



Official Publication of the Northern California Oracle Users Group

NoCOUG

J O U R N A L

Vol. 23, No. 2 · MAY 2009

\$15

Let Knowledge Spring Forth at NoCOUG

Fresh Perspectives

An interview with Karen Morton.

See page 4.

ASM Test Environment

An excerpt from a new book.

See page 9.

SQL Corner

Graphical query execution plans.

See page 16.

Much more inside . . .

Fresh Perspectives

from Karen Morton



Karen Morton

For over 20 years, Karen has worked in information technology, starting out as a mainframe programmer and then growing and changing with the times to do development, DBA, and data architect work. She has been using Oracle since the early 1990s and began teaching others how to use Oracle over a decade ago. She has authored several Oracle training course books, including leading the development of the Hotsos and Method R SQL Optimization curriculum. Karen is an Oracle Ace and a member of the Oak Table Network. She blogs at karenmorton.blogspot.com.



Why Oracle? Enterprise Edition with Partitioning now costs \$59,000 per CPU. The annual support fee is 22% of the purchase price. MySQL has partitioning and the price is right: it's free.

Oracle has led the industry for the past two decades and I don't see that changing. My world revolves around Oracle, so I can't really speak to the differences with MySQL as far as functionality, reliability, scalability, and performance. Yes, Oracle is expensive. Is it worth the cost to your business? Only your business can decide that. I hear the old adage "you get what you pay for" floating through my mind as I write this. Honestly, I think that the cost of something (i.e., higher cost equates to better) as a measure of its ability to satisfy your business needs is less true today than it has ever been. Open source products have made big gains in the market and they do that for a reason: they meet the demands of the businesses that implement them, and they have an attractive TCO (total cost of ownership).

While I think Oracle is the winner for the large enterprise environment, I think it is only prudent that each business evaluate their needs and their cost limitations and choose what best fits for them. I also think that if open source vendors continue to improve, Oracle may one day have to seriously consider pricing model changes in order to maintain their market share. As the gap between free or very low-cost open source databases and Oracle closes, cost may win more often than not.



A company I know has been using Oracle 9i for a long time. Is it time for them to upgrade to Oracle Database 11g? What are the risks? What are the rewards? Their database is very stable and performance is rock solid. They don't have a support contract

anymore, don't seem to need one, and can't afford to pay for one. The application was written in-house.

Well, I think you answered the question for me by saying they don't have a support contract, don't seem to need one and can't afford one even if they did. In this case, it sounds like a case of "if it ain't broke, don't fix it." They have stable performance using what they've got and no apparent need for any features 11g might provide. Until they can justify a need, then they're doing the best they can for their business by staying with 9i. The risks of upgrading, particularly the possibility of execution plan instability that might cause performance issues, isn't worth it when they are happy and rock solid where they are.



We spend a lot of time tuning databases. Why does it have to be that way? Is Oracle difficult to tune? Are we building the databases or applications wrong?

The whole idea that tuning is difficult is a very pervasive belief. Many people think it is more magic than science and believe if you don't know how to tweak some 200-plus instance parameters in just the right way, you're doomed to failure (which translates to having to hire an expensive consultant to tune things for you). In a 2003 Oracle white paper entitled "The Self-Managing Database: Automatic Performance Diagnosis," authors Graham Wood and Kyle Hailey state that one of the main reasons the many advisors were developed starting in 10g was to prove that tuning was in large part about following a logical, methodical approach. And since computers were pretty good at doing just that, the goal of the advisors was to simplify the process of diagnosing problems and make it seem less difficult.

So, we've now got access to a set of advisors that can help diagnose problems quicker and easier than ever before. However, I think that when problems do occur, the best defense is a good offense. In other words, we (the DBAs and developers who manage the database and write applications) must have the knowledge and skills to know how to quickly and efficiently diagnose problems when they occur. Gaining the knowledge and skills will also help ensure that we don't make many of the big performance mistakes to start with. I really believe that knowledge is power and that tuning doesn't have to be as hard as it is often made out to be.

(continued on page 7)

(continued from page 4)



In his latest book, Oracle on VMware, Dr. Bert Scalzo makes a case for “solving” performance problems with hardware upgrades. What is your opinion about this approach?

If you have performance problems that have been proven, through careful analysis, to have their root cause in substandard hardware, then the solution may be to upgrade. The investment made to make sure personnel are adequately trained to properly diagnose and repair performance problems can benefit your company in the long run much more so than taking what appears to be a quick fix by throwing hardware at the problem in the short run.

I think that when personnel are properly trained and knowledgeable, they will not only be able to quickly, efficiently, and cost effectively find and fix problems when they arise, but they will not make the mistakes that cause the problems in the first place.

I believe the truth is that if hardware isn't your real problem, you'll eventually have no other choice than to address the actual root cause, because hardware upgrades can only be a Band-aid on the wound for so long. I've seen too many people try this approach only to pay hundreds of thousands of dollars for hardware they don't really even need and *still* have performance problems. When they finally do address the real root cause and fix it, they often find that they could have bypassed the entire expenditure on hardware and saved more time and money if they'd just done it right in the first place.



I make a very good living adding hints to Oracle queries to improve performance. Does the optimizer really need a hint or is it something the developer or DBA is doing wrong in the first place?

I think hints are phenomenal testing tools and are *sometimes* the only viable avenue for getting a query to perform as you desire. But I also think that if you understand the Oracle optimizer and why it makes choices like it does, then you can often find a way to rewrite queries to get the results you want without a hint.

So no, I don't think the optimizer *needs* a hint as a rule. The bottom line is that you have to do what is necessary to get the performance you need. My first goal would be to do everything I could with my SQL to give the optimizer the best chance possible to get it right without a hint. If I still can't get the execution plan I want/need, then I'll use a hint. However, I'll document what I believe is the reason why the hint is needed so that its use will be understood by those that follow me.



My manager keeps nagging me to get certified. Certification requires purchase of at least one instructor-led training course. For the price of a five-day training course, I can buy dozens and dozens of good books. I tried to persuade my manager to give me a week

off to spend at the beach, reading a good book like Effective Oracle by Design by Tom Kyte, but he wouldn't bite and keeps nagging me about certification. My manager doesn't take my opinion seriously but he'll listen to you. Is he right about this or am I?

Personally, I think certifications can only effectively answer one question: Can a person study and memorize well enough to pass the required tests? There are people out there who attend some kind of certification boot camp for a couple of weeks and take all the tests and pass. There are others who buy exam study guides and take sample tests until they basically have most everything they'll need to know memorized so they can pass the tests. And there are people out there who have years of experience that don't even pick up a book or spend a very small amount of time to prep that pass the tests.

I really don't think certification is a measure of experience or ability. I wonder which employee a manager really wants managing their database: one with a certification but no proven experience or one without a certification but years of experience and a great track record.

If a manager is willing to pay for your certification, get it. If having a certification is the difference between you being chosen for a job over someone else equally qualified, then get it. But I think it's a mistake to use certification as a measuring stick for experience or an indicator of someone's ability to handle real-world, tough situations. Certification may get you in the door, but it's how you apply the knowledge you have and your ability to quickly diagnose and repair problems that will keep you there.



Will there be any jobs left for me in five years? Should I move to Bangalore? Are the Indian shops any good? How should I adapt? One of my previous employers suffered wave after wave of layoffs. I personally know many IT professionals who have

switched careers: a PeopleSoft engineer became a police officer, a project manager became an insurance agent. Should I plan on a career change? I've always wanted to be a bee-keeper.

My experience has been that outsourcing may provide a low-cost way to quickly build an application, but it is not a solution for optimally performing and easily maintainable applications for the long haul. IT staff will still be needed here in the database trenches, where they will have to provide ongoing real-time support for applications. Since lower costs and speed of delivery are often the only criteria placed upon foreign outsourced vendors, that leaves *lots* of room for in-house staff to be needed to monitor, fix, and improve what arrives from these outsourced shops.

What that means to us is that there may be fewer jobs available in the short term at least. So, it becomes imperative that people are on top of their game as far as their knowledge and skill sets are concerned. I know of one company who outsourced their entire development staff and sent the work overseas. They had more performance problems than they could count, very disgruntled users, and ended up spending more time and money in the end. Within one year, they'd hired back about 50% of the original staff to try and fix the work that was delivered from the overseas vendor. Within two years, they'd hired back 95% of the original staff (in terms of the number of developers). They found that if they'd just stayed fully staffed to start with, they would've actually saved more money!

I don't think anyone can make a blanket statement about the quality, or lack thereof, of outsourcing. But, I believe that well-trained in-house personnel can get the job done better and with less need for re-work than people who are thousands of miles away with no personal stake in the company that hires them.

Personally, I think if you love what you do and enjoy being good at it, then it's not time for a job change. It's time to work harder, learn more, and prove your value to your employer. But, if you're *not* doing something that makes you glad to get up every day and go to work, then that's the best reason to change jobs that I know of. Perhaps for those who aren't in IT because they love it, when times are tight and lay-offs occur, it may be time for them to re-evaluate where their heart is and move on to something else.

And, by the way, a friend of mine was a beekeeper and that's one job I can easily say that I'd never want to have! I'm terrified of (not to mention allergic to) the little creatures!



It costs us almost \$30,000 per year to produce and distribute the Journal and a little more than that to organize four conferences. We have about 500 members and a little less than 200 attendees at each conference. Our membership levels have been stagnant for a long time. Has Google made us obsolete? I begged my previous company to buy a corporate membership but, except for my manager, not a single person from that company ever showed up at our conferences. Should we close shop? What could we do better to attract more members?

I think that user groups play a very important part in the Oracle community. With economic conditions being what they are now, many companies want to keep spending on continuing education to a minimum. User groups like NoCOUG provide that local lower-cost option and not only help educate but provide great networking opportunities as well. I certainly don't think they are becoming obsolete. I think they're more important than ever. The issue is making sure people understand the value they get from having a good user group in their area.

First, there are cost-cutting strategies to help the organization be viable without need for as much financial backing. I've seen some user groups across the country reduce quarterly training day events down to just once or twice a year. I've seen reductions in general meetings from once a month to once a quarter. I've seen printed newsletters and periodicals converted to on-line formats. I've seen short online training events replace on-site training events. Google hasn't made user groups obsolete, but Google has shown us the power of getting quick access to information online and in a hurry.

Second, it's my belief that people are always willing to pay for things they believe provide meaningful value. Even when you can get some information online, the quality of that information isn't guaranteed. One of the great things about user groups is that they provide a place to get good information from reputable sources. Even when the user group doesn't sponsor an in-person event with a speaker of note, the group

can post links to qualified information on the Web and save members from having to weed out a ton of questionable content and get them straight to the good stuff.

Third, meeting with people who share common experiences with you is one of the best reasons to belong to a user group, in my opinion. It gives you an opportunity to learn from others who have already faced problems you're currently mired in and to be able to leverage their experience to help you find solutions for you. And, as we've already discussed, many of us are facing layoffs and outsourcing trends that make our jobs less secure than we once thought. Having a community of colleagues that can help you in searching out new job opportunities is critical, and user group connections can often provide the lead to your next job when you need it.

In the end it's all about giving people what they want and what they need in the most convenient and low-cost way possible. I certainly hope this is the time when user groups really rise and shine versus decline. There are few other opportunities to get so much for so little than what can be gained through a great local user group like NoCOUG.



Thanks for answering all my cheeky questions today. I'm always interested in buying good Oracle books. Do you have some personal favorites that you can recommend?

Cost-Based Oracle Fundamentals by Jonathan Lewis; *Troubleshooting Oracle Performance* by Christian Antognini; *The Art of SQL* by Stephane Faroult; *Optimizing Oracle Performance* by Cary Millsap and Jeff Holt; and anything and everything from Tom Kyte.

I've got a huge library of books that I've really found to be helpful, but these are the first few that come to mind. ▲

Interview conducted by Iggy Fernandez

Editor's Note: Here's the full quote from Dr. Scalzo's book: *"Person hours cost so much more now than computer hardware even with inexpensive offshore outsourcing. It is now considered a sound business decision these days to throw cheap hardware at problems. It is at least, if not more, cost effective than having the staff [sic] tuned and optimized for the same net effect. Besides, a failed tuning and optimization effort leaves you exactly where you started. At least the hardware upgrade approach results in a faster/better server experiencing the same problem that may still have future value to the business once the fundamental problem is eventually corrected. And, if nothing else, the hardware can be depreciated, whereas the time spent tuning is always just a cost taken off the bottom line. So, with such cheap hardware, it might be a wiser business bet to throw hardware at some solutions sooner than was done in the past. One might go so far as to make an economic principle claim that the opportunity cost of tuning is foregoing cheap upgrades that might fix the issue and also possess intrinsic value. Stated this way, it is a safe bet that is where the business people would vote to spend."*