the DBA to be sitting at a desk in the company's offices. Those who do use offshore DBAs still rely on offshore DBAs mainly for overnight on-call work. At the very least, most maintain a core team of local professionals to drive direction, engineering, and standards. Most companies have a primary DBA team in the U.S. and a

secondary one overseas.

I think that offshoring may push some of the less-qualified positions out of the country, but that plenty of opportunities will remain for DBAs that are skilled, experienced, and intelligent. I think the positive outcome of overseas competition will be U.S. DBAs striving to achieve excellence in areas where they once might have too lazy to bother.

Parting Wisdom



Good Oracle books to read? Good online resources? Favorite conferences?

My favorite Oracle book is *Oracle Insights: Tales of the Oak Table*. The Oakies really put together a great collection of essays and I hope they'll follow up with an equally compelling sequel (no pun intended).

I am a participating member of Oracle-L, the listserv hosted by freelists.org and managed by Steve Adams. This list is a great place to be able to ask questions directly to some of the top talent in the field. The discourse on Oracle-L is polite and respectful compared to the kind of juvenile abuse that is the norm on newsgroups like comp.databases.oracle.server.

For conferences, I prefer local users group conferences in terms of value for money. Also a lot of pretty brilliant people seem to coalesce at UKOUG. OpenWorld is slick and fun if you can afford it, but you have to go with the understanding that it is primarily a marketing event.



Thank you for spending so much time with us. Do you have any parting advice for us? Perhaps a favorite quote?

Thanks for the interview. I always enjoy participating in NoCOUG. My only advice to others is this: The two worst problems I see today among DBAs are adversarial attitudes toward others in their organizations and complacency with respect to their own skills. Build bridges and build your skills, and you will have no trouble being productive in your work and remaining gainfully employed regardless of the offshore DBA phenomenon or other market forces. ▲

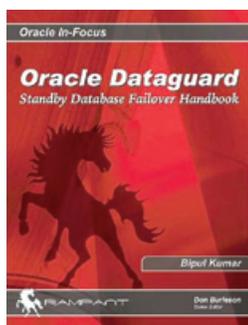
Jeremiah Wilton is principal consultant and owner of ORA-600 Consulting. Before becoming independent, he spent seven years at amazon.com, initially as their first database administrator in 1997. Using expertise he gained during amazon.com's period of exponential growth, he now specializes in scalability, high availability, stability, and complex recoveries. Jeremiah has presented at numerous conferences and user group meetings, and is the author of a variety of technical whitepapers and articles. His publications are available at www.ora-600.net.

Oracle Dataguard: Standby Database Failover Handbook

Reviewed by Brian Hitchcock

Details

Author: Bipul Kumar
ISBN: 0974599387
Format: Paperback
Pages: 230
Year of Publication: 2005
Edition: 1
Price: \$27.95
Publisher: Rampant Techpress



Overall Review

The best reason to read an Oracle book is that you actually need the information. This also provides a good motivation to review such a book. Since I was (suddenly) tasked with supporting Data Guard for several of our Oracle environments, I was motivated to read carefully and put into action what I learned.

From this viewpoint I believe this book offers a good value. It isn't as thorough as you might like, but it gets to the point quickly and doesn't take a long time to read. When I was looking for a book about Data Guard, I didn't find very many. Other than this title, there is a section in the Oracle Press book *Oracle Database 10g High Availability with RAC, Flashback, and Data Guard*. I chose to focus on Data Guard only. While this book is somewhat out of date (published in 2005), that may have helped me, because the book covers how Data Guard 9i works and then adds the new features of 10g. Since most DBAs I know (but would they admit to knowing me?) support many Oracle databases that aren't 10g (yet?), I think covering both 9i and 10g would be valuable to most of the intended audience.

Note that using Data Guard means you are tying two (or more) databases together. This means you will need to review your support and monitoring procedures to reflect the new dependencies between primary and standby databases, as well as how to handle failover and switchover. This book does not cover these issues.

Chapter 1—Introduction to Oracle Data Guard

This introduction offers a review of failover options from 1990 to the present, including the options besides Data Guard such as Oracle Parallel Server, Replication, RAC, and Streams. The discussion includes the issue of Oracle licensing, which is an example of the kind of practical information available in

this book. The pros and cons of manual standby using scripts (which avoids licensing costs) are discussed. Standby database concepts are presented, covering when the standby is read-only and a history of the specific Oracle standby database products in each of the major Oracle database versions 7.3 through 9i. A later chapter updates this for Oracle 10g. A detailed description of physical standby versus logical standby, and a comparison of the advantages and disadvantages of Data Guard including cost (hardware and software), are given next. This is followed by the operational constraints that Data Guard imposes, since you now have two databases that are intimately linked. The creation process is reviewed at a high level, including a sample configuration.

Chapter 2—Data Guard Architecture

This chapter provides a more detailed discussion of exactly how the process works as changes on the primary database are moved to the standby database. Each of the components of Data Guard is discussed, along with the critical role that the redo log plays in this process. The two ways that redo logs can be applied to the standby database are covered (redo apply and SQL apply) for the physical and logical standby respectively. A brief description is given of how Log Miner is used to extract SQL from the redo logs from the primary database. The issue of gaps in the sequence of redo logs from the primary is covered, as well as how Data Guard handles this. The architecture and components of Data Guard Broker are discussed. This is an Oracle utility that can be used to manage the Data Guard configuration from either the command line or as part of Enterprise Manager. *Failover*, the process in which the standby takes over when the primary fails, versus *switchover*, where the role of primary database can be switched back and forth between the primary and standby databases, is discussed. Switchover is a feature of Data Guard that you don't have if you are implementing a standby using your own scripts; it can be useful for testing failover and in certain upgrade scenarios.

Chapter 3—Implementing Standby Databases

Descriptions of a physical standby, logical standby, and setting up a standby using manual recovery—which means writing your own scripts to transport and apply archived redo logs to the standby—are covered. The physical standby is the more traditional standby in that it is constantly applying, or waiting to apply, the next archived redo log from the primary, and does not allow database users to connect even for read-only. You can, however, stop the process of applying archived

redo logs and the physical standby will allow read-only database access. The physical standby is an exact copy of the primary. The logical standby is a newer feature whereby archived redo logs from the primary are shipped to the standby and LogMiner is used to extract the SQL statements that caused changes in the primary database. These SQL statements are then applied to the standby database. This means the process that applies the SQL statements is just like any other user connection to the database, and the standby database can be open for users at the same time it is applying changes from the primary. Data Guard will prevent users from making changes in the logical standby database of objects that are being maintained (guarded) by Data Guard.

The additional details of creating a physical standby on the same machine as the primary are covered. Basically, this means the standby must use different filesystems than the primary, and this requires setting up `init.ora` parameters that will map between the primary and standby filesystems.

The logical standby does have some limitations compared to the physical, and these are covered here as well. The logical standby does not support all the data types that the physical standby does.

Chapter 4—Standby Database Administration

Issues covered here include general administration and monitoring. For example, once you install Data Guard for a standby database, the primary and standby databases are intimately related. If the standby database becomes unreachable for any reason, an “oracle unavailable” message appears in the primary database alert log. Since you may have existing monitoring that scans the primary database alert log looking for errors and warnings, it can be very upsetting the first time you are paged because the primary database is “unavailable,” only to find out that it is really just an informational message about the standby database, not the primary. Depending on the criticality of the standby, this specific message may or may not be critical.

General administration of Data Guard covers starting and stopping the process of applying archived redo logs to the physical standby database, starting and stopping the SQL apply process for a logical standby, and shutting down the standby altogether. Again, since the primary and standby are linked under Data Guard, if you simply shut down one or the other, you may get messages from the other database. You need to develop your own support processes to handle the new complexities of Data Guard. Also covered is the fact that triggers

that fire in the primary database don't fire in the standby, and primary database constraints aren't enforced in the standby. How to skip tables is explained, which means that you can tell Data Guard not to replicate specific tables into the standby. This feature, combined with the constraints not being enforced, means you can configure a standby such that it doesn't maintain the referential integrity of the primary.

In the event of a switchover or failover, details of what must be done are presented. This depends on the particular scenario you are in, and there are several. You have to worry about archived redo logs that haven't been shipped to the standby or that haven't been applied to the standby, and so on.

This chapter also covers how you can set up cascading standby databases in which a primary database sends archived redo logs to a standby database, which then sends them to one or more additional standby databases.

Chapter 5—Log Management Services

This chapter covers details of how to configure various options for getting the archived redo logs to the standby, as well as protection modes. Protection modes refer to what Data Guard will do if the primary can't send an archived redo log to the standby. If you configure Maximum Protection and the standby isn't available, the primary database will eventually stop, since it can't reuse a redo log until it has been copied to the standby. This reminds me of two-phase commit, but I doubt anyone else is old enough to remember that! Maximum Protection mode is only available for the physical standby. Maximum Performance, which is the default, means the primary database will try to send the archived redo log to the standby, and if it can't, it keeps going. A third available option is covered: Maximum Availability, which is a combination of the two previous options. In this configuration, the primary will try to write to both the local and standby archived redo log destination, but if it can't, it will move on. However, unlike Maximum Performance, as soon as the primary can communicate with the standby again, it will, and it will ensure that the archived redo logs are sent to both primary and standby as long as possible.

This chapter also covers the options for the shipping of the archived redo logs themselves. By default, Data Guard will wait until the archived redo log is complete on the primary before shipping it to the standby. You can also configure what is called *standby* redo logs on the standby database, and Data Guard will ship the entries from the primary archived redo log just as fast as they are written to the primary.

“I have seen reviews of books that say the book is simply a rehash of the manuals. By definition, to some extent, this will always be the case, assuming that the Oracle manuals are any good at all. However, to make this criticism is like saying you don't need a map of San Francisco because it's just a rehash of the globe. True, but if you need to find the NoCOUG conference at the PG&E building in the rain, you will find a map of San Francisco very useful.”

An interesting detail of the logical standby is that since it is a normal database that is open for user access as well as applying the SQL from the primary database, it is also generating its own archived redo logs. This can be confusing. On the standby database server, you need to manage disk space for both the archived redo logs shipped from the primary database and the archived redo logs that are generated by the logical standby database itself. Both of these locations on disk can run out of disk space, and you need to be backing up the archived redo logs of the logical standby database to support recovery. This is critical if you add any database object to the logical standby database. This is a feature of the logical standby. You can add schemas—for reporting, for example—that are not present in the primary. These additional schemas might be used to store aggregates of primary table data, and so on.

Chapter 6—Switchover and Failover

An advantage of Data Guard over your own scripts that maintain a standby database is the ability to switchover. This means the primary and standby databases can swap roles back and forth. This is useful for testing failover and upgrades. When using scripts to maintain the standby, you can only failover, which means that the standby database is opened. After this, you can't switch back; i.e., the former standby database, which is now the primary, can't become a standby again without recovering the previous primary database and then rebuilding the former standby from the recovered standby. For Data Guard, failover means the same thing, but by using switchover, the primary and standby databases can swap roles repeatedly.

This chapter covers the details of the role transition process, by which switchover (or failover) is accomplished. This includes details for transitioning to a logical standby versus the more typical physical standby. Note that switch-over (or failover) to a logical standby, such that the logical standby becomes the primary database, can be problematic. Recall that a logical standby has limitations, such as non-support of some data types. If the primary database uses any of the data types that Data Guard doesn't support, you can't switchover (or failover) to the logical standby because it can't function as a complete re-creation of the primary database.

Chapter 7—Performance Tuning of Data Guard Configuration

When using Data Guard you need to be aware of the possible performance impacts on the primary database, as well as the possible impact on the standby database users if the standby isn't kept up-to-date relative to the business requirements. Monitoring Data Guard performance focuses on the log transfer service, which gets the archived redo logs from the primary to the standby, and the log apply services, which apply the transferred redo logs to the standby database.

Scripts are provided to get performance metrics from various views for monitoring the progress of both redo log transfer and the redo log apply processes.

If the Data Guard environment is configured for maximum protection, the primary database may be forced to stop if there is a significant delay in transferring and applying the redo logs to the standby.

From the perspective of the standby database, performance means getting out of synch with the primary. This requirement will be defined by the business requirements. A standby that must be up-to-date within 5 minutes of the primary at all times will have different performance issues than a standby that is not updated during business hours and must catch up each night. Clearly the network bandwidth available is a factor, as is the use of logical standby, which must scan the redo logs and apply individual SQL statements.

Specific tuning advice for the log transfer process covers network bandwidth, TNS parameters, and I/O issues on the standby server. For the log apply services, the tuning advice focuses on the logical standby and covers LogMiner memory usage, impact of full table scans (caused by the primary table lacking a primary or unique key), transaction consistency, and shared pool size.

Chapter 8—Data Guard Broker

This chapter covers the server-side and client-side components provided by Oracle to help monitor (and create) Data Guard environments. There are command-line and Enterprise Manager versions of each component. The process of using these components to configure, monitor, and maintain a Data Guard environment is covered. Differences between the 9i and 10g features of the Broker are presented. Data Guard Broker can also be used to execute the role transition between primary and standby databases.

Chapter 9—Recovery Manager and Data Guard

Since building the Data Guard standby database starts with obtaining a backup of the primary database, RMAN can be instrumental in administering a Data Guard environment. Incremental RMAN backups can be very useful in reducing the time needed to create a standby for a very large primary database. RMAN terminology and the process of creating a physical standby database using RMAN are covered.

RMAN can also be used to back up the standby so that backups don't have to be made on the primary. Details of how to recover the primary database from RMAN backups made on the standby are covered. While this chapter is not a complete reference for RMAN, it points out the advantages of using RMAN and identifies the main issues that must be addressed.

Chapter 10—Oracle 10g: New Features of Data Guard

This chapter covers specific changes made between Data Guard 9i and 10g. Examples of what is covered include real-time apply and simplified creation of a logical standby. Real-time apply means the log apply services don't need to wait for a complete archived redo log from the primary before applying the redo log information to the standby. This means the standby can be kept more up-to-date relative to the primary. This feature uses standby redo logs, which are created on the standby database. Data Guard 10g makes creation of a logical standby easier by not requiring a shutdown of the primary database. Syntax changes allow creation of the logical standby control file, which contains the information needed to establish the logical database without a primary database shutdown.

Appendix A—Data Guard Reference

This covers init parameters, dynamic performance views, and PL/SQL packages used by Data Guard.

Appendix B—Troubleshooting Guide

This presents procedures to solve common issues for log transfer services, SQL apply operations, and missing temp files, among others.

Conclusion

As with the other book I've read from Rampant (*Oracle Performance Troubleshooting with Dictionary Internals SQL & Tuning Scripts, Oracle In-Focus Series* by Robin Schumacher, published by Rampant TechPress in 2003; see www.brianhitchcock.net for a review), this book is a good value. You learn what the author has to tell you in an efficient manner. Yes, some detail is left out, but I liked the fact that I could get through the book quickly. I have seen reviews of books that say the book is simply a rehash of the manuals. By definition, to some extent, this will always be the case, assuming that the Oracle manuals are any good at all. However, to make this criticism is like saying you don't need a map of San Francisco because it's just a rehash of the globe. True, but if you need to find the NoCOUG conference at the PG&E building in the rain, you will find a map of San Francisco very useful. For my purposes, I needed a quick overview of how Data Guard works, what the basic options are, and how to set it up. I got all of that from this book.

The writing could use some further editing. There are more apparent typos than I am used to in the Oracle Press books,

but this didn't detract from the value of the information. I would have benefited from more diagrams that showed in detail how the redo information flows between primary and standby, and I would have benefited from more details about the setup and refresh processes.

A final testament to the value of what I learned from this book is that I was able to review the architecture that had been put in place by the consultants and offer an alternative approach that was simpler for my group to support. The consultants agreed that the alternative approach would work, so I believe I got what I needed from this book. ▲

About the Author(s)

Bipul Kumar is a senior application engineer for a telecommunications company. He is responsible for Oracle database administration, performance tuning, recovery, and application development. He has more than six years of experience in creating and managing Oracle databases of varying sizes, primarily on Unix operating systems.

Brian Hitchcock has worked at Sun Microsystems in Newark, California, for the past 11 years. He is a member of a DBA team that supports 2400+ databases for many different applications at Sun. He frequently handles issues involving tuning, character sets, and Oracle applications. Other interests include Formula One racing, finishing his second Tiffany Wisteria lamp, Springbok puzzles, Märklin model trains, Corel Painter 8, and watching TV (TiVo rules!). Previous book reviews by Brian and his contact information are available at www.brianhitchcock.net.

Copyright © 2007, Brian Hitchcock



We can't change the past, but... we could have predicted it!

- Analyze performance by workload/application
- Predict performance shortfalls in the next 12 months
- Understand the effects of change: new apps, 10g, more users, new hardware...in seconds

BEZ systems The Prediction People™



Real-World Experience

For Oracle database consulting and support, it makes sense to work with a company that has a proven track record. Since 1995 our clients have relied on us for:

- Performance tuning
- Migrations and upgrades
- Backup and recovery strategies
- Database security audits

Plus, we offer ongoing **remote DBA** support plans that are tailored to your business needs and budget.

Call Us Today!

(415) 344-0500 • (888) 648-0500

www.dbspecialists.com


DatabaseSpecialists

ORACLE | CERTIFIED SOLUTION PARTNER

The Future Is Now! CA Database Command Center to the Rescue

by Steve Lemme

Introduction by Iggy Fernandez

Until not so long ago, we got to work at 8 a.m. and left work at 5 p.m.

Until not so long ago, we had no access to the databases from home.

Until not so long ago, all the members of the team worked in one location.

Until not so long ago, most companies only had a few databases.

Until not so long ago, most companies used only one database technology.

Until not so long ago, the business did not need the databases on nights and weekends.

Those days are gone.

The number of databases has increased tremendously.

Now, SQL Server databases coexist side by side with Oracle databases.

Now, the business needs the databases around the clock, during the night and on weekends and holidays.

Now, database administrators take cell phones and wireless-equipped laptops with them when they go fishing.

Now, distributed teams, outsourcing, and offshoring are the order of the day.

The future is now!

CA Database Command Center to the Rescue

Computer Associates recognized the challenges faced by the modern database administrator and developed the CA Database Command Center (CA DCC), which enables enterprise-wide database monitoring, diagnostics, and problem resolution from within a single console. CA's DCC browser-based interface supports schema browsing and alteration capabilities for SQL Server, Oracle, and DB2 for Linux, Unix, and Windows databases, as well as DB2 for z/OS databases.

Available for download at no charge at www.ca.com/databasemanagement, CA DCC provides a consistent look

and feel across database platforms. It enables both inexperienced and seasoned database administrators to adopt new techniques that enable effective cross-database best practices.

“Available at no charge, CA DCC provides a consistent look and feel across database platforms.”

Administrators familiar with one vendor's RDBMS can perform similar work on another RDBMS—even if they have not used it before. Through a common interface and action-driven wizards, databases are managed in

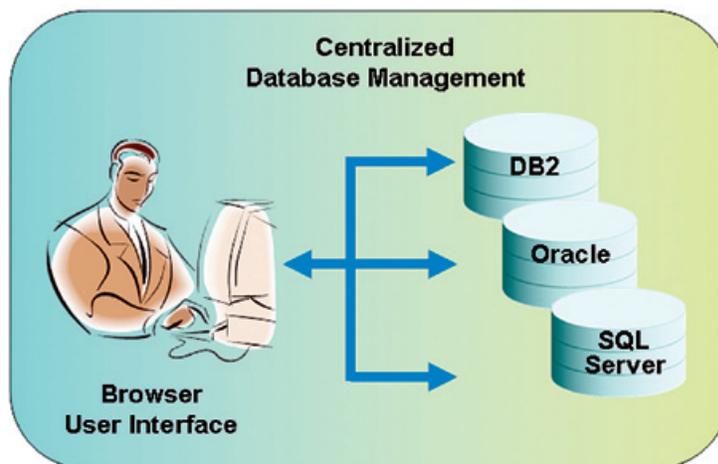
a unified and simplified manner. This allows true cross-database management for increased availability and greater productivity.

Database Agnostic

CA DCC supports multi-vendor database platforms, enabling schema manipulation across all major relational databases. Administrators can create, alter, rename, copy, and

drop mainframe and distributed database objects, and perform many other database-specific functions. Unlike some tools that use non-standard GUIs or functionality limited to a single proprietary database architecture, CA database utilities are both vendor and platform neutral. Administrators can reduce or eliminate use of standalone utilities and multiple customized scripts. While some tools are tied to specific database vendors and releases, CA DCC

is database independent. When additional database management functionality is required, CA DCC can provide it directly, without the need to upgrade database installation or purchase database-specific options.



Whether DB2, Oracle, or SQL Server, database management tasks have a common look and feel within the CA Database Command Center.

Browser-Based

Since CA DCC is browser-based, administrators can access, manage, and monitor their databases from virtually anywhere. There is no user-based console counting or licensing, or any need to install client software for each database or system.

Extensible and Role-Based

CA DCC presents information in a customizable portal workspace, utilizing standards-based J2EE technology powered by an Apache Tomcat Web Server. Portal functionality enables administrators to easily add external sources of information into the console or customize the workspace by user. CA DCC provides an interface customization capability—both by user and role—eliminating the constraints of a fixed interface. For example, each database administrator can include specific SQL reference guides or set up a standard workspace for junior administrators to view test and development databases.

Single Sign-On

CA DCC stores and encrypts login information for each managed database. This removes the need to constantly reenter login credentials each time work is performed with a particular database. This helps safeguard database integrity and security without burdening the administrator.

Advanced Integration

In addition to the single console that provides a common look and feel for database object alteration, built-in integration and common services make CA DCC easily extensible. A modular yet integrated approach allows functionality to be easily added when it is needed. As the need for greater operational capabilities grows, additional multi-vendor database functionality can be licensed based on where databases reside. There is no extra charge for the number of database instances on a specific server, nor are there charges by database vendor type.

Advanced integration functionality, available for additional licensing fees, integrates in context with CA DCC for increased multi-vendor database management capabilities.

Database Monitoring and Performance Management

As the console, CA Database Command Center, has integration with other products that can be purchased from CA, enabling even greater management capabilities. With CA Insight Database Performance Monitor for Distributed Databases (CA Insight), administrators can automate monitoring, data collection, and performance analysis of all monitored databases within one summarized display. CA Insight pro-

vides threshold capability and alert notification when an event occurs, enabling administrators to quickly identify, diagnose, and resolve issues in real time. Historical data collection trending capabilities help isolate recurring performance issues and enable additional advanced problem remediation for faster problem response and resolution.

CA Insight also includes advanced integration for CA Unicenter[®] NSM and CA eHealth[®] System and Network Management products. Much more than simple alert passing, CA Insight provides additional console visualization integration, sophisticated event management capabilities, and streamlined workflow for comprehensive and combined systems, network, and database management.

Fast Unload

CA Fast Unload[®] for Distributed Databases can be added to streamline data extraction and reduce the amount of time that database tables are unavailable to applications. Easy-to-navigate context-sensitive menus within the CA DCC guide administrators through the extraction process and enable extraction scheduling. CA DCC invokes the appropriate output format for easier and faster cross-vendor data re-loading.

Automated Reorgs

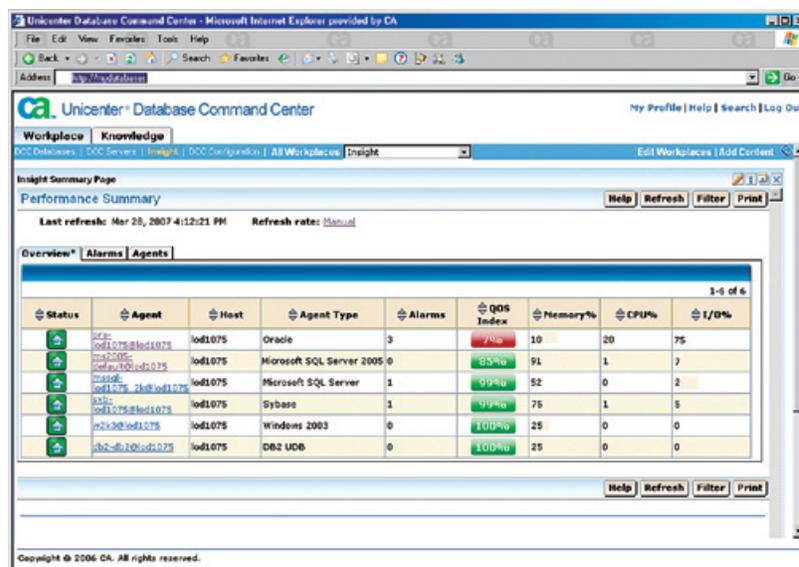
By adding CA TSreorg[™] for Distributed Databases, administrators can improve the efficiency and availability of databases by automatically restructuring

databases that have become fragmented due to data inserts, updates, and deletions. Availability of business-critical data and applications can continue during online reorganization.

Summary

From basic database discovery to the most complex administrative tasks, CA DCC employs a browser-based UI and wizard-driven workflow to guide administrators through tasks for working with multiple databases from a single console. Administrators experienced with a particular database technology can seamlessly translate that knowledge to all major databases without additional training. CA DCC addresses the challenge of managing multiple database brands across disparate platforms with a single tool. ▲

Steve Lemme is director of product management for CA Database Management, the published author of Implementing and Managing Oracle Databases, and a columnist in Database Trends and Applications. He also served as director for the Independent Oracle Users Group (IOUG).



The screenshot shows the Unicenter Database Command Center interface. The main content area displays a 'Performance Summary' table with columns for Status, Agent, Host, Agent Type, Alarms, QOS Index, Memory%, CPU%, and I/O%. The table lists several agents, including Oracle, Microsoft SQL Server, Sybase, and Windows 2003, with their respective performance metrics.

Status	Agent	Host	Agent Type	Alarms	QOS Index	Memory%	CPU%	I/O%
OK	lod1075@lod1075	lod1075	Oracle	3	7%	10	20	75
OK	ms-sql06@lod1075	lod1075	Microsoft SQL Server 2005	0	85%	51	1	7
OK	ms-sql07@lod1075	lod1075	Microsoft SQL Server	1	99%	52	0	2
OK	sybase@lod1075	lod1075	Sybase	1	99%	75	1	5
OK	win2k3@lod1075	lod1075	Windows 2003	0	100%	25	0	0
OK	db2@lod1075	lod1075	DB2 UDB	0	100%	25	0	0

One summarized view for Oracle, DB2, SQL Server, and Sybase.

Oracle Annoyances for Geeks: Deadlocks

by Iggy Fernandez

“It’s kind of hard to understand the rules. It’s like you can’t do nothing no more.”

—Football star Terrell Owens, after being fined for a flamboyant celebration after scoring a touchdown.



Iggy Fernandez

Case Study

I came across a book called *Windows XP Annoyances for Geeks* and wondered why there wasn’t a book about Oracle annoyances. Nothing’s perfect, Oracle included, and it always helps to know exactly where the deficiencies lie and how to work around them. Perhaps somebody at Oracle Corporation with the power to change things might read the book and the problems would get fixed some day! How about the inability of SQL*Plus to handle blank lines in long SQL statements? Did you know that the SET SQLBLANKLINES ON command forces SQL*Plus to process blank lines correctly?

I would definitely write extensively about deadlocks. They are annoying because Oracle could do a better job of generating the information needed to find out what is causing them; in fact, Oracle contributes to the misinformation about them. They are also annoying because applications don’t usually handle them gracefully, perhaps because they are not well understood. The end result is endless nagging by pointy-haired managers and countless hours wasted by database administrators.

I’d need lots and lots of space—space that I don’t have—to write all that it is possible to write on deadlocks, so I thought I might try the approach of a quiz today. Don’t worry if you can’t answer all of the questions; I learned the answer to some of them only when doing the research for this article. Detailed answers are provided after the quiz.

Deadlock Quiz

Here are a few questions to test how well you understand deadlocks. Pick the best answer in each case.

- Which of the following situations is an example of an Oracle deadlock? Pick all that apply.
 - A situation in which a blocking session becomes inactive but does not release its locks.
 - A situation in which a blocking session is itself blocked by another session.
 - A situation in which two sessions are trying to modify a row of data that has previously been modified by the other.
 - A cycle in the “wait-for” diagram.
 - A situation in which a session is blocking itself, i.e., “self-deadlock.”
- What kind of situation cannot result in an Oracle deadlock? Pick all that apply.
 - A situation involving a single table, because deadlocks cannot occur if everybody is modifying the same table.
 - A situation involving a single user, because sessions owned by the same user cannot deadlock with each other.
 - A situation involving a single session, because a session cannot deadlock with itself.
 - A situation not involving any locking activity whatsoever.
- Who receives the “ORA-00060: deadlock detected while waiting for resource” error message?
 - The participant that is blocking the others.
 - Only one of the participants in the deadlock, randomly picked by Oracle.
 - All the participants in the deadlock.
- What is the effect on any session that receives the “ORA-00060: deadlock detected while waiting for resource” error message?
 - The statement that it was executing is rolled back.
 - Its transaction is rolled back.
 - It is terminated.
 - It depends on the circumstances.
- What action must the database administrator take to resolve the deadlock?
 - Terminate the session that is blocking the others.
 - Terminate any one of the sessions participating in the deadlock.
 - Terminate all the sessions participating in the deadlock.
 - None of the above.
- What is the recommended action that a session should take when it receives an ORA-00060 error message?
 - Issue the ROLLBACK TO SAVEPOINT command and repeat the statement that it was executing when it received the ORA-00060 error message.
 - Repeat its entire transaction; there is no need to issue an explicit ROLLBACK command because

- Oracle has already rolled back the entire transaction.
- (c) Repeat the entire transaction; the session must first issue an explicit ROLLBACK command because Oracle has not rolled back the transaction.
7. What causes a deadlock in Oracle? Pick all that apply.
- Contention for the same rows of data by multiple sessions.
 - Physical database design issues such as unindexed foreign keys.
 - Architectural issues such as block-level locking.
8. How does Oracle detect that a deadlock has occurred?
- It checks for deadlocks whenever a session cannot acquire a lock.
 - It checks for deadlocks at regular intervals.
 - It checks for deadlocks whenever the wait time of a session has exceeded a certain threshold.
 - It checks for deadlocks whenever the number of waiting sessions has exceeded a certain threshold.
9. How can you obtain a list of all the rows locked by a transaction? Pick all that apply.
- By querying V\$LOCK in order to identify the locked rows.
 - By querying V\$LOCKED_OBJECT in order to identify the locked rows.
 - By querying DBA_LOCK in order to identify the locked rows.
 - None of the above.
10. A transaction issues a SAVEPOINT command and then locks a data row that is subsequently required by another transaction. What happens if the blocking transaction issues a ROLLBACK TO SAVEPOINT command?
- The blocking transaction reverses the changes made to the data row but does not release its lock on the row.
 - The blocking transaction reverses the changes made to the data row and releases its lock on the row; the blocked transaction is automatically unblocked.
 - The blocking transaction reverses the changes made to the data row and releases its lock on the row, but the blocked transaction remains blocked.

Answers

1. The correct answers are (c), (d), and (e). Deadlock is often confused with regular blocking situations, but a situation in which a blocking session becomes inactive without releasing its locks is not a deadlock. Neither is a situation in which a blocking session is itself locked by another session or an even more complicated logjam involving multiple sessions. The technical definition of a deadlock is a cycle in the “wait-for” diagram (e.g., A waiting for B, B waiting for C, C waiting for A). A situation in which two sessions are each trying to modify a row of data that has previously been modified by the other is obviously an example of a deadlock. Less obvious examples are situations in which a deadlock arises even though sessions do not require the same data rows or when a session

Es-Cue-El Challenge

Be It Known By These Presents that the great Wizard of Odds at Hogwash School of Es-Cue-El needs your help in solving the riddle of the ancient jade icosahedron found in the secret chamber of mystery. A great tournament has been organized and all practitioners of the secret art of Es-Cue-El have been invited to demonstrate their prowess.

The best entry will win the Wooden Pretzel award, instituted in honor of another great wizard, who famously observed that “some people can perform seeming miracles with straight Es-Cue-El, but the statements end up looking like pretzels created by somebody who is experimenting with hallucinogens.” As if that singular honor is not enough, a magnificent iPod Shuffle will also be bestowed on the champion.



Unsolvable Riddle

An ancient twenty-sided die (icosahedron) was discovered in the secret chamber of mystery at Hogwash School of Es-Cue-El. A mysterious symbol was inscribed on each face of the die. The great Wizard of Odds discovered that each symbol represented a number, and he also discovered that the die was biased—some numbers were more likely to be displayed than others when the die was used in a game of chance. The great wizard then recorded all this information in tabular fashion as described below.

```
SQL> describe die
```

Name	Null?	Type
FACE_ID	NOT NULL	NUMBER (2)
FACE_VALUE	NOT NULL	NUMBER (2)
PROBABILITY	NOT NULL	NUMBER (10, 10)

*The great wizard now implores you to create an Es-Cue-El spell that displays the probabilities of obtaining various sums when the die is thrown N times in succession.**

May the best wizard win!

** N is a “substitution variable” or “bind variable.”*

deadlocks with itself. Examples are provided in the references listed in the Further Reading section.

2. The correct answer is (d). The only listed situation that cannot result in a deadlock is a situation that does not involve any locking activity at all. Deadlocks can certainly occur if everybody is modifying the same table. Further, the fact that two sessions are owned by the same user does not prevent them from blocking each other and, when two sessions block each other, a deadlock can result. It is not, however, obvious that a session can deadlock with *itself*. Documented examples include parallel query processing, autonomous transactions, and recursive SQL.

3. The correct answer is (b). Only one of the participants in the deadlock receives an error message. This is the one that Oracle arbitrarily picks as a victim in order to resolve the deadlock. For example, if A is waiting for B, B is waiting for C, and C is waiting for A, then Oracle may arbitrarily pick B as the victim; B will then receive the error message “ORA-00060: deadlock detected while waiting for resource.” The other sessions do not receive an error message. Option (a) is not the best answer because all the participants in the deadlock are blocking someone else.

4. The correct answer is (a). Oracle performs statement-level rollback on a session that receives the “ORA-00060: deadlock detected while waiting for resource” error message.

5. The correct answer is (d). The database administrator does not have to resolve a deadlock because Oracle automatically does all that is necessary—let no one tell you otherwise! However, if deadlocks happen frequently and the applications do not handle them gracefully, the database administrator might well be drawn into the investigation of their cause.

6. The correct answer is (c). Oracle resolves a deadlock by performing a statement-level rollback operation on the victim but all the modifications made by the victim in previous statements, as well as all the locks previously acquired by it in the course of those statements, are still in effect.

Let's suppose that A is waiting for B, B is waiting for C, and C is waiting for A: a deadlock situation. If Oracle picks C as a victim and performs statement-level rollback, then B will still be unable to proceed. If C submits its last statement again, then the deadlock situation will be created again and C may be picked as the victim again. In an automated situation, the correct thing for C to do is to issue the ROLLBACK command before retrying its entire transaction.

In the following example, each session receiving a deadlock error retries without rolling back the rest of its work. Deadlocks happen repeatedly until one session finally decides to roll back its work and the other sessions are then able to complete their work. The session that rolled back then resubmits all its statements for processing and successfully completes its work.

```
SESSION A> UPDATE dummy SET status = 1 WHERE dummy = 1;
SESSION A> 1 row updated.

SESSION B> UPDATE dummy SET status = 1 WHERE dummy = 2;
SESSION B> 1 row updated.

SESSION C> UPDATE dummy SET status = 1 WHERE dummy = 3;
SESSION C> 1 row updated.

SESSION A> -- A tries to update the row locked by B
SESSION A> UPDATE dummy SET status = 1 WHERE dummy = 2;

SESSION B> -- B tries to update the row locked by C
SESSION B> UPDATE dummy SET status = 1 WHERE dummy = 3;

SESSION C> -- C tries to update the row locked by A
SESSION C> UPDATE dummy SET status = 1 WHERE dummy = 1;

SESSION A> ORA-00060: deadlock detected while waiting for resource

SESSION A> -- A retries without issuing a rollback command
SESSION A> UPDATE dummy SET status = 1 WHERE dummy = 2;

SESSION B> ORA-00060: deadlock detected while waiting for resource

SESSION B> -- B retries without issuing a ROLLBACK command
SESSION B> UPDATE dummy SET status = 1 WHERE dummy = 3;

SESSION C> ORA-00060: deadlock detected while waiting for resource

SESSION C> -- C retries without issuing a ROLLBACK command
SESSION C> UPDATE dummy SET status = 1 WHERE dummy = 1;

SESSION A> ORA-00060: deadlock detected while waiting for resource

SESSION A> -- A decides to issue a ROLLBACK
SESSION A> ROLLBACK;
SESSION A> Rollback complete.

SESSION C> -- C is no longer blocked by A
SESSION C> 1 row updated.
SESSION C> COMMIT;
SESSION C> Commit complete.

SESSION B> -- B is no longer blocked by C
SESSION B> 1 row updated.
SESSION B> COMMIT;
SESSION B> Commit complete.

SESSION A> -- A retries its entire transaction and is successful
SESSION A> UPDATE dummy SET status = 1 WHERE dummy = 1;
SESSION A> 1 row updated.
SESSION A> UPDATE dummy SET status = 1 WHERE dummy = 2;
SESSION A> 1 row updated.
SESSION A> COMMIT;
SESSION A> Commit complete
```

7. The correct answers are (a), (b), and (c). Oracle contributes to the misinformation about deadlocks by always using the following language in the trace file where the deadlock is recorded: “The following deadlock is not an ORACLE error. It is a deadlock due to user error in the design of an application or from issuing incorrect ad-hoc SQL.” In fact, deadlocks can result from no fault of the application.

In particular, physical database design issues such as un-

Oracle contributes to the misinformation about deadlocks by always using the following language in the trace file where the deadlock is recorded: “The following deadlock is not an ORACLE error. It is a deadlock due to user error in the design of an application or from issuing incorrect ad-hoc SQL.”

indexed foreign keys and indexing choices such as bitmap indexes can make the database prone to deadlock. Also, block-level locking can be encountered in an Oracle database in the guise of “ITL waits.” The details can be found in the references

“The database administrator does not have to resolve a deadlock because Oracle automatically does all that is necessary—let no one tell you otherwise!”

listed in the Further Reading section.¹

8. The correct answer is (b). It is computationally too expensive for Oracle to check whether a deadlock has occurred every time a transaction is blocked. Instead, Oracle invokes its deadlock detection algorithm at regular intervals.

9. The correct answer is (d). When a deadlock occurs, it would be helpful if Oracle would list all the locks held by the participants. However, row locks are not held in memory but

in the data blocks themselves. The advantage of this approach is that Oracle does not have to place artificial limits on the number of rows that can be locked by one transaction; the disadvantage is that it is not easy to produce a list of rows locked by a transaction.

10. The correct answer is (c). When the blocking transaction requested a ROLLBACK TO SAVEPOINT, it released the row lock acquired since the save point, but the blocked trans-

action is *still blocked*. This is because the blocked transaction is waiting for the “TX enqueue” (visible in V\$LOCK) which represents the blocking transaction, not any particular row lock.

Further Reading

Mark Bobak’s paper “Understanding and Interpreting Deadlocks or What to Do When You Encounter ORA-00060” is a good introduction to the topic. It is available for download at the Oak Table website www.oaktable.net/userFiles.jsp.

The following Metalink documents are also good sources of information on deadlocks: Metalink document 164661.1: ORA-60/Deadlocks Most Common Causes; Metalink document 62365.1: What to do with “ORA-60 Deadlock Detected” Errors. If you’d like to discuss this topic in a public forum, you can do so at oracleannoyances.blogspot.com. ▲

Iggy Fernandez is a database administrator for Verizon. Please send any comments or corrections to iggy_fernandez@hotmail.com.

Copyright © 2007, Iggy Fernandez

¹ My colleague Junho Whang points out that even ordinary indexing choices can increase the probability of deadlocks. If appropriate indexes are not available or the query optimizer does not use them appropriately, queries may hold locks longer than necessary, which increases the probability of deadlocks. The greater the locking activity, the longer the locks are held, and the wider the scope of the locks; the higher the chances are of a deadlock occurring. Batch jobs, long transactions, table locks, etc., increase the chances of deadlocks.

Database Management

Develop	Ensure the highest quality code is developed regardless of user skill set or location
Optimize	Maximize code performance and eliminate time-intensive manual tuning processes
Validate	Performance test your code for scalability before deploying to production

Can Toad® Turn Your Team Into Experts?

Sit back and relax. You have Toad’s development best practices on your side.

You could spend your day tracking down bad code.
Or you can take control with Toad® for Oracle.

Promote development best practices in your organization.
Read “Implementing Database Development Best Practices” at www.quest.com/relax




©2007 Quest Software, Inc. All rights reserved. Quest and Quest Software are trademarks or registered trademarks of Quest Software. All other brand or product names are trademarks or registered trademarks of their respective holders. www.quest.com/NoCOUG.

Simplify Identity Management



Enhance security, provision users quickly, and effectively protect your resources with IT Convergence’s Oracle Identity Management solutions.



IT Convergence
EXPERIENCE the power of results

Contact us today: www.itconvergence.com

A Matter of Life or Death!

by Chris Lawson



Performance tuning requires creativity and innovation. It's not at all just a routine, drab job—it requires versatility and stepping into different roles. That's why I like it so much.

Those who read this column know that I call the overall performance tuning process “Physician to Magician.” This means that we play a different role at different times. Sometimes the analyst needs to gather facts, like a detective. At other times we simply listen to our patient, like a kindly physician. Later, when we need to synthesize a solution, we are like an artist, creating a work of art that our customer will appreciate.

Now, I admit that not all performance tuning is exciting. There are plenty of cases that are solved by adding an index, fixing the statistics, and so on. These cases require more scientific competence than creativity. In general, however, performance tuning is not a mechanical process of turning the crank to spit out a solution to a performance problem. Designing a fix that meets your customer's needs is often not so trivial.

A Misunderstanding of the Process

Those not truly experienced in tuning tend to trivialize the process, frequently using trite phrases such as “add missing indexes” or “increase buffer cache.” This misunderstanding helps explain why relatively few DBAs become really good at performance tuning—their view of the task is simply too shortsighted. If your solution set comprises only trivial solutions such as new indexes, you're not likely to excel in this field.

Here's a sampling of various root causes of performance logjams:

- Overly strict business requirement
- Optimizer incorrectly handling new Oracle functionality
- Optimizer predicting one execution plan, but really running another

“Those not truly experienced in tuning tend to trivialize the process, frequently using trite phrases such as ‘add missing indexes’ or ‘increase buffer cache.’ In general, however, performance tuning is not a mechanical process of turning the crank to spit out a solution to a performance problem.”

- Program calling custom function millions of times

Note that in each case above, a good solution would not be forthcoming by a “turn the crank” approach. The solution for each case required understanding of the particular environment. Sometimes, however, even with all facts in hand, a solution is not apparent or is very expensive for our client. Allow me to suggest a novel alternative—one that I have found very helpful.

A Matter of Life or Death!

Here's a tactic I use when I'm faced with an apparently unsolvable performance problem. I imagine that the problem is so critical that people will actually die if I don't resolve the difficulty. I imagine that the lives of people are actually in my hands.

Yes, I admit this is a bizarre idea, but it actually works. If you can temporarily pretend that there are no limits, you'll be amazed at the variety of solutions you can invent. Here's an actual case where I “saved” a life.

Fix Performance or Someone Dies!

A major retailer had a problem with a long-running batch job. Although we were able to identify several bottlenecks, the fix required a program redesign; however, this was not acceptable. Along with QA testing and all the paperwork, the ensuing delay would be at least several weeks, if not a month or more.

In cases like this, it's tempting to throw the problem back at the developers. Of course, this isn't too helpful to the client. So just for fun, I pretended that my performance problem was life threatening. I asked myself, “Is it really true that I can do nothing to speed up the job? Even if someone dies I can't do anything?”

The Performance Boost

Given my new motivation, all limits were off. I considered all kinds of crazy ideas—server changes, init.ora changes, disk

“You don’t really know what you can do unless you’re pushed really hard. So the next time you face a seemingly impossible problem, don’t be too quick to throw in the towel. Instead, pretend, just for a minute, that you absolutely have to fix it—or else!”

changes, etc. Finally, I thought of one very simple, but admittedly weaselly idea. Since the bottleneck was mostly due to disk I/O, any improvement in disk access would directly affect the problem job. Since we knew the exact SQL, we could pre-run SQL that duplicated the “real” SQL that would shortly follow. That is, we would pre-cache many of the blocks that would shortly be needed. Caching would take place within the SAN unit, as well as at the Oracle database cache. The entire performance improvement would take place with absolutely no change to the batch program!

To speed up the pre-caching program, we used multi-threading to run about 20 simultaneous database sessions (we had the spare CPU capacity). This was really simple and made use of Oracle’s parallel query option. (For details on how to cause Oracle parallelism to perform multi-threading, see OracleMagician.com/mag/magic12.pdf.)

We successfully ran the “boost” program just prior to the problem batch job, reducing the runtime by several hours. This satisfied the short-term needs of the client. (Incidentally, I have used this “boost” idea several other times, where an urgent, short-term fix was required.)

Wrap-Up

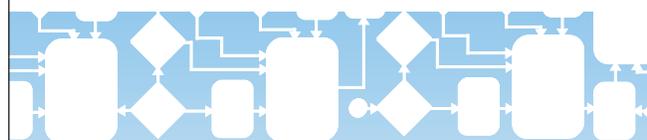
Okay, I admit my “life and death” scheme is an unusual way to approach performance tuning. While our fix would certainly not be considered a long-term solution, it did give the client some breathing room, so that they could fix the program design. That’s all the customer needed.

The concept discussed here is really more about the art of human creativity and motivation than about databases. You could argue that it’s really just another way to think outside the box. Here’s the point: You don’t really know what you can do unless you’re pushed really hard. So the next time you face a seemingly impossible problem, don’t be too quick to throw in the towel. Instead, pretend, just for a minute, that you absolutely *have* to fix it—or else! ▲

Chris Lawson is an Oracle DBA consultant in the San Francisco Bay Area, where he specializes in performance tuning of data warehouse and financial applications. He is a frequent speaker at NoCOUG, and has written for a number of publications such as Oracle Internals, Exploring Oracle, SELECT, Oracle Informant, and Intelligent Enterprise. Chris has held a variety of positions in the IT field—ranging from systems engineer to department manager—and is an instructor for the University of Phoenix. He can be contacted via www.oraclemagician.com.

Copyright © 2007, Chris Lawson

Announcing Embarcadero® EA/Studio™



Join your processes and data
with EA/Studio.

- Conceptual modeling
- Real-world process modeling
- Speak the same language as business users

Visit www.embarcadero.com/eastudio
to download a free trial.



ENTERPRISE STORAGE SOLUTIONS FOR ORACLE.
SCALABLE STORAGE INFRASTRUCTURE.
SCALED DOWN STRESS.

PICK ANY THREE.

SETTLE FOR EVERYTHING WITH NETAPP.

Find out why we’re trusted by the world’s
greatest companies that run Oracle.

Visit netapp.com/bizapps



© 2007 Network Appliance, Inc. All rights reserved. Specifications subject to change without notice. NetApp and the Network Appliance logo are registered trademarks and Network Appliance is a trademark of Network Appliance, Inc. in the U.S. and other countries. All other brands or products are trademarks or registered trademarks of their respective holders and should be treated as such.

Bag o' Tricks

by Danny Chow



Danny Chow

There are times when your users complain of a “hung” database and the DBA is under the spotlight almost instantly. You log in, run through a textbook-standard checking routine on the database vital signs, error logs, network, and server host. You find no capacity problem, no error logged, and, while you are constructing a query to identify blocking locks, your world is on fire. Not a good scenario.

There are different tools available to identify locks. However, when someone important in your company is breathing down your neck, a quick script will save the day, and I am going to present one of the scripts I use the most. The queries presented here will identify the session causing blocking locks, the object being locked, and the SQL statement that acquired the lock. The script was tested on both Oracle 9i and Oracle 10g.

```
-- blocking_lock.sql

SET pagesize 999
SET linesize 132
SET heading on
SET feed off
SET echo off

COL sid_serial heading "Sid,Serial" format a10
COL user_name heading "User" format a10
COL pid heading "Pid" format a5
COL locked_obj heading "Locked Object" format a22 WRAP
COL lock_held heading "Lock Held" format a23 WRAP
COL blocking heading "Block?" for a6
COL last_convert heading "Time(s)" format 99999
COL hash_value heading "Hash Value"
COL sql_text heading "Sql Text" format a60 WRAP

BREAK on locked_obj skip 0 on user_name skip 0

SELECT /*+ORDERED */
       b.owner
       || ','
       || b.object_name
       || ','
       || INITCAP (b.object_type)
       locked_obj,
       a.oracle_username user_name,
       d.process pid,
       d.SID || ',' || d.serial#
       sid_serial,
       c.lock_type
       || '-'
       || c.mode_held lock_held,
       DECODE (c.blocking_others,
              'Not Blocking', NULL,
              'Blocking', 'Yes',
              c.blocking_others
              ) blocking,
       last_convert
FROM   v$locked_object a,
       dba_objects b,
       dba_lock c,
       v$session d
WHERE  a.object_id = b.object_id
AND    a.session_id = c.session_id
AND    ( c.mode_held = 'None'
OR      c.blocking_others !=
        'Not Blocking'
        )
AND    c.session_id = d.SID
ORDER BY locked_obj,
         blocking,
         last_convert DESC,
         user_name;

BREAK on sid_serial skip 0 on hash_value skip 0

SELECT /*+ORDERED */
       c.SID || ',' || c.serial#
       sid_serial,
       DECODE (c.sql_hash_value,
              0, c.prev_hash_value,
              c.sql_hash_value
              ) hash_value,
       d.sql_text sql_text
FROM   v$lock b,
       v$session c,
       v$sqltext d
WHERE  b.BLOCK > 0
AND    b.SID = c.SID
AND    d.hash_value =
       DECODE (c.sql_hash_value,
              0, c.prev_hash_value,
              c.sql_hash_value
              )
ORDER BY sid_serial, hash_value, piece;
```

Here is an example of the script in action.

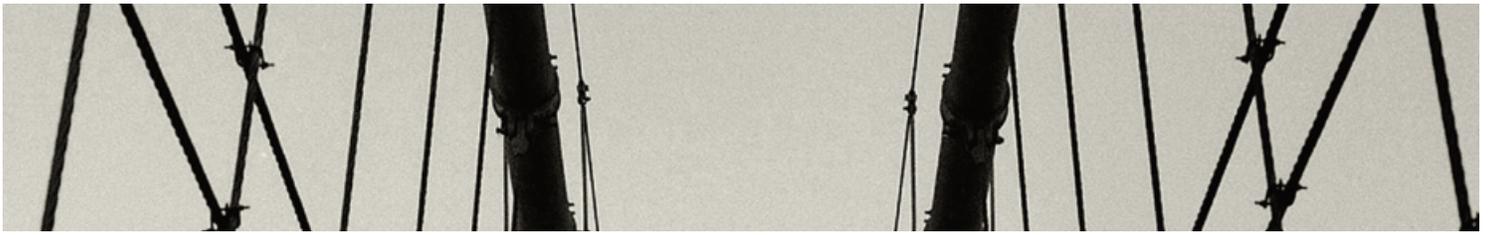
```
SQL> @blocking_lock
```

Locked Object	User	Pid	Sid,Serial	Lock Held	Block?	Time(s)
TEST.TEST_TABLE Table	TEST	8609	42,30975	Transaction - Exclusive	Yes	9
		12267	19,26286	Transaction - None		2

Sid,Serial	Hash Value	Sql Text
42,30975	1080166690	update test_table set flag='Y' where flag='N'

Danny Chow is an independent Oracle consultant with many years of DBA and development experience, and holds certifications in Oracle 7, 8, 8i, 9i, and 10g. His email address is dannychow@earthlink.net.

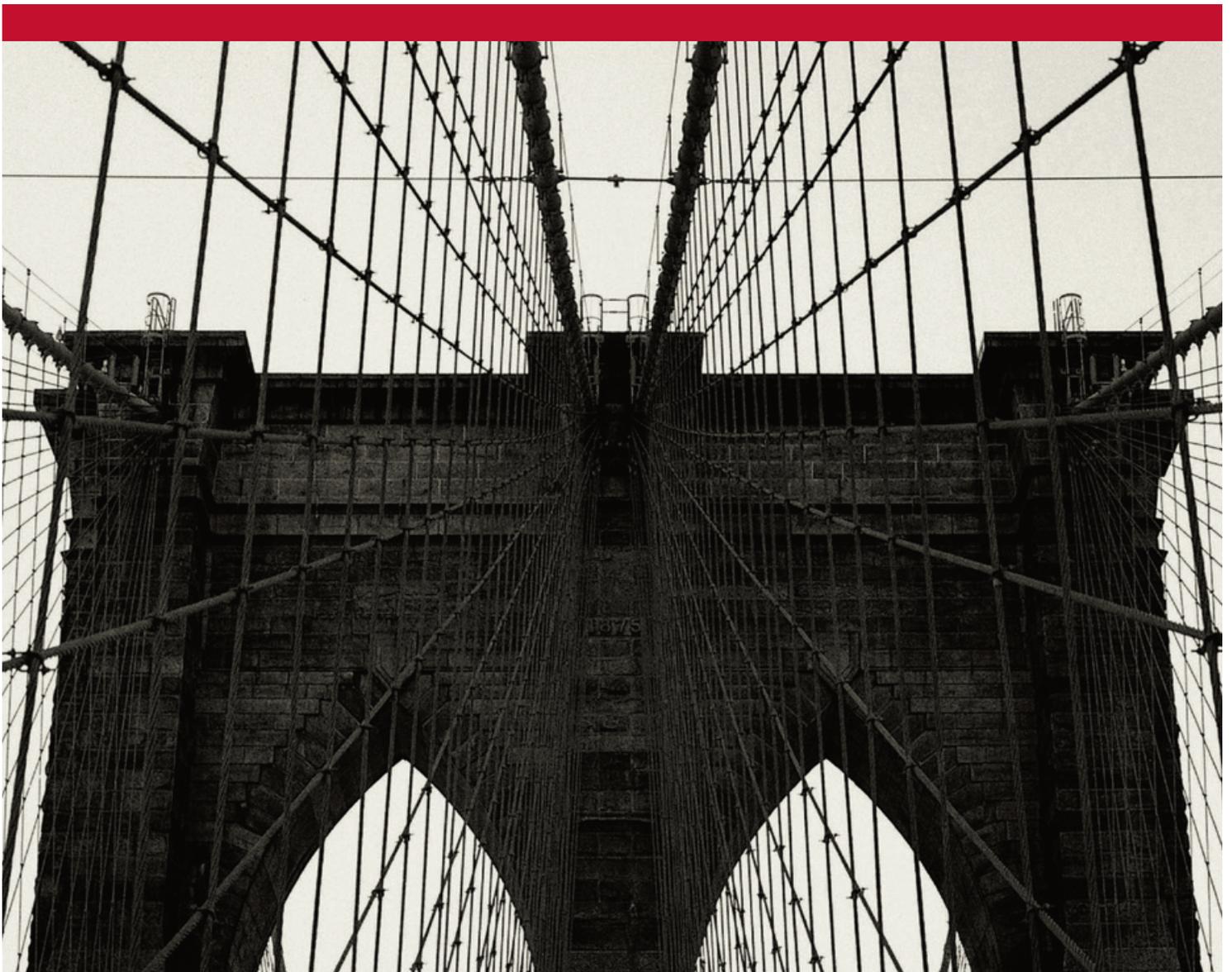
Copyright © 2007, Danny Chow



Enterprise data. Aligned.

You dedicate huge resources to enterprise applications like Oracle,[®] PeopleSoft,[®] JD Edwards,[®] Siebel.[®] And every year they grow in size and complexity — increasing maintenance challenges. Now you can take back control with Princeton Softech Optim.[™] Optim gives you the ability to align application data management to performance goals. Optimizing results, mitigating risk and controlling the cost of every IT investment — servers, software, storage, networks and people. Is this heaven? No, it's the power of Enterprise Data Management.

Learn more at princetonsoftech.com.



NoCOUG Fall Conference

Session Descriptions

For more detailed descriptions and up-to-date information, see www.nocoug.org.

—Keynote—

Empiricism vs. Guessing Among Oracle Professionals

Jeremiah Wilton, *ORA-600 Consulting* 9:30–10:15

As Oracle has provided ever-richer facilities for solving Oracle problems using facts and figures, the reliance of so many DBAs and developers on (sometimes educated) guesses to solve problems is shocking and inexcusable to many whose approach is strictly science. Conversely, proponents of (some) guessing see the strict empiricists as excessively doctrinal, and ask why, if they can solve problems faster by checking a few usual suspects, their approach is any less valid than solving the problem using hard data. In this polarizing debate, both sides have a point, and have much to learn from each other.

Jeremiah Wilton is principal consultant and owner of ORA-600 Consulting. Before becoming independent, he spent seven years at Amazon.com, initially as their first database administrator in 1997. Using expertise he gained during Amazon.com's period of exponential growth, he now specializes in scalability, high availability, stability, and complex recoveries. Jeremiah has presented at numerous conferences and user group meetings, and is the author of a variety of technical white papers and articles. His publications are available at www.ora-600.net.

—Auditorium—

RAC Be Nimble, RAC Be Quick

Bert Scalzo, *Quest Software* 11:00–12:00

Often relying on default operating system setup and database configuration parameters can lead to very poor RAC performance right out of the box—a symptom that I jokingly refer to as “RAC in the Box” failure. But through simple application analysis and tuning corresponding network, operating system, and database parameter settings to match, DBAs can often get past the initial RAC performance issues. Furthermore, the technique serves as a nice and simple “best practices” checklist for any RAC setup.

Bert Scalzo is a database architect for Quest Software and a member of the TOAD team. He has worked extensively with TOAD's developers and designed many of its features—including the DBA module, Code Road Map, and Code Xpert. Mr. Scalzo has worked with Oracle databases for well over two decades, starting with version 4. Mr. Scalzo is an accomplished speaker and has presented at numerous Oracle conferences and user groups including OOW, ODTUG, IOUGA, and OAUG. He has written four books: Oracle DBA Guide to Data Warehousing and Star Schemas, TOAD Handbook, TOAD Pocket Reference (2nd Edition), and Database Benchmarking: Practical Methods for Oracle 10g & SQL Server 2005. Mr. Scalzo can be reached via email at bert.scalzo@quest.com or bert.scalzo@yahoo.com.

Automatic Hang Detection in Oracle 11g

Jeremiah Wilton, *ORA-600 Consulting* 1:00–2:00

When an Oracle database instance hangs, DBAs have often been left with few options other than killing and restarting the instance. The root cause of a hanging issue often goes undiscovered even after several occurrences. Oracle 11g introduces Automatic Hang Detection to produce useful diagnostics on the first occurrence of a hang. In this PowerPoint-free session, we will make an Oracle 11g database instance hang, watch hang detection do its job, and examine the diagnostic information produced.

Jeremiah Wilton is principal consultant and owner of ORA-600 Consulting. Before becoming independent, he spent seven years at Amazon.com, initially as their first database administrator in 1997. Using expertise he gained during Amazon.com's period of exponential growth, he now specializes in scalability, high availability, stability, and complex recoveries. Jeremiah has presented at numerous conferences and user group meetings, and is the author of a variety of technical white papers and articles. His publications are available at www.ora-600.net.

Oracle 11g New Features: Live in SQL*Plus

Daniel Morgan, *University of Washington* 2:30–3:30

Oracle 11g offers a variety of great new capabilities that extend the value of your investment in the Oracle database. This presentation, live in SQL*Plus and reading a single PowerPoint slide, will feature demonstrations of Virtual Columns, Invisible Indexes, Flashback Archive, Native Compilation, the new table trigger Follows clause, and the Result Cache. Note: This will be a PowerPoint-Free Zone!

Daniel Morgan has over 36 years experience in information systems architecture, administration, development, technical publishing, and education. He developed the curriculum for the University of Washington's Oracle Education program and has taught the program since its inception. He has also acted as an advisor to the University of California at Berkeley and Emory University. Daniel has been a frequent guest lecturer at technology events and has spoken to audiences at Oracle Open World; user groups in the U.S. and Canada; and Argonne, Battelle, and Lawrence Livermore National Laboratories. He serves as the education chair of the Puget Sound Oracle User Group.

Simulating Oracle I/O Workloads to Accurately Configure Storage

James Koopmann, *LSI Corporation* 4:00–5:00

Using Oracle's ORION workload tool, we can effectively develop a workload that mimics and stresses a storage array in the same manner as a planned application with an Oracle backend database. Only after benchmarking potential storage can we gain a finer understanding of the performance capabilities of our storage and be confident in its ultimate deployment. This presentation discusses storage concepts such as RAID level, Host Bus Adapters, Controllers, and Oracle's ORION workload tool. Then the attendee will be walked through a variety of benchmark tests and results that show how to run the ORION tool and make decisions on how to configure storage for an Oracle database. This presentation will show the results of various workloads using RAID 0, 1, 10, and 5 on a 30 TB storage system. So come see how your favorite RAID level performed.

James F. Koopmann is dedicated to providing technical advantage and guidance to companies within information technology. Over the years James has worked with a variety of database-centric software and tools vendors as a strategist, architect, DBA, and performance expert. He is an accomplished author with technical writings appearing in publications such as Oracle Magazine, Oracle Professional, and Exploring Oracle, and across the Web in many portals and newsletters.

—Tassajara—

Oracle Database 11g: Real Application Testing

John Sheaffer, *Oracle Corporation* 11:00–12:00

In addition to meeting user Service Level Agreements, every organization has to upgrade their software, rollout database changes, and apply patches and other system changes on a regular basis. During this session we'll discuss how Oracle Database 11g addresses these challenges with the new Real Application Testing option and then perform a live demonstration of the 11g Database Replay feature.

John Sheaffer is a principal platform solution specialist at Oracle Corporation.

Performance Testing Fundamentals

Karen Morton, *Hotsos Enterprises* 1:00–2:00

Editor's Pick

How do you know if your SQL and PL/SQL code is performance optimal? Can you efficiently and adequately determine how well your applications perform in your test environment before you place your code into production? Are you considering all the “right” things when you test for performance? This presentation will focus on the fundamentals that should be the basis for code testing, and will review a simple set of SQL*Plus scripts that you can use to conduct complete, effective, and easy performance tests.

Karen Morton is the lead designer and developer of the Hotsos SQL

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 memberships. If your company is able to offer sponsorship at any level, please contact NoCOUG's president, Lisa Loper, at lloper@dbspecialists.com. ▲

Long-term event sponsorship:

LOCKHEED MARTIN

CHEVRON

ORACLE CORP.

PG&E

Thank you! Year 2007 Gold Vendors:

- ▶ BEZ Systems
- ▶ Confio Software
- ▶ Database Specialists, Inc.
- ▶ Embarcadero Technologies
- ▶ IT Convergence
- ▶ Network Appliance
- ▶ Princeton Softech
- ▶ Quest Software
- ▶ Roundstone Systems

For information about our Gold Vendor Program, contact the NoCOUG vendor coordinator via email at:
vendor_coordinator@nocoug.org



TREASURER'S REPORT

Jen Hong, Treasurer

Beginning Balance

July 1, 2007 \$ 53,261.78

Revenue

Membership Dues	4,937.50	
Meeting Fees	920.00	
Vendor Receipts	4,500.00	
Advertising	550.00	
Training Day	13,295.00	
Sponsorship	3,000.00	
Interest	41.34	
Total Revenue		\$ 27,269.52

Expenses

Regional Meeting	6,595.06	
Journal	7,224.87	
Membership	121.30	
Administration	41.00	
Website	30.00	
Board Meeting	575.72	
Marketing	-	
Insurance	-	
Vendors	59.20	
P.O. Box	-	
Training Day	8,401.55	
Miscellaneous	100.00	
Total Expenses		\$ 23,148.70

Ending Balance

September 30, 2007 \$ 57,382.60

Optimization curriculum. She has over 19 years of experience in relational database management systems, including 13 years of experience working with Oracle. She is the co-author of several Oracle training course books and has taught hundreds of individuals in many Oracle topic areas. Prior to joining Hotsos, she owned her own Oracle training and consulting business. The business focused on mentoring individuals and small groups to improve database administration, manageability, and performance. She has presented at several user group conferences, including last year's Hotsos Symposium.

Made from the Same Mold: Templating Approaches for Fusion Applications

Peter Koletzke, Quovera 2:30–3:30

Users will more easily understand their application and will therefore be more productive if all pages of their web application are designed to look and act the same. The requirement for a common look and feel has traditionally been met by the use of templates. However, the methods for implementing template systems change with each new technology. This presentation discusses how to work with templates and other common look-and-feel options in JDeveloper 10.1.3. It explains and provides code examples for template systems using ADF Faces and JSP tags, as well as additional Java frameworks. It also describes the Oracle Browser Look and Feel (BLAF), which is used for the E-Business Suite. The presentation also describes the ADF Faces skinning feature, as well as how the JHeadstart plug-in to JDeveloper uses templates.

Peter Koletzke is a technical director and principal instructor for the Enterprise e-Commerce Solutions practice at Quovera, in Mountain View, California, and has over 23 years of industry experience. Peter has presented at various Oracle users group conferences more than 200 times and is an Oracle Certified Master, Oracle Ace Director, and coauthor of the Oracle Press Books: Oracle JDeveloper 10g for Forms & PL/SQL Developers (with Duncan Mills), Oracle JDeveloper 10g Handbook and Oracle9i JDeveloper Handbook (with Dr. Paul Dorsey and Avrom Roy-Faderman), Oracle JDeveloper 3 Handbook, Oracle Developer Advanced Forms and Reports, Oracle Designer Handbook (2nd Edition), and Oracle Designer/2000 Handbook (all with Dr. Paul Dorsey).

Just Get It Written: Deploying Applications to OC4J Using JDeveloper and EM

Peter Koletzke, Quovera 4:00–5:00

Deploying a J2EE web application consists of wrapping all application files into a single Java archive file (the EAR, enterprise application archive) and then copying this file to the application server. Although the Oracle Containers for J2EE (OC4J) service of the Oracle Application Server is a standard J2EE runtime process, it has its own deployment utilities and procedures. This presentation explores the process deploying to an Oracle Application Server OC4J container. It outlines JDeveloper utilities and profiles that help you create an EAR file with all required libraries. The presentation explains how to deploy the application using JDeveloper. It also discusses the benefits of and techniques for testing deployment on a local stand-alone OC4J server. In addition, the presentation describes the deployment services of the Application Server's Enterprise Manager Console (EM). The presentation also discusses deployment plans and common deployment process and runtime errors.

Peter Koletzke is a technical director and principal instructor for the Enterprise e-Commerce Solutions practice at Quovera, in Mountain View, California, and has over 23 years of industry experience. Peter has presented at various Oracle users group conferences more than 200 times and is an Oracle Certified Master, Oracle Ace Director, and coauthor of the Oracle Press Books: Oracle JDeveloper 10g for Forms & PL/SQL Developers (with Duncan Mills), Oracle JDeveloper 10g Handbook and Oracle9i JDeveloper Handbook (with Dr. Paul Dorsey and Avrom Roy-Faderman), Oracle JDeveloper 3 Handbook, Oracle Developer Advanced Forms and Reports, Oracle Designer Handbook (2nd Edition), and Oracle Designer/2000 Handbook (all with Dr. Paul Dorsey).

—Diablo—

The Oracle Fusion Development Platform: Simplify Your Application

Shay Shmeltzer, Oracle Corporation 11:00–12:00

This session discusses the development platform for the Oracle Fusion Applications focusing on Oracle JDeveloper and Oracle ADF, as well as the integration with BPEL and WebCenter. Come see the tools in action and learn about the development experience. You'll also learn how you can leverage these tools to speed up your development today.

Shay Shmeltzer is a principal product manager in the Oracle JDeveloper group. He has been working with Oracle's development tools since 1990, using everything from Forms 2.0 and Oracle 5.0 to the latest things you can get from OTN. He has been with Oracle Corporation since 1993 in various roles spanning development, sales, consulting, and even marketing.

Moving Oracle Databases across Platforms without Export/Import

Roger Schrag, Database Specialists 1:00–2:00

Moving an existing Oracle database to a new server that uses a different operating system or processor architecture is more complicated than just copying data files from the old server to the new one. Prior to Oracle 10g, the only supported way to move a database across platforms was to export the data from the existing database and import it into a new database on the new server. This works pretty well if your database is small, but it can require an unreasonable amount of down time if your database is large. Beginning in Oracle 10g, the transportable tablespace facility has been enhanced in a way that makes it possible to move large databases across platforms much more quickly and simply than the export/import method. So if you've got a few terabytes of data in a database on a Solaris server and you want to move it to a Linux server, for example, there is now a much faster way to do it. In this presentation we will begin by briefly discussing Oracle's cross-platform support for transportable tablespaces. Then we'll walk through a real-life example of moving an Oracle database from a Sun E450 to a Dell server running Red Hat Enterprise Linux. We will discuss the steps involved and the pitfalls, limitations, and things to keep in mind when planning a database server platform change.

Roger Schrag is CTO of Database Specialists, Inc., which he founded in 1995 and led from its early days to its current expansion of remote database administration services and onsite consulting. Roger is well known among the Oracle user community as an authority on Oracle. He is a popular author on Oracle subjects, and he is frequently rated a top 10 speaker at the national and local Oracle conferences he attends. He has served on the "Oracles of Oracle" expert panels at Oracle OpenWorld events and has also been selected as an instructor for the "Oracle Masters" Class series of the Independent Oracle Users Group (IOUG).

Data Guard Standby Features—Physical vs. Logical Standby

Brian Hitchcock, Sun Microsystems 2:30–3:30

Data Guard maintains a standby database by sending and applying the archived redo logs from the primary database to the standby database. This is a simple concept and can be put to good use in many situations. Within the Data Guard product you have the choice of Physical or Logical Standby. This presentation covers the features of both.

Brian Hitchcock has worked at Sun Microsystems in Newark, California, for the past 11 years. He is a member of a DBA team that supports 2400+ databases for many different applications at Sun. He frequently handles issues involving tuning, character sets, and Oracle applications. Other interests include Formula One racing, finishing his second Tiffany Wisteria lamp, Springbok puzzles, Märklin model trains, Corel Painter 8, and watching TV (TiVo rules!).

Oracle Data Guard Logical Standby Support Issues

Brian Hitchcock, Sun Microsystems 4:00–5:00

The Logical Standby offers many features that are not available with the Physical Standby. However, these features, while being very attractive, also bring with them numerous support issues. This presentation covers the real-world support issues that should be reviewed before making the decision to use Logical Standby.

Brian Hitchcock has worked at Sun Microsystems in Newark, California, for the past 11 years. He is a member of a DBA team that supports 2400+ databases for many different applications at Sun. He frequently handles issues involving tuning, character sets, and Oracle applications. Other interests include Formula One racing, finishing his second Tiffany Wisteria lamp, Springbok puzzles, Märklin model trains, Corel Painter 8, and watching TV (TiVo rules!). ▲



Sometimes
what's slowing
you down is obvious.

Usually, it's harder to pinpoint.

**Amazing what you can accomplish once you have
the information you need.**

When the source of a database-driven application slowdown isn't immediately obvious, try a tool that can get you up to speed. One that pinpoints database bottlenecks and calculates application wait time *at each step*. Confio lets you unravel slowdowns at the database level with no installed agents. And solving problems where they exist costs a *tenth* of working around it by adding new server CPU's. Now that's a vision that can take you places.

A smarter solution makes everyone look brilliant.

CONFIO
SOFTWARE

Download your FREE trial of Confio Ignite™ at www.confio.com.

Oracle | DB2 | SQL Server | Sybase | J2EE

NoCOUG Fall Conference Schedule

October 25, 2007, at CarrAmerica Conference Center, Pleasanton, CA

Please visit www.nocoug.org for updates and directions, and to submit your RSVP.

Cost: \$40 admission fee for nonmembers. Members free. Includes lunch voucher.

8:00 a.m.–9:00	Registration and Continental Breakfast—Refreshments served
9:00–9:30	Welcome —Lisa Loper, NoCOUG president
9:30–10:15	Keynote: <i>Empiricism vs. Guessing Among Oracle Professionals</i> —Jeremiah Wilton, ORA-600 Consulting
10:15–11:00	Break
11:00–12:00	Parallel Sessions #1 Auditorium: <i>RAC Be Nimble, RAC Be Quick</i> —Bert Scalzo, Quest Software Tassajara: <i>Oracle Database 11g: Real Application Testing</i> —John Sheaffer, Oracle Corporation Diablo: <i>The Oracle Fusion Development Platform: Simplify Your Application</i> —Shay Shmeltzer, Oracle Corporation
12:00–1:00 p.m.	Lunch
1:00–2:00	Parallel Sessions #2 Auditorium: <i>Automatic Hang Detection in Oracle 11g</i> —Jeremiah Wilton, ORA-600 Consulting Tassajara: <i>Performance Testing Fundamentals</i> —Karen Morton, Hotsos Enterprises Diablo: <i>Moving Oracle Databases across Platforms without Export/Import</i> —Roger Schrag, Database Specialists
2:00–2:30	Break and Refreshments
2:30–3:30	Parallel Sessions #3 Auditorium: <i>Oracle 11g New Features: Live in SQL*Plus</i> —Daniel Morgan, University of Washington Tassajara: <i>Made from the Same Mold: Templating Approaches for Fusion Applications</i> —Peter Koletzke, Quovera Diablo: <i>Oracle Data Guard Standby Features—Physical vs. Logical Standby</i> —Brian Hitchcock, Sun Microsystems
3:30–4:00	Raffle
4:00–5:00	Parallel Sessions #4 Auditorium: <i>Simulating Oracle I/O Workloads to Accurately Configure Storage</i> —James Koopmann, LSI Corporation Tassajara: <i>Just Get It Written: Deploying Applications to OCAJ Using JDeveloper and EM</i> —Peter Koletzke, Quovera Diablo: <i>Oracle Data Guard Logical Standby Support Issues</i> —Brian Hitchcock, Sun Microsystems
5:00–	NoCOUG Networking and Happy Hour at Faz, 5121 Hopyard Road, Pleasanton

Session descriptions
appear on page 24.

RSVP online at www.nocoug.org/rsvp.html

NoCOUG

P.O. Box 3282
Danville, CA 94526

RETURN SERVICE REQUESTED