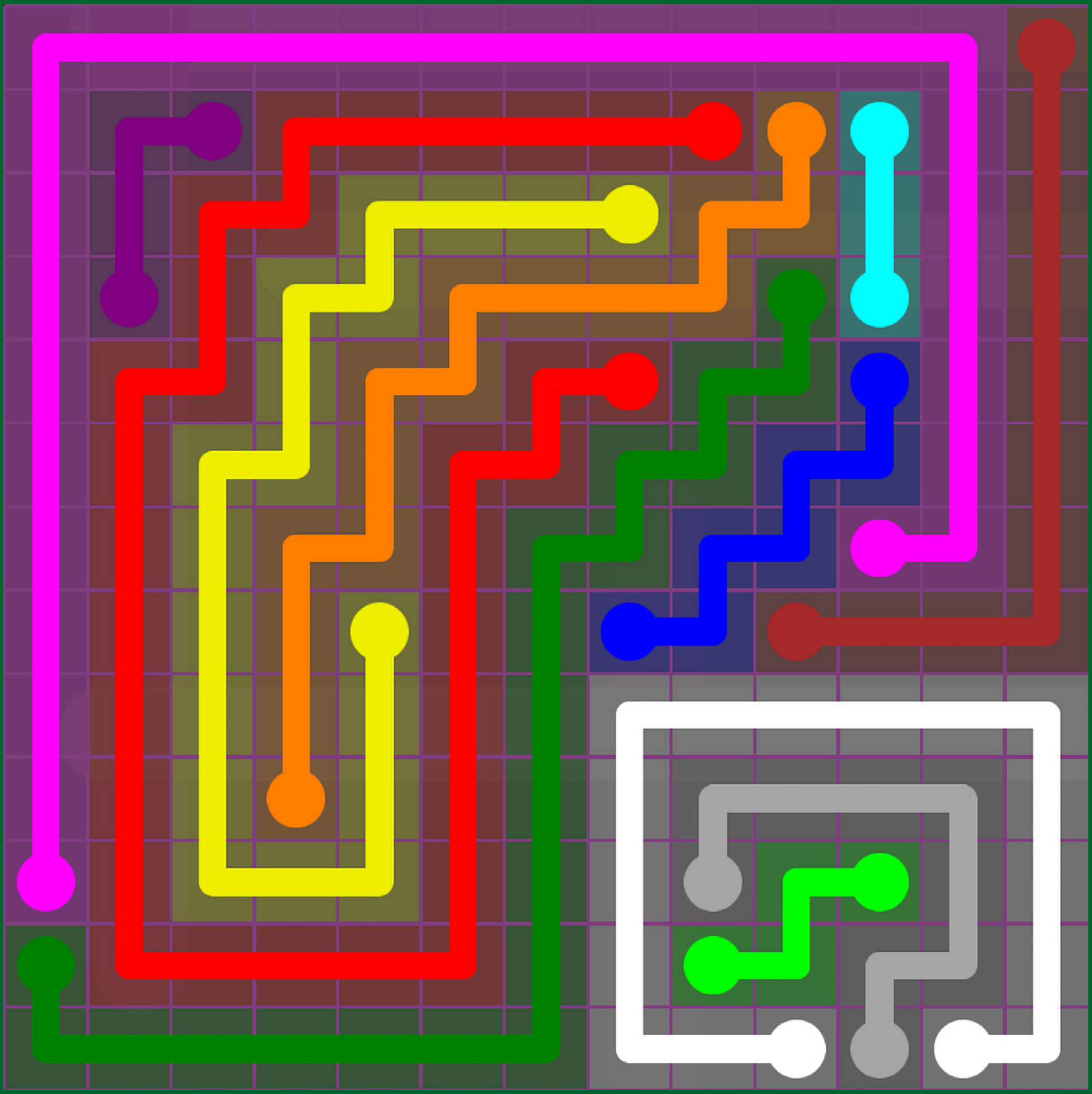
Official Publication of the Northern California Oracle Users Group

NoCOUG JOURNAL

Vol. 29, No. 3 · AUGUST 2015 $15



YesSQL!

With Stéphane Faroult

See page 4.

Raiders of the Data Dictionary

By Lothar Flatz.

See page 13.

Fourth International NoCOUG SQL Challenge

We have a winner!

See page 19.

Much more inside . . .

ce Antoinette Meredith Daisy Nada Liz Amie Jeannine John
a Robbi Kathyrn Steven Gisele Tim Laura Myron Elizbeth
e Sun Darell Karan Janine Bert Larry Rodrigo Rachel Todd
an Brittaney Sterling Nedra Simon Christy Guy Clare Rege
m Margaret Dan Sunny Nicho aitlin Kareem Loyd Meg
Brian Torrie Renato Jar Andy Sharlene Andre
na Frank Arlena mm la Ka en Tom
erri Sab en An enry Bev
holle C areem Er Na
Theron Stev Gisel
ty Elizbeth Jeann rice Antoinette M
ie Jeannine Theron elina Robbi Kathyr
inette Meredith Da nine Theron John
Robbi Kathyrn Gisel te Meredith Daisy

Professionals at Work

First there are the IT professionals who write for the *Journal*. A very special mention goes to Brian Hitchcock, who has written dozens of book reviews over a 12-year period. The professional pictures on the front cover are supplied by Photos.com.

Next, the *Journal* is professionally copyedited and proofread by veteran copyeditor Karen Mead of Creative Solutions. Karen polishes phrasing and calls out misused words (such as "reminiscences" instead of "reminisces"). She dots every i, crosses every t, checks every quote, and verifies every URL.

Then, the *Journal* is expertly designed by graphics duo Kenneth Lockerbie and Richard Repas of San Francisco-based Giraffex.

And, finally, Jo Dziubek at Andover Printing Services deftly brings the *Journal* to life on an HP Indigo digital printer.

This is the 115th issue of the *NoCOUG Journal*. Enjoy! ▲

—*NoCOUG Journal* Editor

Table of Contents

Interview....4

Book Review....9

SQL Corner....13

Special Feature....15

Tips & Tricks....17

SQL Challenge....19

Member Stories....22

Session Descriptions....24

Treasurer's Report....25

ADVERTISERS

Dell Software....2

OraPub....7

Axxana....8

Delphix....23

HGST....23

Database Specialists....27

SolarWinds....28

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 each issue is eight weeks prior to the quarterly conference. Article submissions should be made in Microsoft Word format via email.

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

ADVERTISING RATES

The *NoCOUG Journal* is published quarterly.

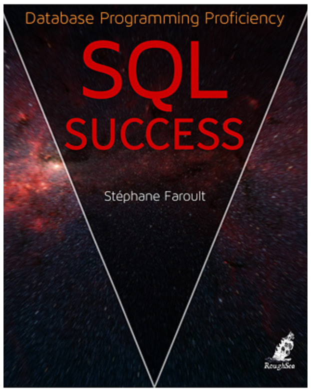
Size	Per Issue	Per Year
Quarter Page	$125	$400
Half Page	$250	$800
Full Page	$500	$1,600
Inside Cover	$750	$2,400

Personnel recruitment ads are not accepted.

journal@nocoug.org

YesSQL!

with Stéphane Faroult



When I was a beginning IT professional, RDBMS was a hot technology and that was where the money was. Companies paid big salaries to RDBMS professionals, so that is what I wanted to be. Today, the same applies to NoSQL technology. It is being used for highly visible applications such as healthcare.gov (Obamacare). How would you advise a beginning IT professional eyeing the job market? Is NoSQL a flash in the pan ?

The question of advising beginning IT professionals is one of some concern to me, as I have recently switched to teaching—first as adjunct faculty in French IT schools and currently as visiting faculty at Kansas State University—and whenever I teach, I imagine my students in their first job. My advice to them is to master SQL databases, because they are sure to bump into one of them pretty soon, but also to focus on data management as a whole more than on a particular technology. It's important to know what exists, what is available to solve real-life problems, and which specific problems a particular technology is trying to address. I would definitely not bet the house on one technology, especially as you often have in a company, from a technological standpoint, what you find in traditional societies: several generations under the same roof. Now, NoSQL? When I was a student myself, in the early 1980s, everybody was talking about the Japanese "5th generation of computers" and "Total Quality"; another hot topic in IT circles was IBM's "Information Center" concept. I have grown skeptical about everything that was hot in IT very quickly and, perhaps because IT came late in my engineering education, I'd rather think in terms of problems, solutions, and pros and cons than trendiness.

I am strongly convinced that the only proper way to handle data is the relational one, except perhaps for some niche applications, merely because the relational model was born out of logic; I'm not talking here about implementation—and definitely not about SQL either—but about normalization, constraints, and all that. Some applications can certainly thrive on a key-value model, but how much is the key-value model conceptually different from the ISAM files I was taught about (and never used)? After all, ISAM is definitely another kind of NoSQL data management. It's when you try to evolve your model and plug new applications into your data that plumbing becomes so complicated that after a while the whole building collapses under its own weight. If the relational model became ubiquitous, it's not by chance.

I also believe that SQL will remain around for a while (you just have to see the number of "NoSQL" products putting on an SQL layer), although not necessarily for the best of reasons: legacy applications, number of people claiming to know it, and its apparent easiness.

Something very interesting is happening these days, though—at least that's how I'm seeing it—which is that people are increasingly conscious that there is a mismatch between the architectures of SQL DBMS products and available hardware architectures, even if most vendors are trying to paper over the gap. When you think about it, today's big SQL products were all designed in the first half of the 1980s. The only exception is MySQL, the main innovation of which was to decouple storage engines from the upper layers. Since the switch from Oracle 5 to Oracle 6, which happened for most Oracle customers in the late 1980s and required a full database export and import, the basic Oracle architecture has remained the same. Computers and systems have changed a lot since then. Of course, many features have been added to DBMS architectures, some of them quite important ones, but it's commercially impossible today for Oracle to require for an upgrade what customers did in the late 1980s, when they had far smaller databases. As a result, Oracle and all big players with them are kind of stuck with the architecture they have had for years, architectures designed for a very different type of hardware, because their main customers are big corporations and you cannot push big corporations too fast. Once you have become an established name you have to respect the investment of your customers, and you are as tied to them as they are often tied to you. What we are increasingly seeing, though, are small companies that are really data-focused, and that's pretty new. A startup company hasn't the same constraints as a company that is part of the Dow Jones Industrial Average; it may not have enough funding to invest in upmarket hardware and highly priced software licenses, and yet it may require massive data management capabilities. Big companies also start data projects that are independent from traditional information systems, and I shouldn't forget cloud services providers, for whom flexibility, scaling, and resilience are core issues. This change in the ecosystem has opened a window of opportunity for a lot of people trying to think about data management in a different, far more distributed way on commodity hardware; and, even if I have so far been rather unimpressed by the NoSQL "movement," it raises a lot of very interesting questions for people who, like me, are concerned about data integrity. To take only one example, how do you manage uniqueness across nodes that you want to add or remove on the fly, especially when data is duplicated for availability concerns? Should transactions be ACID or not? How can you hide latency in a distributed environment? It's all about compromise, the type of data you are working with, how you process it, and whether one feature is worth its flip side. For me, that's what is important for a young IT professional today: having a global idea about available technologies and, as important as knowing what a technology brings to you, keeping a healthy dose of skepticism to discover its limitations, what you will have to give up if you adopt it, not forgetting deployment and maintainability issues, and how fast you'll be back on your feet that day when everything goes wrong. In this respect, traditional SQL products still have the upper hand, but it may not last forever.

It seems to me that we spend too much time tuning SQL. What is it about SQL that makes it so hard to get performance right the first time? I notice that your first book is called* The Art of SQL. *The choice of words seems to imply that SQL is a mysterious and difficult art instead of the easy-to-use technology that its creators envisioned it to be, intended to be used by accountants, engineers, architects, and urban planners who, "while they are not computer specialists, would be willing to learn to interact with a computer in a reasonably high-level, non-procedural query language."* ("SEQUEL: A Structured English Query Language," *Donald Chamberlin and Raymond Boyce, 1974)

I'm glad you quote the Chamberlin and Boyce paper, because when you read this paper you understand what kind of misunderstanding SQL has been from the very start, and I'm not far from relating most performance issues to the way SQL was designed and marketed. I haven't directly known this period (I was in high school in 1974), but when I came on the job market some 10 years later, you could still have a good idea of what it had been like, which is probably hard to imagine for most of today's IT professionals. There were no PCs in 1974 (well, the first PCs appeared at the beginning of the 1970s, but PCs became respectable enough to stand on a desk in a major corporation only when IBM announced one in the early 1980s). There weren't any spreadsheets, either—the first one, VisiCalc, only appeared five years later. Programmers were rare and expensive; whenever you wanted to extract data (from files), you had to beg for a Cobol program as thick as a phonebook (another disappearing dinosaur), and with luck you were getting the data extraction result on a printed listing six months later. Ted Codd's tables were, at least conceptually, easy to understand, and I guess that in the mind of the IBM management of that time all that was lacking was an "easy" language to lay above them, if possible with a syntax close to English (Shakespeare must be rolling over in his grave), and not the more mathematically oriented and rigorous language that Codd was working on. So, the focus of SQL was on being able to say, "Look, it's easy, all you have to write is "select * from employees where salary > 500 order by name" (I haven't adjusted the salary for inflation . . .), and that's it. Most of Codd's ideas, especially that tables that satisfied some conditions could be seen as variables on which you could operate, were completely occulted. Unfortunately that's still the way SQL is usually taught, more than 40 years later, with mostly a focus on an easy syntax. How often do you hear that although ORDER BY is definitely convenient for displaying a result, it's not a relational operation and whether rows are ordered or not you get the same information? Chris Date certainly repeats it, so do I as well as a few people I know, but if you google for "SQL tutorial" like most people do, chances are that you will end up believing that ORDER BY is a major relational feature, when it's not relational at all.

We have inherited a language that willfully sacrificed logical correctness to what I might call "easiness to complete level 1"; problems occur, very quickly usually, when you reach level 2 and above. As I have said previously, for Codd, and it's rather clearly stated in his 1970 paper, the beauty of relations was coming from the ability to combine them through various operations to obtain new relations. One key feature of relations, the absence of duplicates, is something that was never built into SQL. Uniqueness constraints can enforce it for tables, but nothing constrains you to have constraints, and I'm not sure that every developer clearly sees that having a sequence number or auto-incrementing column as primary key and no other uniqueness constraint doesn't turn a table into a relation. More importantly, nothing can ensure that the result of a query will be a valid relation. If you take an operation as basic as a projection (selecting only some columns), chances are that you will get duplicates as soon as you aren't querying a complete key. Allowing duplicates in a result makes sense when you consider the initial intent of easiness, because if, for instance, every query had automatically eliminated duplicates, many people would have wondered why querying only some columns was returning fewer rows than what the table contained. Unfortunately, it's when you want to combine one result with another that everything gets murky. Sometimes you get wrong results; sometimes, after much patching of small queries together, a developer notices that some rows are returned several times and adds a DISTINCT at the worst possible place—the outer query—to "fix" the problem. That's how you get some of the performance issues that you were referring to, because you do a lot of useless work and process a lot of data before ultimately discarding most of it. Even more critically, people who are stuck at level 1—and they are numerous—multiply simple queries when they could, with better skills, write only one, and they end up spending all of their precious time in context switches and network latency.

I called my book *The Art of SQL* primarily because of Sun Tzu's *Art of War* and because there was, for me, an obvious parallel between strategy and designing queries, but also because "art" implies both technical ability and enough intelligence (in its etymological sense of "embracing with one's mind") to follow one's purpose rather than strict academic rules. I went not long ago to a Velasquez exhibition in Paris, and I was struck by something that even a good painting reproduction doesn't quite render, which is how modern his painting was; the same is true of some of his contemporaries: Frans Hals or Rembrandt come to mind. Forget about ruffs; some portraits could have been painted two centuries later. Art is all about first understanding in depth and then letting your imagination work: in other words, not mystery but a solid set of basic skills used to innovate. Using an ORM is like painting by numbers. I find the "artistic" approach too often lacking in IT; far from being mysterious and difficult, "art" is for me the exact antithesis of Cargo cults. In Greek mythology, the goddess of Arts was Athena, who was also the goddess of wisdom and civilization (and of war strategy).

Your second book also has an unusual title:* Refactoring SQL Applications. *What's that all about?

This book has a complicated story. My O'Reilly editor for *The Art of SQL*, Jonathan Gennick, was more or less working on a book he wanted to call SQL Refactoring, partly because refactoring was hot then. He had worked with another author on an outline, which he submitted to me for feedback. I added a lot of things to the original outline because even if "refactoring" was a new word for me, I had spent years at customers trying to salvage hopelessly slow queries and applications with a minimum number of changes (sadly, you cannot scrap everything, redesign the database, and rewrite an application from scratch as often as you'd like), and I wanted to add the idea that very often problems lie not with a single bad query but a whole process that, for instance, loops when it shouldn't. As a result, Jonathan suggested that I join the team. I was just out of *The Art of SQL* and not really in the mood for writing another book, but the subject looked very interesting to me: there were things to say that I had never seen clearly explained anywhere and there were two good co-authors, so I accepted. Soon after, the third author fell out of the project for personal reasons. Then Jonathan quit O'Reilly and, after a while, joined Apress, which of course made writing a book for O'Reilly a bit difficult for him, and I remained alone on the project. I had other projects of my own, so for several months I didn't follow through; I only returned to the idea of writing this book when my other projects failed to materialize. That was an interesting book to write, because very often performance issues are detected extremely late: at best during the last rounds of testing when a production date has already been widely announced and at worst after a few satisfactory months in production. When you discover that the application isn't up to expectations, there is usually a lot of pressure, crisis meetings, and in some cases different teams that are quick to blame each other. It's very difficult to keep a cool head, especially for inexperienced developers. I have tried to write a methodical guide, starting from things that are easy to change or check, such as stats and indexing (it's always surprising to me how much as simple a topic as indexing on several columns is misunderstood), followed by a method for analyzing and rewriting queries, then rethinking database accesses when everything else either fails or isn't enough (which is why I changed the original title; I didn't want to narrow the topic down to individual queries). The idea was to guide the reader into what I usually do whenever I visit a customer: first try to extinguish the fire but also analyze processes more widely and tell them, "This, this, and this may be a big problem before too long; if I were you I would do it that way, so try to work on it and improve it with the next release before the ticking bomb explodes."

What's your approach to solving a SQL performance problem?

I assume that by "SQL performance problem" you mostly refer to an SQL development problem! The first point is to check that the issue really is with how the application is written, even if this is usually the case. I usually try to find out where all the time is spent and look for queries for which elapsed time is globally the greatest when people complain (with Oracle, I'll get the information from the Statspack tables or the DBA_HIST views). Elapsed time is the only thing that really matters to end users. If there is a huge difference between elapsed and CPU time, unless an obvious index is lacking, the issue usually lies with locking, I/Os, or network exchanges, and it's a very different kind of problem. When elapsed time is mostly CPU time, if the query is a simple, fast query executed a few zillion times every minute, there is nothing you can do apart from rewriting the application differently (and dumping that ORM). If the application was developed in-house, I can at least brief developers on the virtues of queries that go once to the database and return exactly the data you need. When you can identify queries that could probably do better, it depends on how many degrees of freedom you have. Can the query be rewritten? If not, if statistics look decent and selective columns are indexed, I mostly look for two things: One of them is the use of functions that would prevent using an index; you can try to create a function-based index in that case, although I'm always reluctant to add indexes as it's hard to predict what side effects there will be. It's easier to drop an index that is obviously useless than add one that may be excellent for one query but may also badly hurt a nightly batch process. The second thing I look for is correlations among columns that are used as search criteria. Oracle assumes that, to use Bill Kent's famous formula, "non-key columns depend on the key, the whole key, and nothing but the key," and therefore all columns are independent. Even when the database was neatly designed (unfortunately many databases look more like mushroom towns than carefully planned urban areas), columns rarely are as independent in practice as they should be in theory, because there is always an implicit connection between, say, *start_date* and *end_date* or *city* and *country*, which may make a combination of conditions far more selective than the optimizer thinks it is; very often it doesn't make much of a difference, but sometimes it's precisely the problem and it may be a good opportunity to use extended statistics. I rarely check execution plans (which surprises many people), because I find them more difficult to read than queries and it's far easier for me to say "this will [or should] be fast" or "this will be slow" by looking at a query rather than an execution plan. In the same way, I haven't used hints for decades, and I only use SQL profiles or Plan Management when I have no other option left. My priority is to ensure that the optimizer has all the information it needs to do its work properly. If there is a stability issue because of weirdly distributed bind values, I'd rather go, if I can, toward higher dynamic sampling values than profiles.

When you can rewrite queries, you can go further. I try (and this is a process I have explained in detail in *Refactoring SQL Applications*) to classify tables: tables I return data from—the output—tables on which I have conditions—the input—and tables that link the one to the others. Far too often, I have found tables that fit in none of those three categories and were purely useless, probably remnants from a copy-and-paste operation; I have often seen useless tables in some parts of UNION queries. I check that tables really are tables, not views; if I have a view I insert its text as a subquery in the FROM clause, and I ruthlessly remove everything that isn't necessary for the query. Sometimes it doesn't retain much of the view after this process. Same story with user-written PL/SQL look-up functions (a red flag for me): if possible, I try to get rid of the function and merge its code into the problem query. As a first step I try to concentrate on a query that returns the number of rows that I ultimately want, but not necessarily all the data, and I get rid of joins that cannot possibly result in fewer rows: joins on foreign keys, left outer joins for which there is no *is null* condition. Of course a *distinct* at the beginning of a complicated query may be a symptom of something wrong in join conditions. Tables that appear many times in the FROM clause often call for analytic functions or a single pass with an *in ()* condition and a *group by*. I like subqueries, but I like them nested. A multitude of independent subqueries begs for rewriting. Talking of subqueries, there is also an animal I have met a couple of times (including in a database course on the Internet, where it was presented as a regular subquery example), the correlated *in ()* subquery. No, it doesn't make any sense, and I have seen the Oracle optimizer completely at a loss on such a query.

With experience, I easily spot a lot of things that just don't look right to me. What takes the most time is usually trying to understand the intent of the original developer. There is a natural tendency in humans to first think of the most complicated way of doing something; when you are comfortable with a topic and when you have time, you rewrite, rethink, and rework what you have done. When you have tight deadlines and had a hard time writing a query that returns what it was supposed to return, the first draft remains the final version. And, of course, there is no code review. Very often, when I edit the text of a package, I feel like an archaeologist entering an untouched Egyptian tomb. I have found some surrealistic code in PL/SQL packages, such as a function that was counting, with a cursor loop, how many rows were containing a given value. I had no idea where it was used; it may still be in production in spite of my report, several years ago. When you rework an atrocious query, it may end up not being that atrocious after all. But when I reach the point when I can say "So *that's* what they want to obtain is," the toughest part of my job is done.

Thank you for spending so much time with us today. I have a final question that is of interest to prospective authors like me: I noticed that your last book, SQL Success, was self-published. What are the pros and cons of self-publishing instead of working with established book-publishing companies?

SQL Success is another curious story. I once got an email from a community college database instructor who had very much enjoyed my previous books and asked me whether I could recommend a book in the same spirit but for beginners, to use as a textbook with her students. I suggested a few good professional books, which she didn't like too much for students; besides, she would have liked the additional material that is usually supplied with textbooks (slides, test banks, etc.). I emailed her a number of slides I had done for seminars and professional training, and she used them apparently with great success in spite of the complete disconnect with her official textbook. As she is a person who knows what she wants, she gently pestered me for weeks, not to say months, until I agreed to write the book to go with the slides, which I did at a time when business was running slow. So I wrote the prequel to my previous books, what I wish young developers had been taught at school about SQL and databases but weren't, and it was field-tested as I was writing it. I first contacted trade publishers, and I had two proposals including one that came with an honest warning that textbooks weren't their cup of tea and that they hadn't the same network in academic circles as true textbook publishers. I signed a contract (with the other publisher, in fact), and was assigned an editor whose ideas about what this book should be weren't quite compatible with mine. We tried to work together for months without getting anywhere; I often had the feeling he wished me to guess what he wanted. Finally, as no other editor was available, I agreed with the publisher to cancel the contract and contacted a handful of true textbook publishers. One refused the book because it was competing with one of their own titles, one answered positively but after a few weeks without much happening, I was asked for a proposal rewrite, and then, after a while, for a list of professors who were ready to adopt the textbook. I already had a few eagerly awaiting for the book to be published, but if I had contacted a textbook publisher it was especially for their connections! At that point, I just started to wonder what their added value was, and I feared that the book would never be out before a few years had gone by; after a bit of research I resolved to publish it in the name of my company. I had enough acquaintances to have some very strong reviewers on DBMS products I was a little less familiar with (the book covers all the main SQL dialects), some people with a strong interest in design took the cover in charge, and it went rather smoothly.

Concerning pros and cons, one pro for me was that I could get the book out in time for people who were waiting for it, and it really was what decided me; another obvious pro is that instead of the standard 10% royalties on the bulk price, you get everything above printing and distribution costs, which is around 70%. Another important issue, and that's a problem I had with SQL Success, is that if what is new and original in your book is more how you approach the subject than the subject itself, chances are that most acquisition editors will only notice that it's about the same topic as another, not-too-old book in their catalogue, and won't want to torpedo a book that may still be selling. Whatever the qualities of your book, you won't get a chance. Contrary to what happens in SQL, the order is important in a book, and if I'm talking about subqueries very early, of ORDER BY very late, and of GROUP BY before joins, it's by design (and I was glad to see most of my undergraduate K-State students manage to write in a short time during their mid-term exam queries that would have required a lot of time from many more-seasoned developers). Another pro is that you also have some control on the final book price.

There are also quite a number of pros in favor of publishing companies. First of all, if you are writing your first book, and if you have the luck to get as good an editor as I found in Jonathan, you will learn a lot of things, and believe me it's worth the publisher's share of the pie. The editor is the one you are most in touch with, but you shouldn't neglect copyeditors, people who draw figures and who design the layout and the cover, and people who index, if you don't want to do it yourself. If you have a coauthor, the publisher can also act as a referee, and give more unity to the book. Because publishing houses have full-time marketing and sales teams, your book will get far more exposure, which doesn't necessarily translate into far more sales, but it's a good starting point, especially abroad (more than half the royalties are lost in translation, but it's good for the ego.) Finally, there is some prestige associated to publishing with a well-known name, and if promoting your name is part of your motivation, that may be important.

I don't regret what I did with *SQL Success*, even if I think it hasn't yet quite met the popularity it deserves; the problem isn't with the absence of a publishing house but more with academic inertia and professors who are reluctant to redesign a stable course, even if the 27th edition of the official textbook is so-so. A textbook change is a bit like switching from SQL Server to Oracle or the reverse. SQL Success might be useful to young professionals as well, and perhaps here a publishing house might have helped, but I'm not completely convinced. However, I don't exclude (in fact, I even have a project in mind) working with a publishing house again. Apart from putting on your resume, "I have published a book with so and so," the advantage of a well-known publishing house is the kind of burst it gives you on the market when the book is launched. It can be interesting in two cases: when you are writing on a topic with a close "Best By" date and your book is likely to get outdated in two years' time or less, and when you are writing on an almost unchartered topic, because it will help you get the "first mover advantage." Otherwise, if you are writing on a niche topic, if you know that whatever happens you won't sell tens of thousands of books, and if you have access to enough talent for publishing a book that looks professional enough, the DIY approach is quite valid. ▲

Stéphane Faroult first learned SQL in 1983 on IBM's SQL/DS, and in 1986 he joined a very small team at the newly created Oracle France after teaching for a year and a half at the University of Ottawa in Canada and then spending a year with IBM France. He left Oracle in 1988 for software development and returned to databases through consulting in August 1989. From then on, he has worked with databases, mostly Oracle, consulting for major French companies for 25 years. He has published books on SQL, given a series of seminars for Oracle in Asia, posted database tutorials on YouTube, and taught database classes to French IT students while consulting with a major bank. Enjoying teaching quite a lot, Stéphane is currently teaching computer science full-time at Kansas State University.

Safari Books Online

A Review by Brian Hitchcock

Summary

Overall review: Safari, provided by my employer, is a good resource, but I wouldn't pay for it myself.

Target audience: Anyone who uses Oracle and other enterprise software.

Would you recommend Safari to others?: Only if their employer pays for it.

Who will get the most out of Safari?: Anyone needing to learn about recent enterprise software and development products.

Is Safari platform specific?: No. I used it on a Windows laptop and PC as well as an iPad.

Why did I review Safari?: NoCOUG asked me to review it.

Overall Review

Safari provides online access to a large number of technical books covering many topics, including Oracle and other database technologies as well as many software development topics. The interface to read a book online is very good, with minor irritations. It is easy to get the information you need from the books that Safari provides. Safari appears to be very up to date with books as recent as a few months from their publication date. If you have access to Safari through your employer, there is no reason not to make use of it. If you are considering paying for Safari on your own, you need to make sure you will use it enough to make it worth the expense.

Editor's Note: *San Jose Public Library and San Francisco Public Library offer free access to Safari Books Online. San Jose Public Library also offers free access to Books24x7. Residents of California are entitled to become members of both libraries.*

Introduction

Technical books have been very expensive for many years. They also were already out of date by the time they were printed because it takes so long to get a hard copy book produced. The number of bookstores—physical brick-and-mortar stores where you can look at technical books—has also been greatly reduced. It is easier than ever to order almost any book online, but you don't get to look at the book very thoroughly before you buy.

All of these issues would be addressed if you could look at a good selection of technical books online for a reasonable fee. Safari, safaribooksonline.com, does this. But does it work and does it work well enough to really be useful for working professionals?

How to Get Started

I work for Oracle and through my employer I was able to create a Safari account at no cost to me. If you don't have access through your employer, Safari offers three plans: Pro, Teams, and Enterprise. The Pro plan is priced at $39/month or $399/year for a single user. Teams pricing starts at $399/year per user and is billed annually. You have to contact Safari for Enterprise pricing.

All three plans offer unlimited books, video courses, audio books, and support, along with an iOS app. They also offer an app for Android and BlackBerry. After that, the only additional feature of the Teams plan is some billing options. The Enterprise plan has additional features, but they all relate to account management options, usage reporting, and options to reassign licenses. I made sure my login worked on my Windows laptop and on my iPad.

Initial Reaction

I started by connecting to Safari using Chrome on my Windows laptop. When I logged in, I saw My Dashboard, which showed the following: Last Viewed in: All Recent Pages, and 35 Tokens (0 Expire Soon). I don't know what tokens are but we'll get to that. I also saw that My Account had 44,051 titles and a button for STATS. Along the left side there were links for Books and Browse Categories. The Featured Categories were Java, Oracle, Solaris, and Oracle Press. Since I signed up for Safari as an Oracle employee, I assume this caused these categories to be featured. You might see different categories if you access Safari as an employee of another company. Below this I had categories for Business, Math & Science, and many others. I could choose Browse Publishers as well.

- **Last Viewed in:** This didn't link to anything, but I hadn't looked at anything yet. After I accessed some books, this linked to the last page I had looked at.
- **All Recent Pages:** When I clicked on this, I got a page that displayed the 20 most recent books and videos I viewed, including a thumbnail of the book cover and the date of the last access.
- **35 Tokens (0 Expire Soon):** When I clicked on this, I was told that I had 35 tokens that could be redeemed for downloads. I'm not sure if these get renewed at some point (perhaps annually?) or not. Nor am I clear on what downloading means. Can I download a whole book for one token?
- **Browse Publishers:** Clicking this link brought up an extensive list of publishers. There were the familiar ones: O'Reilly,

Apress, and Oracle Press, and many that I didn't recognize: dpunkt, for example.

In the center of the screen was Featured Video, which showed a rotating set of videos. I noticed that none of them were database or Oracle related.

There were also sections labeled Java Titles, Oracle Titles, and What's New. I assume that these sections would be different if you connected as an employee of a company other than Oracle.

Ease of Use

I had selected two books, and when I logged in again these titles appeared in Last Viewed and All Recent Pages. When I clicked on the WebLogic Server 12*c* Admin title, I saw the cover artwork; the author and publisher information; and tabs for Overview, Table of Contents, Notes & Tags, Bookmarks, and Search This Book. In the upper right there was a large button labeled "Start Reading." Clicking this button brought up the cover of the book, almost filling the browser window. Down the left side there were items for Notes, Bookmarks, Search, and Contents. There were multiple options for viewing the page, including going full screen or displaying a grid of thumbnails of the pages. In the upper right were the arrow buttons to move forward or back through the pages, and for the first few pages, they would shift left and right as I moved along, making it more difficult to click on them. This went away once I got to the main text of the book.

I hadn't used the Zoom buttons to enlarge the page, but it was very large on my screen and very easy to read—much easier to read than the actual book. Being able to enlarge the pages is a great feature, one advantage that online reading has over reading a physical book.

Another welcome—and unexpected—feature was along the entire left and right edge of the page, where there were narrow columns that I could click to page forward and back. This was great, since I needed to scroll down to read to the bottom of each page, and the buttons at the top of the screen would scroll out of view.

When a page has a photograph, the loading speed is noticeably slower. There aren't many images in this book, but I wonder if this would be more of an issue in a book with images on every page—a book on Photoshop for example.

I noticed that there aren't any page numbers at the bottom of the pages as I was reading. I'm not sure if this is a problem or not, but it seems strange. For example, I was reading Chapter 1 of the WebLogic Server 12*c* Admin book, and as I got to the end of each page, I didn't know where I was, other than in that chapter. I checked the table of contents on the left of the page and again, no page information. I didn't know if the chapter was 10 or 20 pages long. Again, I'm not sure this is a bad thing, but I did notice the lack of page numbers. I also noticed that there wasn't any sort of graphic or slider or display that showed me where I was within the whole book. I've used apps for reading magazines online that have a bar across the bottom of the window that shows you where you are in the whole magazine. I like this feature and expected to see something similar while reading a book in Safari.

At the bottom left of the browser window I found buttons for creating a bookmark, creating a note, emailing the current page, and printing. Clicking the bookmark button turned that button yellow, which I assumed meant that the current page had been bookmarked. Clicking this button again removed the bookmark. I didn't see how to look at or use my bookmarks until I exited full-screen mode, and then I could see the Bookmarks link on the left. I also noticed that the upper-left corner of the page now had a yellow triangle, so it looked like that page had been dog-eared. As I bookmarked multiple pages, each of the bookmarks appeared in a list on the left. The date when each page was bookmarked was also shown. I had bookmarked three different pages in the first chapter, but the Bookmarks list only told me the section heading of each page. I didn't know if the bookmarks were from the first chapter or not. I wanted to open the Contents list as well as the Bookmarks list, but I could only have one list open at a time. I haven't used Safari long enough to say that this is a problem, but I would like the list of bookmarks to somehow give me a feel of where they are in the overall book. I think it would be useful to know if most of my bookmarks are in Chapter 1 and to be able to see where my bookmarks are among all of the pages of the book. Perhaps some sort of graphic display of the bookmarks would be helpful.

Clicking the Notes button brings up a separate window to create a note for this page. There is a button to Share This Note, which brings up another window where you can email the note to others. After I created a note for the current page, a yellow tag appeared in the upper left of the page and the note appeared in a list of notes along the left edge of the browser. Clicking the yellow tag, I could edit the note. When I made notes in a physical book, either by attaching a Post-it note and writing on that or by writing on the page itself, I could review the notes in order as I moved through the physical pages. As I saw each note, I knew which chapter it was in. With the list of notes I have in Safari, all I have is a list of the notes and their titles; I don't know where they are in the book.

You can also make notes within each page. You do this by selecting the text on a page. This brings up a window where you can choose Add Note/Tags or Highlight Only. You can create a note that will show up in the list of notes for pages, or you can simply highlight the selected text.

I clicked on the This Book link at the upper left, and it took me back to the page showing the cover art; details about the book; and tabs for Overview, Table of Contents, etc. Clicking on the Table of Contents tab, I could see a listing of the chapters and sections within each chapter. I would like to see my notes and bookmarks integrated with this display.

As I read more of the book, I noticed that when I was not in Full Screen mode, I could see the Table of Contents along the left side of the window. As I moved through the pages, the section of each chapter that I was in was highlighted, and this moved along as I read further. Instead of page numbers, each chapter of the book was broken up into a series of sections. For example, for Chapter 1, while I was reading through the "Upgrading Oracle WebLogic Server" section, it was highlighted. When I got to the next section, "Using the Administration Console," it was highlighted. When I was using Full Screen mode, the Table of Contents was not visible down the left side of the window, so I didn't know where I was in the chapter.

Printing was offered for each page, and when I tried to print the page I was reading, it was sent to my printer. The output looked fine.

I tried several times, and I couldn't find my way back to the first screen I saw when I logged in. This seemed odd, and I kept trying; I found that if I clicked on the "Safari Books Online" logo at the top of the page, I got back to the first screen.

Selection

When I first connected, My Account was showing 44,051 titles. When I connected again, a week or so later, there were 44,439 titles. I don't know if the rate at which new books are added is consistent, but this shows that new books are being added all the time.

In a previous life, I designed antennas for spacecraft so I thought it would be interesting to search for "antenna design." My impression of Safari was that it was primarily for software topics, and I didn't expect to find anything about antennas. I was wrong: there are 3310 titles for antenna design. Scrolling through the list of titles, some of them only had the word "antenna" in the description, but many of the titles covered hardcore antenna design and theory. This was a good example of what Safari has that I didn't expect. Safari covers a lot more topics than I expected.

Safari provides access to many publishers I didn't know about. When I did have access to bookstores that stocked technical books, I don't think they had as many books on the shelf for me to look at. The selection of titles available through Safari is much better that any physical bookstore I have seen.

Quality

I decided to pick two measures of book quality: how recent they are and how a book looks as I read it in the Safari interface.

Looking at Amazon for Oracle Press books, I found that the most recent one was *Oracle Database 12*c *Security*, which shows a publication date of January 5, 2015. Safari has this title. I wrote this review in early June of 2015. While this is not at all scientific, it appears that Safari is, at most, five months behind the latest publications from Oracle Press. I don't know if this applies to the other publishers that have books on Safari.

When I first connected, I noticed a button "STATS" but didn't look at it. It turns out that this leads to a graph showing the number and percentage of titles available to me that were published in each year from 2000 (and earlier) up through 2014. From this graphic, roughly 10% of the titles available are less than a year old, and another 40% are less than 5 years old.

Judging by the ones I looked at, books look great in the Safari interface. I was not aware of any issues when looking at a page in a book or any problems with images.

Searching

Clicking on the Books link on the left, I got a page with a Featured Book as well as several Latest Book choices. I also saw a message that Safari Books Online has over 41,000 books and that my subscription may not include all the books. I clicked on the VIEW ALL BOOKS link and saw the covers of 20 books, each with information about the book and a synopsis to the right of each one. By default, these are the 20 most popular titles. Scanning these titles I noticed that while there were titles for management, PHP, communication skills, and time management, the most popular topic was Photoshop! If most popular is not to your liking, you can also sort by Insert Date, Pub. Date, and the ever-popular Alphabetical.

I tried Alphabetical and scrolled down looking for "Oracle," but I didn't get very far. You can only see 20 titles per screen, and the choices for moving ahead in the list are limited. You can advance two screens (40 titles) or you can choose Next. This doesn't seem very useful. I was expecting options to choose letters of the alphabet so I could get close to "Oracle," but that wasn't possible.

Next I tried the search box at the top of the page. I searched for "Oracle," and I got a screen with tabs labeled Books (10285 Results) and Videos (67 Results). On the left I saw a breakdown of the categories for these results: Business, Desktop and Web Applications, Digital Media, and Engineering. At first glance, if 10,000 books have "Oracle" in the title, and I have access to roughly 40,000 titles, this isn't the level of diversity I was expecting. I'm glad they have so many Oracle titles, but I was expecting Safari to provide titles covering a wide range of technical subjects. I don't think Oracle books should be one-quarter of all the titles available.

The first 20 titles displayed are mostly Oracle Press titles; they appear to be very recent and cover a wide range of Oracle topics. I can SEARCH WITHIN RESULTS, so I tried "12c" and I got a list of 81 books and 3 videos.

I decided to look at *Oracle WebLogic Server 12*c *Administration Handbook* and Oracle Database 12*c* New Features. For each title there was a link to ADD TO FAVORITES, and I could view the Table of Contents or click the Start Reading link.

I decided to try one of the videos. I could see that there were 2456 to choose from, but I didn't see any way to search the videos. I could sort the listings, so I chose to sort by Alphabetical. I scrolled down to the end of the first page of listings and found "A Practical Guide to Learning Phone Skills." Perhaps it is not too late for me to learn some real job skills? "Acoustic Guitar Miking Techniques" is not something I was looking for, but there it is. Lots of Adobe and Photoshop videos. It took me a lot of clicking to find only two Oracle videos: "11*g* PL-SQL Fundamentals II" and "12*c* New Features, Part I." I decided to try the "12*c* New Features" video. It ran for a few minutes, describing six separate lessons, each covering a new feature of 12*c*. At the end of this video, clicking on the right arrow brought up the next, and the various lessons played. The videos loaded quickly and the quality was very good, even when displayed full screen.

I hadn't started looking at Safari expecting to look at videos, but now that I had, it occurred to me that the videos I watched were much better than the online training I have to suffer through at work. To experience quality audio and video covering training issues makes me realize just how bad the stuff I see at work really is.

I was puzzled that I couldn't search among the videos, so I decided to try searching using the Entire Site box. Clicking in this box brings up a menu, and choosing Advanced Search allowed me to search for "oracle" among video titles. This showed 23 videos. Much better. Oddly, when I was done looking at videos, I couldn't figure out how to get back to the main screen and had to log out and back in again.

Offline Storage

I was able to print a page as previously described. Now I wanted to try using my tokens. Returning to the main screen and clicking on the Tokens link, I found I had to request a download from the Content Reader toolbar. OK, I felt dumb: what and where is the Content Reader toolbar? My tokens will expire in 158 days! I hope that is long enough for me to find the Content Reader toolbar. I looked for about half an hour and then decided to try the Help link. Here I found Frequently Asked Questions, Offline Reading, and—under Articles—Accessing Content and Offline Reading. Here I was told that some publishers don't allow downloading at all, while some only allow chapters to be downloaded. If downloading is allowed, a button to request downloading will appear when reading the book or looking at the book information. There isn't such a button for the book I'm reading. I'm still a dinosaur, but the reason I couldn't find the download button was because it didn't exist.

I realize this is a touchy issue for publishers and authors. It would have helped me if Safari simply put a note on the book I was reading that downloads were not allowed.

Platform

I had been using my Windows laptop and Chrome browser to access Safari, but I wanted to try the Safari Apps for iOS. When searching the iOS App Store, you see Safari To Go and Safari Queue. I don't know why there are two apps available, but I downloaded both and, in the end, both installed Safari Online. I guess Safari Queue is an old app? I did this on both my iPhone 5 and my iPad. While I wouldn't suggest you try to read a book on an iPhone, I did access Safari through the web browser (confusing because I needed to access Safari Books Online through the Apple Safari browser) and iOS app. Using the browser on my iPhone, the interface looked identical to the one I see when using my laptop or PC. I was able to see where I had stopped reading the WebLogic 12*c* Admin book and I was able to search the book for all the occurrences of the phrase "Node Manager." You can find things you need to look up using the iPhone, but I don't think I would do so. Using the Apple Safari browser to access Safari on my iPad was the same as using my Windows laptop and PC.

For both the iPhone and iPad I tried the Safari To Go app, and the interface was different from using the web browser. I saw the books I had been reading, and I was taken to the page where I had stopped reading my current book. In the browser I could click on the left or right edge of the displayed page, whereas in the app I needed to swipe left or right. The screen on the iPhone was just too small, so I didn't keep using it. Continuing with the app on my iPad, the interface was different. Instead of having links down the left side of the screen (when using the web browser) I had icons in the upper right. When I clicked one of these I got an overlay with options. I didn't have to toggle off of full-screen mode to see the table of contents, for example. At the bottom of the screen was a slider that showed me where I was in the whole book. I liked both of these features. I couldn't find any way to make notes in the text using the app. I tried for a while but had no luck. I was assuming the app would offer the same functionality I had seen when using the web browser to access the book, but this wasn't so. This wasn't what I wanted; I wanted to learn one interface, not multiple interfaces.

Given what I saw of the Safari interface using the app on my iPad, I went back to accessing Safari through the web browser. I would not recommend using the iOS Safari app. I don't have an Android or BlackBerry device, so I can't offer any review of those app interfaces.

Impact on Authors

I don't know how publishers are compensated for their books that are available through Safari, and I don't know how the publishers compensate the authors of those books. Now that I have used Safari, I am less likely to buy an Oracle Press book, because I know I can access most or all of them for free through my employer-provided Safari subscription. Unless authors are well paid for their books to be available through Safari, I think this must be a reduction in what an author can earn from a technical book. On the other hand, more people may look at an author's book, since access is free. I think what an author gets from being published is changing. In the old days (remember, I'm a dinosaur) a band went on tour to sell albums; the album sales were where the money was. Today, the band effectively gives their music away online to entice listeners to attend their tour, and the money is made selling tickets to the tour events. I think an author of a book that is available through Safari isn't being paid much. I guess the author might get more exposure online, but I'm not sure what the equivalent is for the author to the band selling tickets to the tour events. Are modern authors effectively giving away their books to build a brand that can be monetized in other ways? I'm not sure this is a good thing for authors. Time will tell, and it will be interesting to see how many new books continue to be written if the authors are not compensated in some meaningful way.

Conclusion

I wonder if Safari is, like all of us, still learning what works best for reading books online. I am not sure I need page numbers while reading a book online, for example, but I was expecting it, and not having them seemed odd. It is possible that many smarter people have thought all about this and come to the conclusion that things like page numbers are not necessary. Maybe most Safari users don't care whether or not they have page numbers, or they prefer reading a book without page numbers.

Reading the same book in Safari is not the same as reading the physical book. This may not be a bad thing, but it is different. I also fully embrace the simple fact that I'm a dinosaur, doomed to remember a time before the Internet, so maybe I've been reading physical books for too long.

Now let's get to the heart of the matter: would I pay $39/month for Safari? No. Amazon offers Oracle Press books for roughly $40 for the paperback and about $38 for the Kindle edition. Assuming I really want to read several Oracle Press books over the next year, it would be cheaper to buy the specific books I want from Amazon versus paying Safari $39 every month for the next 12 months.

You could pay for Safari and just browse, reading parts of many books, but how many people want to do this for $39 per month? I don't think many people want to pay that much to look at technical books, but I may well be wrong. Safari clearly provides a wide variety of books covering many technical fields. I don't know how many people working in those technical fields work for employers that pay for Safari. My guess, based on my situation, is that most people using Safari do so because their employer provides it to them at no cost. I don't think most Safari users would pay for the access on their own. ▲

Brian Hitchcock worked for Sun Microsystems for 15 years supporting Oracle databases and Oracle Applications. Since Oracle acquired Sun, he has been with Oracle supporting the On Demand refresh group and, most recently, the Federal On Demand DBA group. All of his book reviews and presentations—and his contact information—are available at **www.brianhitchcock.net**. *The statements and opinions expressed here are the author's and do not necessarily represent those of Oracle Corporation.*

Raiders of the Data Dictionary–Part II

The Curse of the Buffer Cache

by Lothar Flatz

Lothar Flatz

In his rules, designed to define what is required from a database management system, Edgar Codd states in rule number four: *"The system must support an online, inline, relational catalog that is accessible to authorized users by means of their regular query language. That is, users must be able to access the database's structure (catalog) using the same query language that they use to access the database's data."* ("Is Your DBMS Really Relational?", ComputerWorld, 14 October 1985)

I have used the exhaustive knowledge that the database holds about its own structure in many ways during my ten years with Oracle Consulting. Sometimes when you study the structures of the dictionary, chance does lead you to a nice discovery, as in this case, when I found that you can draw sort of a map of an application very quickly by querying v$segment_statistics. When will you go ahead and uncover the treasure in your own database?

A Map of Database Activity

On various occasions I have used the query in Figure 1 to understand an application that is new to me. The result is sorted by logical reads percentages, and that means it is sorted by activity.

It is self-evident that the segments showing the highest activity in logical reads are also the ones that have the focus of the application. Therefore, when looking at the top activity segments, we also see the segments that are the most relevant.

Normally you would expect a linear correlation between logical and physical reads. Indeed, the old rule that you should have a hit rate of > 90% would suggest something like "physical read" <= "logical read"*0.1. When I did the query for the first time I was curious to know if reality would be in line with expectation. Since then I have learned that in most real-life cases, logical and physical read percentages do not go together. It is quite often very instructive to explore their distribution, which will reveal weaknesses, most of the time in physical design. With training, you can almost guess by the distribution percentages what the weakness will be.

```
SELECT
  OBJECT_NAME,
  OBJECT_TYPE,
  "logical reads",
  "physical reads",
  ROUND(ratio_to_report("physical reads") over ()* 100,2) "% physical reads",
  ROUND(ratio_to_report("logical reads") over () * 100,2) "% logical reads"
FROM (
  SELECT
    OBJECT_NAME,
    OBJECT_TYPE,
    SUM(DECODE(STATISTIC_NAME,'logical reads', value ,NULL)) "logical reads",
    SUM(DECODE(STATISTIC_NAME,'physical reads', value ,NULL)) "physical reads"
  FROM v$segment_statistics
  WHERE owner = user
  AND STATISTIC_NAME IN ('logical reads', 'physical reads')
```

Fig. 1: Statement to query segment activity

OBJECT_NAME	logical reads	physical reads	% physical reads	% logical reads
PATIENT	11,027,020,480	6	0	72.16
TREATMENT	2,280,336,592	8,438	1.02	14.92
PATIENT_HISTORY	621,398,800	529,544	64.22	4.07
DOCUMENT	471,252,336	79,163	9.6	3.08
PATIENT_HISTORY_ASS	311,690,656	30,524	3.7	2.04
PROCESS	242,127,360	32,042	3.89	1.58
TREATMENT_DECISION	128,403,280	144,862	17.57	0.84
TREATMENT_DETAIL	115,552,080	0	0	0.76
USERINFORMATION	42,014,800	0	0	0.27
PROCESS_ACTIVITY	40,782,768	0	0	0.27

Fig. 2: Segment activity for walk-in clinic

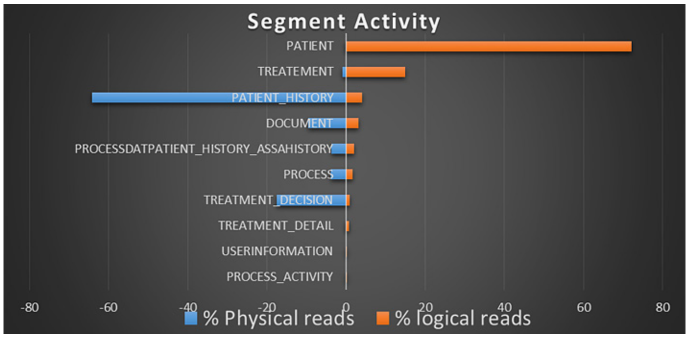


Fig. 3: Graphical comparison between physical and logical reads

Let's have a look at a real-world example from a walk-in clinic. We can see that the activity of the patient segment is dominant (Fig. 2). We can see imbalances in various areas. First, do we believe that 72% of the overall activity is focused on patient? Granted patient is an important table in the context. But is it that dominant? To some extent this rationale also applies to treatment. How about patient history? Why is this segment generating 60% of the overall I/O activity? And treatment decision? Only 1% of all logical reads are generating 17% of physical I/O.

What is wrong with these tables? Graphically the imbalances become even more obvious (Fig. 3).

Segment Statistics for the PATIENT Segment

So how would we investigate the patient issue? Well, my assumption was that the number of logical reads on patient is too high—probably caused by some inefficiency. Why not use a straightforward query for high buffer gets?

We can query for statements with high buffer gets that refer to patient. Unless statistics_level is set to all, we cannot query directly for high buffer gets on patient. However the indirect approach proves to be working well enough. For patient, treatment, and treatment_decision, we discover missing indexes as the root cause.

```
SELECT sql_id, executions, buffer_gets
FROM gv$sql
WHERE sql_id IN (
  SELECT sql_id
  FROM gv$sql_plan
  WHERE object_name IN ('PATIENT')
)
ORDER BY buffer_gets DESC
```

Fig. 4: Statement to query segment activity

Segment Statistics for the PATIENT_HISTORY Segment

Patient history is a classic under the performance issues, and it is worth some discussion. I previously researched a sanatorium's database. As you might expect they have long-term patients. Patients were coming in for a year, several weeks, or a month of treatment. Obviously, history was checked quite often. Since patients were coming over a number of years, the data blocks containing the data of one specific patient were scattered all over the segment.

It is impossible to cache this segment, since any prediction based on a hit ratio is very difficult. Therefore reading patient history generated a high number of physical reads causing a wait time up to ten minutes. My research revealed that we had an average of 77 rows per patient and the maximum was 10,400. Under this circumstance you have only one chance. You need to cluster the data. You can do this using an IOT, an Index Cluster, or a hash cluster. Hash partitioning by patient ID helps somewhat. In the case of the sanatorium's database, it reduced the physical I/O by 50%, which was still an unsatisfying result. But clustering by patient ID using an IOT resulted in dramatically lower I/O and better response times.

Thus, when I saw the segment name "patient history" I blurted out that I knew the issue already. Whenever you think you are rather clever, fate will teach you humbleness. Fortunately I did check the numbers before I proposed a solution. It turned out that we had 2.13 rows per patient ID on average in the case of the walk-in clinic, which are residing in 1.4 data blocks. Actually there is no need for clustering. What is generating so many I/Os?

Hit Rate Fever

We can discover the actual cause of the I/O bottleneck by looking into the buffer cache. When we sort the segments by logical I/O percentage, patient history comes out as number three.

When we sort the segments by the percentage in which they are cached, patient history is number 70!

Let's back up here. Segments are kept in the buffer cache based on hit rate (with some refinements that we skip here), which is basically the number of logical reads per data block. As we know, the number of logical reads for the segment is high; the only possible explanation is that this activity is spread across a high number of blocks compared to other segments that go before patient history in the cache. Basically, that implies that patient history is pushed out of the cache by smaller segments that show less per-segment activity but more per-block activity.

> ***"In most real-life cases, logical and physical read percentages do not go together."***

This push-out mechanism is certainly harmful in the sense that it results in a lot of physical I/O. When we have made sure that the access to the table is efficient and cannot be improved, there is only one possible consequence: we must increase the buffer cache.

That we did, and as expected, the physical reads dropped.

Buffer Cache Contamination

To be complete in buffer cache diseases we need to mention buffer cache contamination. In principle this is a process where your data blocks contain less and less useful content. To maintain the same amount of physical I/O the buffer cache needs to be increased. The most obvious example is time-dependent data.

Let's assume we are talking about a segment where the rows will carry a timestamp, like the effective date in PeopleSoft.

As a consequence any update will lead to a new row being created, and the effective date is set to the current date and time. The previously current row is now automatically outdated. We can safely assume that most of the time the underlying application will deal with current data. Thus, the data block that contains the row no longer current became a little less useful, and the number of rows increased. That way the number of current rows is automatically spread over an increasing number of data blocks, causing caching to become a little more inefficient every time that happens.

The only effective countermeasure seems to be partitioning based on the time stamp, forcing the current rows into concentration in a current partition.

Conclusion

There are many reasons for the buffer cache not to function as well as it could. In many cases, improving the physical design (indexing, clustering, partitioning) does help. When we run out of these options, the buffer cache needs to be increased. ▲

Lothar Flatz started working with Oracle Database in 1989, in the days of Version 5. He worked for Oracle Corporation for fifteen years and was a member of the Real-World Performance Group for two years. He is a member of the Oak Table network of Oracle scientists and specializes in performance tuning.

Re-Platforming Oracle Forms and Oracle Reports in the Cloud

Sri Rajan

by Sri Rajan

Over their 30-year history, Oracle Forms and Oracle Reports development tools and the apps on which they're based have been deployed worldwide by thousands of companies. While innovative when they were introduced in the 1980s and subsequently evolved to become powerful database-driven tools in the 1990s, today they limit an organization's options to run on *their* platform of choice. Organizations have built or licensed Platform as a Service (PaaS)-based cloud apps at an unprecedented pace in the past decade as *the* architecture standard to enable business agility.

Re-Platform to Exploit Advantages of PaaS

Characteristically, these new-generation PaaS apps, unlike Oracle Forms and Oracle Reports, offer customers freedom of choice to run everywhere and run fast:

- ➤ HTML5 "responsive" UI
- ➤ Any application server
- ➤ Any web browser and any iOS/Android device
- ➤ Run *without* plug-ins (applet, flash)
- ➤ Run on 64-bit systems at runtime and *design-time*
- ➤ Support continuous integration
- ➤ Aspect Oriented Programming (AOP), Spring
- ➤ Lower maintenance cost from deployment ubiquity

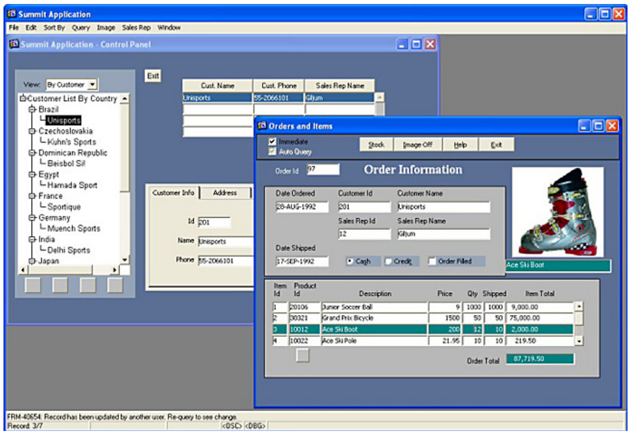
When these PaaS characteristics are deployed on cloud offerings such as IBM Bluemix, Oracle PaaS, Amazon Web Services (AWS), and Google Cloud Platform, customers can respond more effectively to high-velocity markets disrupted by *transient* competitive advantage. (*"Transient Advantage,"* Rita Gunther McGrath, Harvard Business Review, June 2013.) Strictly-speaking, PaaS does not require application multi-tenancy. For end customers, multi-tenant architecture is not meaningful when each customer's application, by origin and definition, is wholly different from other customers' applications. For Independent Software Vendors (ISVs), however, multi-tenant architecture has utility, since well-designed apps will separate customer-specific extensions from an ISV's base application that is served to its end customers. In these contexts, multi-tenant resource sharing in the ISV's PaaS cloud is important to better utilize infrastructure for scalable management.

Compatibility for Strategic Apps

Existing enterprise apps with business processes that are sufficiently complex, specialized, and mission-critical to an organization's strategic operations will not be decommissioned without great deliberation on functionally equivalent replacement alternatives. Given that IT budgets are limited, rewriting such apps is gated by cost, time, and risk barriers, especially in the context of enterprise Oracle Forms app migration.

Gartner elaborates on these considerations in a 2007 research report. (*"How to Maneuver Oracle Forms into an Ideal Position for Next-Generation Challenges,"* Mark Driver, Gartner Research ID Number: G00146666, July 3, 2007). Now, 8 years later, skilled resources in Oracle Forms and Oracle Reports PL/SQL who know the app well and are skilled in PaaS to implement precise equivalent behavior are rare. For enterprise-scale apps that contain substantial client-side processing logic where tolerance for deviation is low, this added constraint of *compatible execution* imposes a daunting challenge to PaaS developers.

Consider the illustrations in Figures 1 and 2 that compare an Oracle Forms v6 client-server app to its re-platformed PaaS equivalent running on Apple iPad.


Summit Application
File Edit Sort By Query Image Sales Rep Window
Summit Application - Control Panel
Exit
Orders and Items
Stock
Image Off
Help
Exit
Customer Info
Address

View: By Customer

Customer List By Country
- Brazil
  - Unisports
- Czechoslovakia
  - Kuhn's Sports
- Dominican Republic
  - Beisbol Sil
- Egypt
  - Hamada Sport
- France
  - Sportique
- Germany
  - Muench Sports
- India
  - Delhi Sports
- Japan

Cust. Name	Cust. Phone	Sales Rep Name
Unisports	55-2066101	Giljum

Id 201
Name Unisports
Phone 55-2066101

☑ Immediate
☑ Auto Query

Order Id 97

Order Information

Date Ordered 28-AUG-1992
Customer Id 201
Customer Name Unisports
Sales Rep Id 12
Sales Rep Name Giljum
Date Shipped 17-SEP-1992
● Cash ○ Credit □ Order Filled

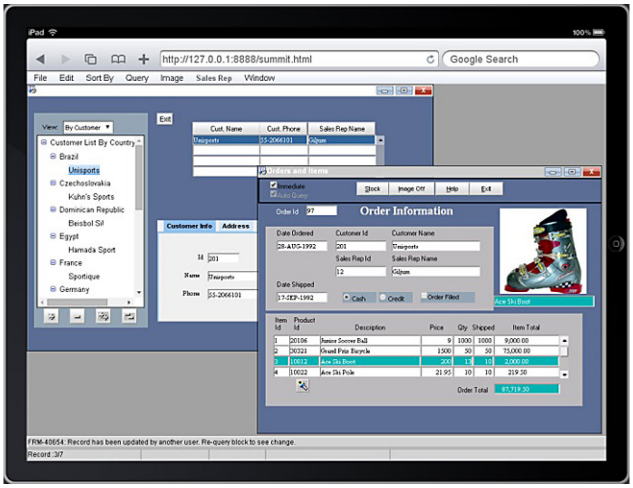
Ace Ski Boot

Item Id	Product Id	Description	Price	Qty	Shipped	Item Total
1	20106	Junior Soccer Ball	9	1000	1000	9,000.00
2	30321	Grand Prix Bicycle	1500	50	50	75,000.00
3	10012	Ace Ski Boot	200	12	10	2,000.00
4	10022	Ace Ski Pole	21.95	10	10	219.50

Order Total 87,719.50

FRM-40654: Record has been updated by another user. Re-query to see change.
Record: 3/7 <OSC> <DBG>

Fig. 1: Pre-PaaS Baseline: An Oracle Forms application illustrating the "Summit Sporting Goods" application from Oracle Corporation's classic "scott/tiger" schema.



Customer List By Country
- Brazil
  - Unisports
- Czechoslovakia
  - Kuhn's Sports
- Dominican Republic
  - Beisbol Si!
- Egypt
  - Hamada Sport
- France
  - Sportique
- Germany

Cust. Name	Cust. Phone	Sales Rep Name
Unisports	55-2066101	Giljum

Id 201
Name Unisports
Phone 55-2066101

Order Id 97

Date Ordered 28-AUG-1992
Customer Id 201
Customer Name Unisports
Sales Rep Id 12
Sales Rep Name Giljum
Date Shipped 17-SEP-1992
● Cash ○ Credit □ Order Filled

Item Id	Product Id	Description	Price	Qty	Shipped	Item Total
1	20106	Junior Soccer Ball	9	1000	1000	9,000.00
2	30321	Grand Prix Bicycle	1500	50	50	75,000.00
3	10012	Ace Ski Boot	200	13	10	2,000.00
4	10022	Ace Ski Pole	21.95	10	10	219.50

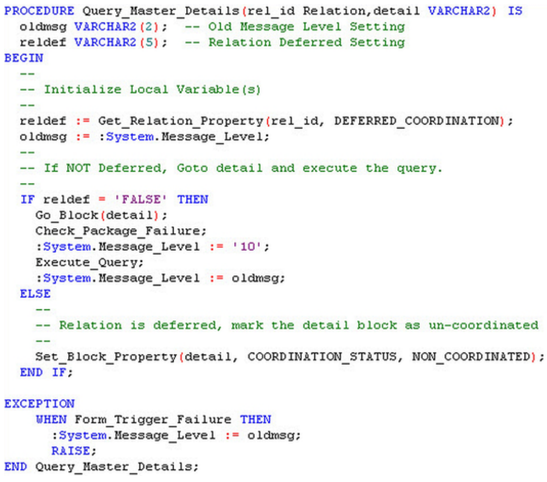
Order Total 87,719.50

FRM-40654: Record has been updated by another user. Re-query block to see change.
Record :3/7

Fig. 2: Re-platformed to PaaS: An HTML5 cloud-equivalent application that preserves execution fidelity and behavior. The above "compatible" design target runs without an applet and without JRE on Apple iOS and Google Android devices. All transient behavior, including pessimistic locks and user and system events, is preserved.

Compared with Figure 1, Figure 2 shows a representative and straightforward target that would be expected of enterprise-scale apps that demand fast data entry in OnLine Transaction Processing (OLTP) workloads. *All* behavior—transient (with intended side-effects) and final (with pre-validated state-based data)—would be expected to run as-is in OLTP apps.

As a former employee of and contributor to Oracle Tools Division in the 1990s, I worked with a team of impassioned developers to change the way customers build database-driven apps. I've seen customer usages evolve as they upgraded through successive releases of Oracle Forms, from SQL*Forms v3.0 to Oracle Forms 9*i*. For the above Summit Sporting Goods app, as an example, let's take a closer look in Figure 3 at the underlying Forms PL/SQL code that handles data synchronization between the "master" ORDER data block and its associated "detail" ITEM data block. As seen in the code fragment, Oracle Forms utilizes relations, system variables, block properties, focus navigation, a PL/SQL library of "built-ins," and exception processing on triggers for data synchronization.



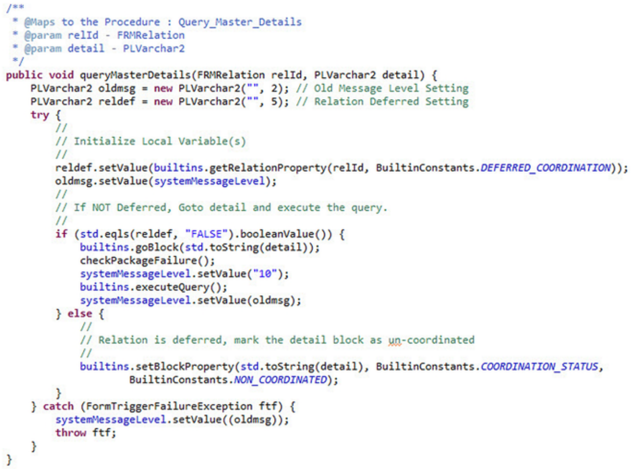
```
PROCEDURE Query_Master_Details(rel_id Relation,detail VARCHAR2) IS
  oldmsg VARCHAR2(2);  -- Old Message Level Setting
  reldef VARCHAR2(5);  -- Relation Deferred Setting
BEGIN
  --
  -- Initialize Local Variable(s)
  --
  reldef := Get_Relation_Property(rel_id, DEFERRED_COORDINATION);
  oldmsg := :System.Message_Level;
  --
  -- If NOT Deferred, Goto detail and execute the query.
  --
  IF reldef = 'FALSE' THEN
    Go_Block(detail);
    Check_Package_Failure;
    :System.Message_Level := '10';
    Execute_Query;
    :System.Message_Level := oldmsg;
  ELSE
    --
    -- Relation is deferred, mark the detail block as un-coordinated
    --
    Set_Block_Property(detail, COORDINATION_STATUS, NON_COORDINATED);
  END IF;

EXCEPTION
    WHEN Form_Trigger_Failure THEN
      :System.Message_Level := oldmsg;
      RAISE;
END Query_Master_Details;
```

Fig. 3: An Oracle Forms PL/SQL local program unit

In a re-platformed PaaS-compatible app, the *degree* of compatibility is a function of the business requirement—if the baseline application cannot be broken in any respect, we must first identify and resolve many fundamental issues, such as differences across programming languages; differences in data type semantics across database, app server, and client processing tiers; transaction integrity; navigation integrity; event architecture; and execution. Additionally, Oracle Forms–specific functions embodied in its associated "built-ins" library that comprise its runtime behavior must be understood thoroughly and support customer use cases, regardless of usage complexity.

While many approaches could be used to achieve compatibility in PaaS, maintaining linearity on a 1:1 basis boosts productivity through developer familiarity with their existing codebase. Figure 4 illustrates this concept of compatibility that is constrained by the technical requirement of structured readability for simplified maintenance in a re-platformed PaaS stack.



```
/**
 * @Maps to the Procedure : Query_Master_Details
 * @param relId - FRMRelation
 * @param detail - PLVarchar2
 */
public void queryMasterDetails(FRMRelation relId, PLVarchar2 detail) {
    PLVarchar2 oldmsg = new PLVarchar2("", 2); // Old Message Level Setting
    PLVarchar2 reldef = new PLVarchar2("", 5); // Relation Deferred Setting
    try {
        //
        // Initialize Local Variable(s)
        //
        reldef.setValue(builtins.getRelationProperty(relId, BuiltinConstants.DEFERRED_COORDINATION));
        oldmsg.setValue(systemMessageLevel);
        //
        // If NOT Deferred, Goto detail and execute the query.
        //
        if (std.eqls(reldef, "FALSE").booleanValue()) {
            builtins.goBlock(std.toString(detail));
            checkPackageFailure();
            systemMessageLevel.setValue("10");
            builtins.executeQuery();
            systemMessageLevel.setValue(oldmsg);
        } else {
            //
            // Relation is deferred, mark the detail block as un-coordinated
            //
            builtins.setBlockProperty(std.toString(detail), BuiltinConstants.COORDINATION_STATUS,
                    BuiltinConstants.NON_COORDINATED);
        }
    } catch (FormTriggerFailureException ftf) {
        systemMessageLevel.setValue((oldmsg));
        throw ftf;
    }
}
```

Fig. 4: Maintaining linearity on a 1:1 basis for readability

Three Compatibility Attributes

To achieve the above transformative technologies consistently and completely for Oracle Forms and Oracle Reports for *any* customer-developed application, our developers at NeoWorks have observed several salient attributes shaped through numerous customer engagements, three of which we'll discuss here: (1) "long running" transactions; (2) distributed session/state across all computing tiers (database, app server, and client); and (3) application-specific logic and its dependency on event architecture.

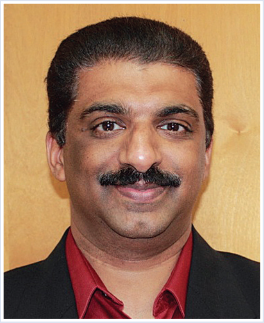
For long-running transactions, row-level pessimistic locks that are held by users for prolonged periods—a hallmark of Oracle Forms OLTP apps—can lead to data inconsistency in modern HTML5 apps if not addressed across all transaction use cases. For distributed session and state management, the Oracle Database allows the application state to be stored in database-stored package variables, which poses a significant challenge in PaaS architectures that leverage connection pooling for better system resource utilization. For customer application-specific logic, Oracle Forms is dependent on Oracle Database–specific datatypes, such as PL/SQL BOOLEAN, PL/SQL RECORD, and PL/SQL TABLE.

These datatypes are not supported in modern programming languages like Java/Javascript. For example, in Figure 4, the code fragment shows an extension to the default "=" operator in a Java

(continued on page 26)

Many Things Oracle

by Biju Thomas



Biju Thomas

Editor's Note: Biju publishes daily Oracle tidbits on Facebook (fb/oraclenotes) and on Twitter (@biju_thomas).

Trace File Analyzer Collector

Trace File Analyzer (TFA) Collector is a simple mechanism that collects diagnostic data to address failures. TFA has grown quite a lot in recent months and now is almost like a product by itself, with a large number of tools bundled in. All of these tools are/were available for standalone download and execution, but having them under one umbrella certainly helps. Also, the latest version of TFA can even generate AWR reports for the time period you specify.

Starting with version 11.2.0.4 of Oracle Grid Infrastructure (GI) install, TFA is installed automatically under the GI home. This version is mostly old; for the newer features you have to upgrade/patch TFA. For non-GI homes and non-RAC instances, TFA can be installed by downloading it from MOS. You install TFA as root user.

Why should you as a DBA have TFA installed and have it running? Well, the best answer I have (from experience) is to get MOS analyst to respond quicker to a failure service request and not bother you with a lot of redundant and unnecessary questions. Once you upload the TFA collection zip file covering the time of the failure, it includes pretty much everything needed for the support engineer to start digging.

Recently I did a 12c 12.1.0.2 GI cluster install and TFA was included as promised, but the version of TFA was 12.1.2.0.0.

```
$ tfactl print status
.--------------------------------------------------------------------------------------------------.
| Host      | Status of TFA | PID   | Port | Version    | Build ID             | Inventory Status |
+-----------+---------------+-------+------+------------+----------------------+------------------+
| zildb806a | RUNNING       | 78190 | 5000 | 12.1.2.0.0 | 12120020140619094932 | COMPLETE         |
| zildb806b | RUNNING       | 69953 | 5000 | 12.1.2.0.0 | 12120020140619094932 | COMPLETE         |
'-----------+---------------+-------+------+------------+----------------------+------------------'
```

According to MOS master note for TFA, the most recent version is 12.1.2.5.0 (TFA Collector – Tool for Enhanced Diagnostic Gathering – Doc ID 1513912.2)

When I applied the latest PSU patch, July 2015, TFA also got updated to a newer version.

```
$ tfactl print status
.-------------------------------------------------------------------------------------------------.
| Host      | Status of TFA | PID  | Port | Version    | Build ID             | Inventory Status |
+-----------+---------------+------+------+------------+----------------------+------------------+
| zildb806a | RUNNING       | 9228 | 5000 | 12.1.2.4.2 | 12124220150629072212 | COMPLETE         |
| zildb806b | RUNNING       | 8898 | 5000 | 12.1.2.4.2 | 12124220150629072212 | COMPLETE         |
'-----------+---------------+------+------+------------+----------------------+------------------'
```

Please refer to MOS note 1513912.2 for detailed TFA install instructions and how to use TFA. In this short article, I would like to familiarize you with few configurations of TFA.

Once you install the newer version of TFA, it is integrated with the cluster and started automatically every time a node is restarted. The TFA control script on Linux is /etc/init.d/init.tfa. You can manually start, stop, or restart the TFA process.

There are certain tfactl commands that only the root user can execute. tfactl –h shows which commands the current user is allowed to run. You may grant permission for users or groups to run tfactl commands. If the failure is on the database, the DBA can collect the diagnostic information without having root privileges. It is appropriate to run TFA with root privileges, especially when troubleshooting cluster failures.

TFA versions 12.1.2.3.0 and above include several tools integrated with TFA. But for some reason, those tools are not part of the distribution that gets downloaded with GI software or PSU patches. I could not find any such reference in the TFA documentation, so I may be wrong here.

```
# /u01/app/grid/12.1.0.2/bin/tfactl toolstatus
.------------------------.
| External Support Tools |
+-------+-------+--------+
| Host  | Tool  | Status |
+-------+-------+--------+
'-------+-------+--------'
```

I am going to download and install the TFA from MOS. I downloaded TFALite_121250.zip and unzipped to get installTFALite. The uninstall was required because the 12.1.2.4.2 had a more recent build date than 12.1.2.5.0.

```
# /u01/app/grid/12.1.0.2/bin/tfactl uninstall
# ./installTFALite -tfabase /u01/app/grid/12.1.0.2/tfa -javahome /u01/app/
grid/12.1.0.2/JRE
```

```
# /u01/app/grid/12.1.0.2/bin/tfactl toolstatus
.-----------------------------------------.
|         External Support Tools          |
+-----------+-------------+---------------+
| Host      | Tool        | Status        |
+-----------+-------------+---------------+
| zildb806b | alertsummary | DEPLOYED     |
| zildb806b | exachk      | DEPLOYED      |
| zildb806b | ls          | DEPLOYED      |
| zildb806b | pstack      | DEPLOYED      |
| zildb806b | orachk      | DEPLOYED      |
| zildb806b | sqlt        | DEPLOYED      |
| zildb806b | grep        | DEPLOYED      |
| zildb806b | summary     | DEPLOYED      |
| zildb806b | prw         | NOT RUNNING   |
| zildb806b | vi          | DEPLOYED      |
| zildb806b | tail        | DEPLOYED      |
| zildb806b | param       | DEPLOYED      |
| zildb806b | dbglevel    | DEPLOYED      |
| zildb806b | darda       | DEPLOYED      |
| zildb806b | history     | DEPLOYED      |
| zildb806b | oratop      | DEPLOYED      |
| zildb806b | oswbb       | RUNNING       |
| zildb806b | changes     | DEPLOYED      |
| zildb806b | events      | DEPLOYED      |
| zildb806b | ps          | DEPLOYED      |
'-----------+-------------+---------------'
```

As you can see in the output, a large number of tools are integrated into TFA, and OSWatcher is automatically enabled and running. All other tools are in deployed status, and you may invoke them with the tfactl command. Please read the user guide in MOS note 1513912.2.

Some of the tools integrated with TFA are familiar to you; the one I like to use even when there is no failure on the system is orachk. In my opinion every DBA must run orachk or exachk (for Exadata systems) at least every quarter and review/remediate the findings. If you have Oracle E-Business Suite database deployed, orachk also performs a few EBS data checks and advises which patches fix those issues . . . nice!

The standalone tools integrated with TFA are oswbb (OS Watcher—collect OS vital statistics periodically), prw (Process Watcher—collect stack traces of process), oratop (Top Oracle sessions similar to OS top command), SQLT (SQLTXPLAIN – diagnose poorly performing SQL statements). There are several commands available in tfactl that help you to get a summary or the status of a component, instead of a DBA writing scripts or using OS commands. For example, to summarize ALL alert log errors on a given system, you can use the alertsummary utility. Other such utilities are ls, pstack, grep, summary, vi, tail, param, changes, ps, and grep. The param utility shows the value of a database parameter or OS parameter matching the input.

```
# /u01/app/grid/12.1.0.2/bin/tfactl param shm

Output from host : zildb806b
------------------------------
kernel.shmmax = 4398046511104
kernel.shmall = 4294967296
kernel.shmmni = 4096
vm.hugetlb_shm_group = 0

Output from host : zildb806a
------------------------------
kernel.shmmax = 4398046511104
kernel.shmall = 4294967296
kernel.shmmni = 4096
vm.hugetlb_shm_group = 0
```

The alertsummary output is something useful to schedule as a daily cron job for DBA review.

Usually the default repository size is good enough, but if you have space available it may be appropriate to increase the repository size; thus more incident history is kept. Usually you collect diagnostic information from TFA to send to Oracle Support. You may enable automatic diagnostic collection for any failure event; thus the logs and trace files are captured as and when a failure occurs.

ORACHK

Though I mentioned orachk earlier under TFA, orachk definitely requires its share of attention, and every DBA must run orachk periodically. This tool gives you a good insight into the cluster health, database health, Oracle Enterprise Manger health, and EBS application health. The newer versions of orachk also include Collection Manager, a companion Application Express web app, which provides a single dashboard view of collections across multiple database systems. For details on orachk or collections manager, please read MOS note 1268927.2.

MOS Notes

As a DBA, you must be familiar with the various tools provided by MOS for diagnostics and health check. Here are few important ones everyone must be familiar with, which will definitely help you when everyone else in the organization is jumping on you to resolve a failure or troubleshoot a performance issue.

- 1513912.2 – TFA Collector
- 1268927.2 – ORACHK
- 301137.1 – OS Watcher
- 215187.1 – SQLT
- 1500864.1 – oratop
- 459694.1 – procwatcher

In addition to the above tools, MOS provides several troubleshooting tools, such as ORA-600, ORA-1578, ORA-2730x, ORA-3137, ORA-4030, ORA-4031, ORA-7445, and RAC Node Eviction. All of these tools are documented in MOS note 559339.1 Diagnostic Tools Catalog.

Happy troubleshooting! ▲

Biju Thomas is an Oracle ACE, Oracle Certified Professional, and Certified Oracle Database SQL Expert. He is a principal solutions architect at OneNeck IT Solutions with more than 20 years of Oracle DBA experience. He is the author of Oracle 12c and 11g OCA, and co-author of Oracle 10g, 9i, and 8i OCP certification guides published by Sybex/Wiley. He is a frequent presenter at Oracle conferences and publishes articles for technical journals. Twitter @biju_thomas. Blog **www.bijoos.com**.

Fourth International NoCOUG SQL Challenge

In an interview for the *NoCOUG Journal* (**http://www.nocoug.org/Journal/NoCOUG_Journal_200608.pdf#page=4**), Steven Feuerstein was asked: *"SQL is a set-oriented non-procedural language; i.e., it works on sets and does not specify access paths. PL/SQL on the other hand is a record-oriented procedural language, as is very clear from the name. What is the place of a record-oriented procedural language in the relational world?"*

Steven replied: *"Its place is proven: SQL is not a complete language. Some people can perform seeming miracles with straight SQL, but the statements can end up looking like pretzels created by someone who is experimenting with hallucinogens. We need more than SQL to build our applications, whether it is the implementation of business rules or application logic. PL/SQL remains the fastest and easiest way to access and manipulate data in an Oracle RDBMS, and I am certain it is going to stay that way for decades."*

To prove Steven correct, NoCOUG has held four international SQL challenges.

First International NoCOUG SQL challenge (2009)

An ancient 20-sided die 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 represents a number. He also discovered that the die was biased: that is, it was more probable that certain numbers would be displayed than others if the die was used in a game of chance. The great wizard recorded this information in tabular fashion as described below.

Name	Null?	Type
FACE_ID	NOT NULL	INT
FACE_VALUE	NOT NULL	INT
PROBABILITY	NOT NULL	REAL

The great wizard then invited all practitioners of the ancient arts of Es-Cue-El to create an Es-Cue-El spell to display the probabilities of obtaining various sums when the die was thrown "N" times in succession in a game of chance.

The contest was a great success; nine solutions were found by participants in seven countries and three continents but the winner, Alberto Dell'Era from Italy, rose above the competition by implementing Discrete Fourier Transforms and becoming the first knight of the August Order of the Wooden Pretzel. You can read an explanation of his wonderful solution at **http://www.nocoug.org/Journal/NoCOUG_Journal_200908.pdf#page=14**. Alberto also implemented Fast Fourier Transforms but we won't even go there.

Second International NoCOUG SQL Challenge (2011)

An ancient manuscript titled "Love Your Data" was discovered in the secret chamber of mystery at Hogwash School of Es-Cue-El. The manuscript was covered with mysterious words and the great Wizard of Odds implored contestants to create an Es-Cue-El spell that revealed the secret message. Here is a short excerpt from the ancient manuscript.

	A	
COMPREHENSION	ABILITY	OLD
	ABOUT	
	ALWAYS	
SCIENCE	AND	PHYSICS
	ANY	
	AS	
SO	ASK	ABILITY

Andre Araujo (Australia), Rob van Wijk (Netherlands), and Ilya Chuhnakov (Russia) submitted solutions and became the second, third, and fourth knights of the August Order of the Wooden Pretzel. Ilya submitted two solutions: one using the MODEL clause and one using recursive common table expressions. You can read their wonderful solutions in the 100th issue of the *NoCOUG Journal* (**http://www.nocoug.org/Journal/NoCOUG_Journal_201111.pdf#page=20**).

Third International NoCOUG SQL Challenge (2012)

The Wicked Witch of the West had invited six friends to the Third Annual Witching & Wizarding Ball at Pythian Academy of Es-Cue-El & No-Es-Cue-El. Burdock Muldoon and Carlotta Pinkstone both said they would come if Albus Dumbledore came. Daisy Dodderidge said she would come if Albus Dumbledore and Burdock Muldoon both came. And so on and so forth. The Wicked Witch of the West needed an Es-Cue-El or No-Es-Cue-El spell to determine whom she needed to persuade to attend the wizarding ball in order to ensure that all her invitees attended.

Master sorcerer Lukasz Plata of Poland not only solved the problem with a single SQL statement but provided a proof that his solution was correct. He became the fifth knight of the August Order of the Wooden Pretzel. You can read his wonderful solution at **http://www.nocoug.org/Journal/NoCOUG_Journal_201211.pdf#page=12**.

Fourth International NoCOUG SQL Challenge (2015)

The fourth challenge was published in the May 2015 issue of the *NoCOUG Journal* (**http://www.nocoug.org/Journal/NoCOUG_Journal_201505.pdf#page=22**) and was quite unlike the first three challenges. The first three challenges required the contestants to devise an efficient mathematical algorithm and then implement it using SQL. For example, the winning solution to the first challenge used advanced mathematical techniques called "convolutions" and "Fourier transforms." This is not a typical use of SQL in the real world. Also, contestants in the first three challenges sent their entries directly to NoCOUG (though they were free to publish their entries on their own websites or blogs).

The fourth challenge on the other hand did not require contestants to be expert mathematicians. Instead the challenge was an exercise in logic and in interpreting a "functional specification." This is closer to the uses of SQL in the real world. Also, contestants posted their solutions on the NoCOUG blog (**https://nocoug.wordpress.com/2015/05/11/fourth-international-nocoug-sql-challenge/**). This meant that all the contestants benefited from the work of others.

The challenge was a disguised variant of the problem called "Cheryl's Birthday" that was featured in the 2015 Singapore and Asian Schools Math Olympiad for 14-year-old students.

Albert and Bernard just became friends with Cheryl, and they want to know when her birthday is. Cheryl gives them a list of 10 possible dates:

May		15	16			19
June				17	18	
July	14		16			
August	14	15		17		

Cheryl then tells Albert and Bernard separately the month and the day of her birthday respectively.

- **Albert:** I don't know when Cheryl's birthday is, but I know that Bernard doesn't know too.
- **Bernard:** At first I don't [sic] know when Cheryl's birthday is, but I know now.
- **Albert:** Then I also know when Cheryl's birthday is.

So when is Cheryl's birthday? (**https://en.wikipedia.org/wiki/Cheryl%27s_Birthday**)

NoCOUG tried to disguise the problem by using characters from Shakespeare's play Romeo and Juliet. Albert became Romeo and Cheryl became Juliet.

Once upon a time, Romeo, the son of Montague, told his cousin Benvolio that he was in love with Rosaline but she was not returning his affections. Benvolio sang a song by the great American songwriter Stephen Foster:

"There are plenty of fish in the sea
As good as ever were caught."

Meanwhile, Count Paris, a relative of Prince Escalus, asked for the hand of Juliet, daughter of Capulet, in marriage. Capulet organized a grand feast and invited Count Paris. Juliet agreed to talk to Count Paris at the feast. Benvolio suggested that Romeo gatecrash the feast so that Romeo could meet other women. Romeo agreed, but only because Rosaline would also be at the feast. At the feast, Romeo instantly fell in love with Juliet and completely forgot about Rosaline. Romeo then sang another song by maestro Stephen Foster.

"I dream of Juliet with the light brown hair,
Borne, like a vapor, on the summer air;
I see her tripping where the bright streams play,
Happy as the daisies that dance on her way."

Very melodramatic. The solution is iteratively obtained by applying each clue in turn:

- The first clue has two parts: Albert (who has been told the month by Cheryl) cannot uniquely determine the day at this stage in the game (Clue 1a) and knows that Bernard (who has been told the day by Cheryl) cannot uniquely determine the month at this stage in the game (Clue 1b). By applying Clue 1a, we (the public) can eliminate all months which only contain a single candidate day (there are no such months in the sample data above) and, by applying Clue 1b, we can eliminate all months which contain a candidate day that is unique (because if Cheryl's birthday occurred in such a month, then there remains a possibility that Bernard could determine the month at this stage in the game if the day given to him by Cheryl was unique). Clue 1a does not help us here but Clue 1b allows us to eliminate all days in May and June from contention.
- The second clue (Clue 2) is that Bernard (who has only been told the day by Cheryl) is able to use Clue 1a and Clue 1b to uniquely determine the month. We (the public) still don't know Cheryl's birthday, but, by applying Clue 2, we can eliminate July 14 and August 14 from contention because, if either one of them was Cheryl's birthday, then Bernard would not have been able to uniquely determine the month. Only July 16, August 15, and August 17 are left in contention.
- The third clue (Clue 3) is that Albert (who has only been told the month by Cheryl) is able to use the previous clues (and hence knows that only July 16, August 15, and August 17 are in contention) to uniquely determine the day. We (the public) can therefore eliminate August 15 and August 17 from contention because Albert would not have been able to uniquely determine the day if Cheryl's birthday had been in August (since August contains two candidate days that are still in contention).

This leaves only July 16 in contention. Cheryl's birthday must be on July 16.

Assuming that the data is stored in a table called Dates with a single column called DateOfBirth, the above exercise in logic can be expressed in SQL as follows:

```
select
  m, d
from (
  select
    m, d,
    count(*) over (partition by m) as m_count
  from (
    select
      m, d,
      count(*) over (partition by d) as d_count
    from (
      select
        m, d, m_count,
        min(d_count) over (partition by m) as min_d_count
      from (
        select
          m, d,
          count(*) over (partition by m) as m_count,
          count(*) over (partition by d) as d_count
        from (
          select distinct
            extract(month from dateofbirth) as m,
            extract(day from dateofbirth) as d
          from dates
          )
        )
      )
      -- Clue 1a and Clue 1b
      where m_count > 1 and min_d_count > 1
    )
     -- Clue 2
    where d_count = 1
  )
  -- Clue 3
  where m_count = 1
```

The original goal specified in the challenge announcement was an ANSI-standard SQL query of minimum length. However, NoCOUG did not list the candidate dates and required that the solution be able to process all data sets that fit the rest of the story. Perhaps because Cheryl's Birthday was a well-known problem, perhaps because contestants were influenced by the work of others, or perhaps because of sheer subtlety, all the contestants initially missed Clue 1a. The initial set of solutions were therefore incorrect. Since the logic of the solution was now an open secret, the original goal of minimum length was discarded. Chris Goerg from Germany then submitted the following correct solution using the MODEL clause.

```
with d as (
  select unique
    extract(month from dateofbirth) m,
    extract(day from dateofbirth) d
  from dates
)
select m, d from (
  select * from d
  model
  dimension by (m, d)
  measures(0 s, 0 t)
  rules (

    -- Count the number of times each day is duplicated
    -- Store the count in s

    s[m,d] = sum(1)[m, cv()],

    -- Apply Clue 1a and Clue 1b
    -- Set t to 1 if a date is still in contention

    t[m,d] = case
             when min(s)[cv(), d] > 1 and sum(1)[cv(), d] > 1
             then 1
             end,

    -- Apply Clue 2
    -- Count the number of times each day is duplicated
    -- Store the result in s
    -- Only dates with s = 1 remain in contention after this point

    s[m,d] = case
             when t[cv(), cv()] = 1
             then sum(t)[m, cv()]
             end,

    -- Apply Clue 3
    -- Count the number of dates still in contention in each month
    -- Store the result in t
    -- Only dates with s = 1 and t = 1 remain in contention after this point

    t[m,d] = sum(case when s = 1 then 1 end)[cv(), d]
  )
)
where s = 1
and t = 1
```

The secret of Chris's solution is his use of *two* measures (s and t), not just one. Both measures are initialized with the value 0. We can visualize the progress of his solution with a little PIVOT magic:

```
select * from (
with d as (
  select unique
    extract(month from dateofbirth) m,
    extract(day from dateofbirth) d
  from dates
)
select * from (
  select * from d
  model
  dimension by (m, d)
  measures(0 s, 0 t)
  rules (
    -- Add rules here
  )
)
)
pivot (min(nvl(to_char(s),'-')||'|'||nvl(to_char(t),'-'))
for d in (14,15,16,17,18,19))
order by m;
```

The following tableaus show the progression of his solution. In the initial tableau, both s and t are set to 0.

M	14	15	16	17	18	19
5		0|0	0|0			0|0
6				0|0	0|0	
7	0|0		0|0			
8	0|0	0|0		0|0		

Next, count the number of times each day is duplicated and store the count in s.

M	14	15	16	17	18	19
5		2|0	2|0			1|0
6				2|0	1|0	
7	2|0		2|0			
8	2|0	2|0		2|0		

Next, apply Clue 1a and Clue 1b. Set t to 1 if a date is still in contention.

M	14	15	16	17	18	19
5		2|-	2|-			1|-
6				2|-	1|-	
7	2|1		2|1			
8	2|1	2|1		2|1		

Next, apply Clue 2. Count the number of times each day is duplicated and store the result in s. Only dates with s = 1 remain in contention after this point.

M	14	15	16	17	18	19
5		-|-	-|-			-|-
6				-|-	-|-	
7	2|1		1|1			
8	2|1	1|1		1|1		

Next, apply Clue 3. Count the number of dates still in contention in each month. Store the result in t. Only dates with s = 1 and t = 1 remain in contention after this point.

M	14	15	16	17	18	19
5		-|-	-|-			-|-
6				-|-	-|-	
7	2|1		1|1			
8	2|2	1|2		1|2		

Chris was judged the winner on the grounds of novelty and originality. He wins an Apple Watch Sport and becomes the sixth knight of the August Order of the Wooden Pretzel. ▲

Why I Drive 1000 Miles Round Trip to NoCOUG Conferences

by Stephen Van Linge

My name is Stephen Van Linge and I've volunteered to be the membership director for NoCOUG. I actually live in San Diego and drive about 1000 miles round trip to attend NoCOUG conferences since the Oracle user groups in my area are considerably less active, to say the least.

I majored in mathematics in college but found my niche in the database world when I discovered the power of SQL and set operations. After finding a job in the database development and architecture team at a small company, I found it difficult to think of new ways to improve our database environment, query response times, and data model. I decided to join a user group and narrowed it down to NoCOUG and RMOUG.

Despite the awesome "ski days" . . . err "training days" that RMOUG organizes in February every year, I felt it best to get my training in smaller quarterly chunks at NoCOUG than to attend one big yearly event at RMOUG. Not to mention that it was more feasible to drive north to the San Francisco bay area in sunny California than it was to drive to Colorado at the height of winter in a 4-cylinder 2-wheel-drive vehicle.

Transitioning from a classroom setting to a conference setting took a little bit of effort, but it was well worth it. After attending my first conference, I came back to work and told my boss *"It was great; I learned a lot,"* but then I realized that I hadn't retained the great ideas that I had heard because I hadn't taken notes. I took copious notes at the next two conferences and constantly asked myself the question *"This is a really cool idea; can I think of a way to use this or a small-scale proof of concept?"* I compiled more than fifteen project ideas from each conference and rated them by risk factor (projects requiring research and uncertainties versus projects I knew that I could accomplish quickly).

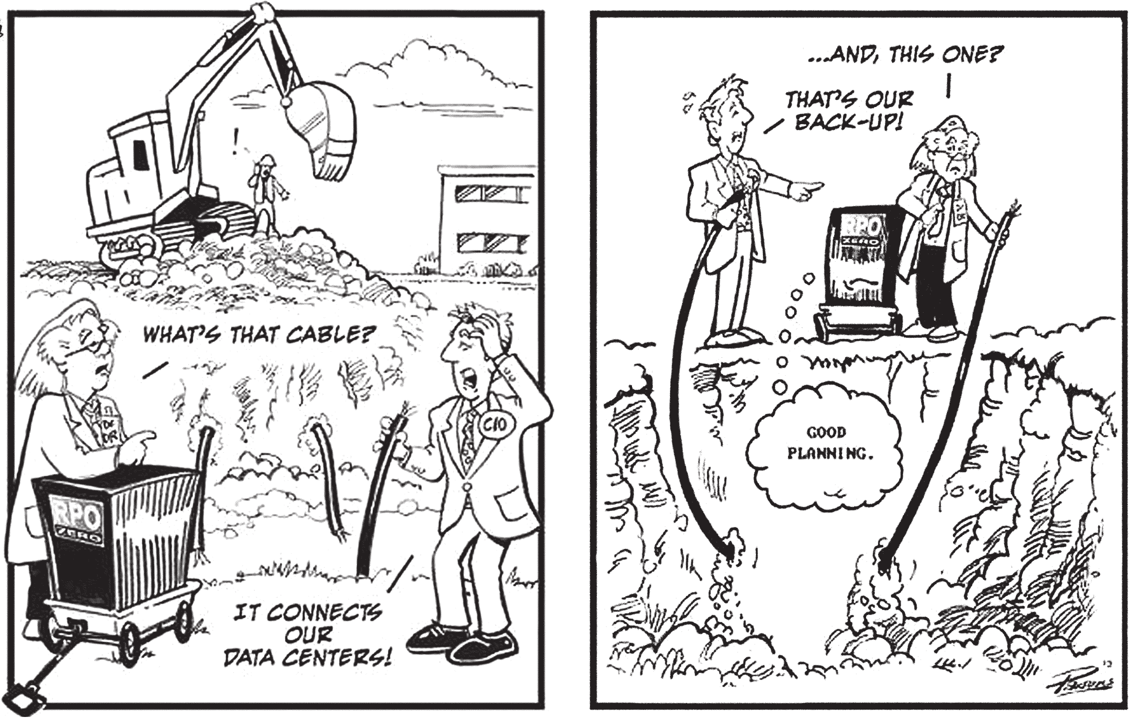
The joint conference with the BIWA SIG in January of this year was a real bonanza, netting me close to thirty new project ideas because the joint conference included presentations on business intelligence (BI) and data science. There was also a tour of the Oracle campus where I got to see the campus highlights including the magnificent Americas Cup-winning yacht on display in "Larry Lagoon."

I'm very interested in meeting new people and I hope to meet you at the summer conference on August 20 at Chevron in San Ramon. NoCOUG has outdone itself once again with a spectacular conference agenda including two presentations on Oracle Database internals. When was the last time you heard a presentation on Oracle Database internals by the Oracle development folks themselves? If you have any membership-related questions or concerns, please send an e-mail message to **membership@nocoug.org** and I'll be glad to help. ▲

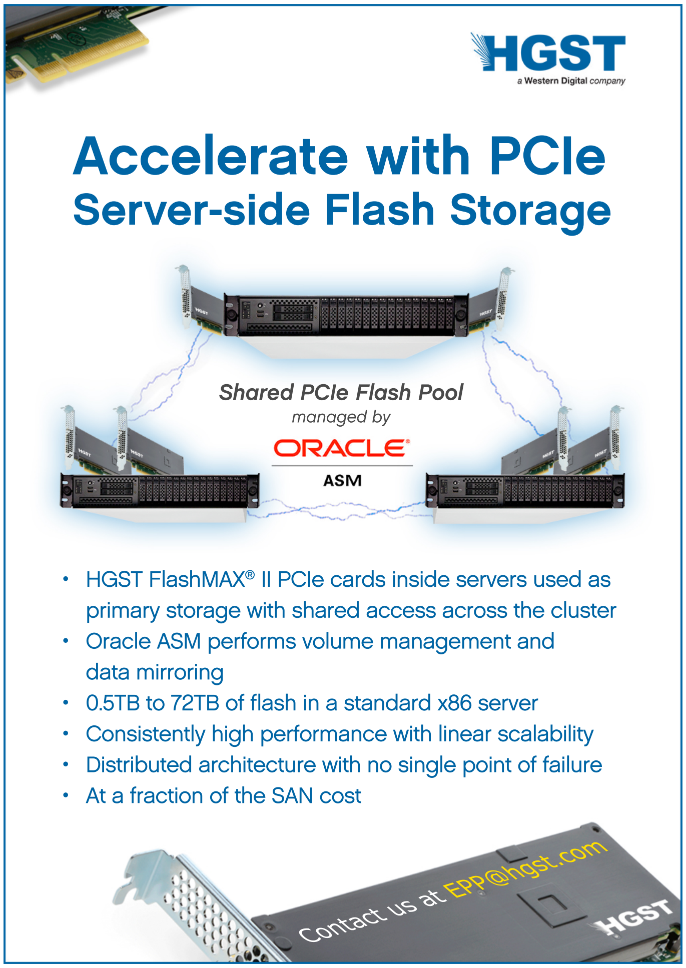
Group picture at Larry Lagoon during the winter conference. I'm the tall guy in the dark-blue T-shirt in the center of the back row.

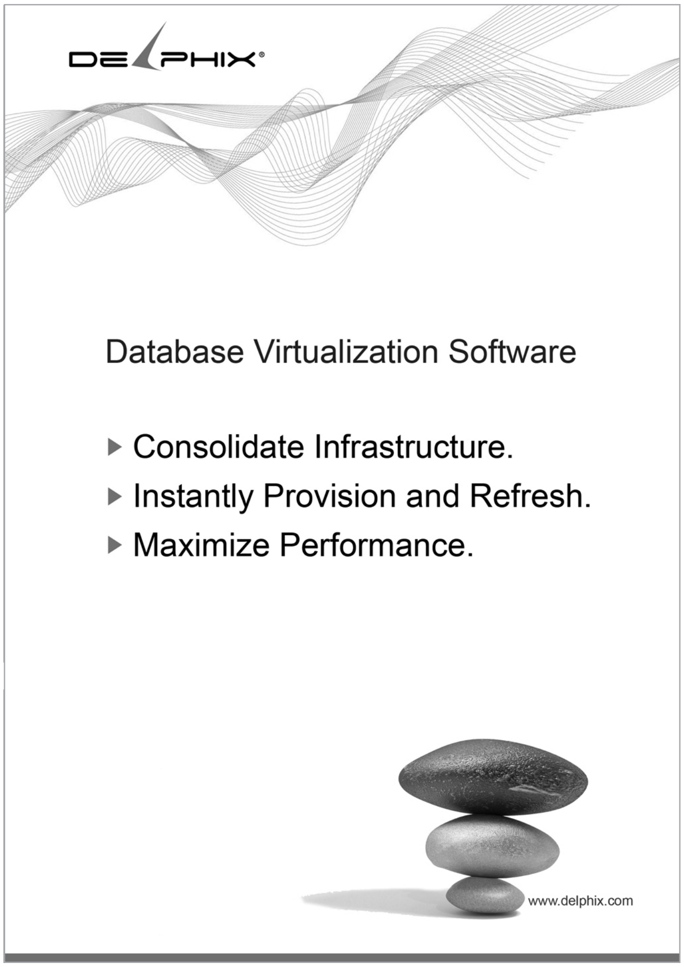
Dr. DR

by Rich Parsons



Dr. DR is brought to you by Axxana.





NoCOUG Summer Conference

Session Descriptions

For the most up-to-date information, please visit **http://www.nocoug.org**.

–Keynote–

The Automotive Data Landscape
Adam Galper, CTO, Xtime 9:30–10:30

From the design lab to the scrap heap, a vehicle generates a large amount of business data, touching wholesalers, retailers, lenders, insurers, service centers, and consumers. In the past few years, vehicle telematics has emerged as a new source of data, providing insight into driver behavior and vehicle health. With the advent of robotic, self-driving capabilities, the amount of data used and generated by vehicles will explode. This talk will provide a survey of the automotive data landscape, exploring relationships and opportunities, with a particular focus on the data technologies required to drive the ecosystem.

–Room 1220–

Lightweight REST Approaches to Data Access
Adam Galper, CTO, Xtime 11:00–12:00

As application development languages and frameworks inevitably change over time, the data access layer for a long-serving database like Oracle typically must adapt as well. After building software for years on homegrown ORM and code-intensive data access layers, we have switched to a lightweight approach to data access, in which tables, views, and stored functions are securely exposed through simple REST interfaces. This approach has dramatically simplified application development, simplifying our code in the database and largely erasing the need for any service layer code. We will describe the approach taken at Xtime and compare it to a few other approaches, including ORDS.

The Database Scaling Saga at Intuit: New Learnings and Techniques for Infinite Scale—*Tushar Thakker, Chief Data Architect, Intuit and Narayanan Gopalakrishnan, Intuit* 1:00–2:00

Scaling a database on any platform is one of the most difficult challenges in today's environment. This is especially true when the database needs relational features. There are many architectural patterns that have been conceived and tried over the years to solve the problem of database scale. For example, architects have used replication patterns, but with this approach there is full downtime, up to 100%, while your primary database switches to the failover database. Another example is to use "beefier" machines to host your database, but this approach presents various hardware limits with CPU and memory, and can be an expensive route. A third example is to have a cluster of database servers (e.g., Oracle RAC), with each server automatically cooperating with the others. But this option, too, has uncovered its own set of issues, limits, and complexities. Therefore, in this session we will learn how Intuit is using a new approach utilizing a swimlane architecture along with SOA to scale our core offering with the intended outcome of getting to scale infinitely.

Understanding and Resolving Free Buffer Wait Contention
Craig Shallahamer, OraPub 2:30–3:30

The free buffer wait event requires the perfect storm to manifest. As performance analysts, our job is to disrupt the storm and return order to our Oracle systems. This presentation focuses squarely on identifying and resolving free buffer waits. We start by developing a clear understanding of the relevant Oracle internals and then develop solutions by focusing on tweaking Oracle, the application (including the SQL), and the I/O subsystem. This is a very practical yet deep internals presentation, filled with amazing discoveries about how Oracle works.

Practical Queuing Theory: An Introduction for Oracle Developers and DBAs—*Craig Shallahamer, OraPub* . . 4:00–5:00

From driving in traffic to being served at a restaurant, every person feels the impact of queuing theory. Queuing theory beautifully relates time and work into terms we can feel, like utilization, workload intensity, response time, elapsed time, and systems architecture design. With only the basics, we can use queuing theory to derive targeted performance solutions, filter and evaluate any performance solution thrown at us, and help non-technical people understand why our solutions make sense. Join us for a shockingly practical session that will impact your Oracle career and beyond.

–Room 1240–

Alternatives to Streams and GoldenGate Replication
Susan Wong, Dell Software Group 11:00–12:00

Now that Oracle is no longer working to improve Streams and change data capture (CDC), check out the powerful, easy-to-use, low-cost alternative: SharePlex. In this session, you'll see how SharePlex delivers real-time data access for BI, ODS, and reporting; reliable high availability and disaster recovery; zero-impact online migrations, patches, and upgrades; in-flight data integrity; automated compare and synchronize utilities; and management interfaces for monitoring and alerting.

AWR Ambiguity: What to Do When the Numbers Don't Add Up—*John Beresniewicz, Teradata Aster* 1:00–2:00

Recently I was asked by a well-known and highly respected performance expert why it is that sometimes the top waits in AWR (Top 10) sum greater than 100 when "waits" for DB CPU are high. The question took me somewhat by surprise, so I asked to see the AWR report in question and sure enough, the section on "Top 10 Foreground Events" indicated that 127% of DB Time was accounted for by three wait events and DB CPU (not a wait event but part of DB Time.) Luckily this was a 12c AWR, which includes information derived from ASH to complement the

(continued on page 26)

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, Hanan Hit. ▲

Long-term event sponsorship:

CHEVRON

ORACLE CORP.

Thank you! Gold Vendors:

- Axxana
- Database Specialists
- Dell Software
- Delphix
- HGST
- SolarWinds

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

$ TREASURER'S REPORT

Ganesh Sankar Balabharathi, *Treasurer*

Beginning Balance		
April 1, 2015		**$ 65,957.11**
Revenue		
Individual Membership	1,615.00	
Gold Vendor Fees	2,000.00	
Corporate Membership	550.00	
Silver Vendor Fees	2,000.00	
Conference Sponsorships	0.00	
Conference Walk-in Fees	450.00	
Training Day Receipts	0.00	
Journal Advertising	1,000.00	
Interest	1.41	
Total Revenue		**$ 6,616.41**
Expenses		
Conference Expenses	7,481.46	
Journal Expenses	1,446.09	
Training Day Expenses	0.00	
Board Expenses	721.20	
PayPal Expenses	185.27	
Software Dues	1,169.94	
Insurance	0.00	
Office Expenses	0.00	
Meetup Expenses	0.00	
Taxes and Filings	600.00	
Marketing Expenses	0.00	
Total Expenses		**$ 11,418.69**
Ending Balance		
June 30, 2015		**$ 61,154.83**

largely counter-based data from Time Model and V$SYSSTAT. In this presentation we will look at this AWR and discuss what to believe and what to suspect when performance data seems ambiguous, and in the process disambiguate this particular AWR report to understand what was really going on.

Database Internals: Transaction Management Concepts

Vivekanandhan Raja, Oracle Corp. 2:30–3:30

This session presents fundamental concepts of the physical-logical model of Undo, transaction lifecycle, isolation models, read consistency, transaction atomicity, and transaction recovery.

Database Internals: Advanced Compression Concepts

Vineet Marwah, Oracle Corp. 4:00–5:00

Oracle Advanced Compression provides a comprehensive set of compression capabilities to help improve performance and reduce storage costs. It allows organizations to reduce their overall database storage footprint by enabling compression for all types of data: relational (table), unstructured (file), network, Data Guard Redo, and backup. Although storage cost savings and optimization across servers (production, development, QA, test, backup, etc.) are often seen as the most tangible benefits, additional innovative technologies included in Oracle Advanced Compression are designed to improve performance and reduce costs for all components of an IT infrastructure, including memory and network bandwidth as well as heating, cooling, and floor-space costs.

–Room 1150–

Enterprise Features of MySQL—Part I: Security

Sastry Vedantam, Oracle Corp. 11:00–12:00

MySQL is the number one open-source database for both enterprise and embedded applications. In this presentation you will learn about MySQL Enterprise Firewall and how it will reduce your vulnerability to data hacks; MySQL Enterprise Encryption to protect confidential data and comply with regulatory requirements; MySQL Audit to implement stronger security controls and satisfy regulatory compliance, including HIPAA, Sarbanes-Oxley, and the PCI Data Security Standard; and MySQL Enterprise Backup to perform online and secure your backups.

Enterprise Features of MySQL—Part II: High Availability

Sastry Vedantam, Oracle Corp. 1:00–2:00

In this session we will discuss different high availability solutions and scalability solutions and how they will fit into the enterprise application HA solutions space. These solutions include MySQL replication; MySQL Fabric; MySQL with shared storage using different OS-level clustering solutions; and MySQL Cluster, a true five 9s HA solution with in-memory database.

Flash Disk Alternatives for Oracle Databases

Roye Avidor, HGST 2:30–3:30

Flash disk and/or SSD technology options have been creeping into Oracle database deployments more and more. Oracle ACE Bert Scalzo will present the many and significantly different alternatives for utilizing flash disk and/or SSD in Oracle deployments. Moreover, Oracle 11*g* features for best utilizing such flash will be reviewed. Knowing what options are out there and how they differ is key, because the flash revolution is here to stay, and not all flash is created the same. ▲

(continued from page 16)

BOOLEAN datatype, which represents only two states: TRUE or FALSE. In the Oracle Database and Oracle Forms PL/SQL, a BOOLEAN datatype has three states: TRUE, FALSE, or NULL. Figure 4 shows the use of an extended "=" operator, namely the *std.eqls* method, that understands tri-state BOOLEAN. Without this, the re-platformed PaaS app would lead to unexpected behavior in conditional processing. Note that this particular example applies to *any* modern PaaS app (with underlying Java metadata) that utilizes Oracle-specific data types, not just re-platformed Oracle Forms/Reports apps.

Concluding Remarks

When re-platformed to PaaS, customers expect that their strategic apps run identically in spite of fundamentally different architecture. As application complexity increases—in number of screens, intricacy and size of code, distributed state across the database, application server, and client tiers—a transformative change will not be adopted if the re-platformed app is not transparent to end users. The NeoWorks development team is hard at work solving technical challenges of extreme compatibility without compromise for mature Oracle Forms and Oracle Reports apps re-platformed to PaaS architecture and our early customers have gone live on our FusionPaaS framework. ▲

Sri Rajan is the founder of NeoWorks, a Silicon Valley company that has built unrivaled expertise in re-platforming Oracle Forms and Oracle Reports. Sri worked at Oracle Corporation from 1990–1998, beginning in the tools division under Sohaib Abbasi, the creator of Oracle Forms. Twitter @neoworksinc. LinkedIn neoworks-inc-. Website **www.neoworks.io**.

In Memoriam

Karen Renee Morton
October 2, 1964–July 23, 2015

She lived well, laughed often, and loved much; she gained the respect of intelligent men and women and the love of little children; she filled her niche and accomplished her task; she left the world better than she found it.

Database Specialists: DBA Pro Service

DBA PRO BENEFITS

- *Cost-effective and flexible extension of your IT team*
- *Proactive database maintenance and quick resolution of problems by Oracle experts*
- *Increased database uptime*
- *Improved database performance*
- *Constant database monitoring with Database Rx*
- *Onsite and offsite flexibility*
- *Reliable support from a stable team of DBAs familiar with your databases*

CUSTOMIZABLE SERVICE PLANS FOR ORACLE SYSTEMS

Keeping your Oracle database systems highly available takes knowledge, skill, and experience. It also takes knowing that each environment is different. From large companies that need additional DBA support and specialized expertise to small companies that don't require a full-time onsite DBA, flexibility is the key. That's why Database Specialists offers a flexible service called DBA Pro. With DBA Pro, we work with you to configure a program that best suits your needs and helps you deal with any Oracle issues that arise. You receive cost-effective basic services for development systems and more comprehensive plans for production and mission-critical Oracle systems.

DBA Pro's mix and match service components

Access to experienced senior Oracle expertise when you need it
We work as an extension of your team to set up and manage your Oracle databases to maintain reliability, scalability, and peak performance. When you become a DBA Pro client, you are assigned a primary and secondary Database Specialists DBA. They'll become intimately familiar with your systems. When you need us, just call our toll-free number or send email for assistance from an experienced DBA during regular business hours. If you need a fuller range of coverage with guaranteed response times, you may choose our 24 x 7 option.

24 x 7 availability with guaranteed response time
For managing mission-critical systems, no service is more valuable than being able to call on a team of experts to solve a database problem quickly and efficiently. You may call in an emergency request for help at any time, knowing your call will be answered by a Database Specialists DBA within a guaranteed response time.

Daily review and recommendations for database care
A Database Specialists DBA will perform a daily review of activity and alerts on your Oracle database. This aids in a proactive approach to managing your database systems. After each review, you receive personalized recommendations, comments, and action items via email. This information is stored in the Database Rx Performance Portal for future reference.

Monthly review and report
Looking at trends and focusing on performance, availability, and stability are critical over time. Each month, a Database Specialists DBA will review activity and alerts on your Oracle database and prepare a comprehensive report for you.

Proactive maintenance
When you want Database Specialists to handle ongoing proactive maintenance, we can automatically access your database remotely and address issues directly — if the maintenance procedure is one you have pre-authorized us to perform. You can rest assured knowing your Oracle systems are in good hands.

Onsite and offsite flexibility
You may choose to have Database Specialists consultants work onsite so they can work closely with your own DBA staff, or you may bring us onsite only for specific projects. Or you may choose to save money on travel time and infrastructure setup by having work done remotely. With DBA Pro we provide the most appropriate service program for you.

DatabaseSpecialists

CALL 1-888-648-0500 TO DISCUSS A SERVICE PLAN

NoCOUG
P.O. Box 3282
Danville, CA 94526

RETURN SERVICE REQUESTED