ORACLE TECHNOLOGY LUNCH AND LEARN TOPICS

When Oracle technology sits at the core of your business, storing critical enterprise data, and is key to your infrastructure, these lunch and learn sessions are a very powerful means of gaining additional insight into how to get the most out of your technology investments. Learn more about increasing operational efficiencies, driving down costs, and securing vital data. Sessions can be provided on or off-site and are adaptable to the targeted audience.

Active DataGuard
Gain an understanding of how Active DataGuard provides significant business benefit. See a demonstration of Active DataGuard in action.

Identity Management for the 21st Century
Learn about business benefits and appropriateness of Oracle's Identity Management solution, including Oracle Identity and Access Management Suite plus components including OID, Virtual Directory, Oracle Access Manager, etc.

Minimizing Risk of Database Changes
Gain a thorough understanding of how Oracle's Real Application Testing can mitigate the risk of upgrades and new feature adoption.

Business Intelligence: An Overview
Oracle offers a complete pre-integrated technology foundation that reduces the cost and complexity of building and deploying enterprise Business Intelligence. Find out how to increase operational efficiency, track effectiveness and identify new revenue streams.

Are You Compliant?
Hear about the benefits of Data Vault, Audit Vault, data masking and other Oracle compliance solutions. Get the "checklist" of security compliance considerations.

All About Content Management
Experience the capabilities and business benefits of Oracle's Enterprise Content Management solutions.

Real Application Clusters
Oracle has made significant improvements to its Real Application Clusters (RAC) technology. This overview highlights the capabilities and benefits of Oracle's Real Application Clustering and why it has moved to the mainstream.

Enterprise 2.0
WebCenter Services 11g provides a comprehensive set of standards-based components that enrich existing portals and Web sites with the industry's most complete and open set of Enterprise 2.0 capabilities. Learn how services like social computing services, personal productivity services, online awareness and communications, content integration, and Web analytics can differentiate your business.

A Primer on Oracle SOA
Gain an understanding of Oracle's SOA offerings — SOA Suite, Business Process Manager and Service Bus — and their benefits to today's business climate.

Purging and Archiving: How?
Hear about the benefits, challenges and solutions surrounding data purging and archival. Gain an understanding of how advanced strategies can facilitate automation and efficiency.

11g Database New Features
Understand the new features of 11g and how they can be utilized to improve performance, manageability, reliability and scalability. Learn about the upgrade process from 9i or 10g.

Stimulus Plan for Rapid Application Development
Learn how Oracle tools such as Application Express (APEX) and BI Publisher Builder can facilitate Rapid Application Development.

Compression: An Overview
Learn about Oracle's various compression features, along with the business, cost and technical benefits of compression.
contents

features

6 Function-Based Indexes  
by Donald Burleson  
Tuning SQL

10 Oracle & Excel  
by Mark Molnar  
Why Fight It?

16 Index Monitoring  
by Dan Holka  
Oracle 9+ Can Monitor Index Usage

18 The Case for CASE  
by Mathias Magnusson  
Extend The Power of SQL

departments

22 RMOUG Membership Focus  
Marimer Panganiban

24 RMOUG Board Focus  
Brad Blake

29 Training Days 2010  
Time To Register

monthly features

4 Letter From The President
5 Stan Yellott Scholarship Fund
26 Membership Application
27 Advertising Rates
28 RMOUG Board of Directors
30 RMOUG Calendar of Events
31 May 2010 QEW
31 Index To Advertisers

On the Cover:
Shrouded in clouds is Vermejo Peak (13,723’), located 7 miles north of New Mexico in the Sangre de Cristo mountains. Vermejo Peak is Colorado’s 135th highest peak. The picture was taken by Tom Peters in November 2004 while hiking nearby Culebra Peak (14,047’). Tom is an Oracle developer currently working for Pinnacol Assurance in Denver, CO.
One of the nice things about the holidays is that you can help people forget the past with a present. And we know this past year has been very challenging for many of our members, so the RMOUG Board of Directors has several presents for you ….

Our first gift came a bit early. It started on December 3rd, but not to worry. It will return each month throughout the coming year – this is our “Database Lab Meetup Series”. RMOUG has partnered with Regis University School of Computer & Information Sciences (SCIS) to provide all of you an opportunity to learn with actual hands-on practice. Each meeting features hands-on labs using the latest Oracle database technologies, products, features, and utilities at a local Regis campus. Format includes: lecture, lab, Q&A, and attendee networking.

Our second gift is RMOUG Training Days 2010 conference scheduled for February 16 – 18, 2010 that promises to be an event you will not want to miss. Our Training Days committee has scheduled over 120 technical sessions with 84 speakers coming from around the world, including ten Oracle Ace Directors and six Oracle Aces.

Here are some of our conference highlights:

• “University Sessions” are back by popular demand for Tuesday afternoon and for details about these half day in-depth sessions, just go to our Training page on the web site.

• “Breakfast with the Oracle Ace Directors” will be our opening session on Wednesday with an informative panel discussion.

• “OTN Campground” will be in the Exhibit Hall. So don’t miss your opportunity to talk with the Oracle Experts; bring your questions/issues/comments to the Experts at the OTN (Oracle Technology Network) sponsored Campground.

• “InstallFest is back again!” Here is your opportunity to learn how to install Enterprise Linux and configure it for installing Oracle 11gR2 Database, SQL Developer, APEX, Fusion Middleware, and Oracle VM.

Our committee recognizes that budgets have been tight, so with that in mind they have kept the early registration fee for the two day conference at $240.00 for RMOUG members. For this low fee you get a full two days of learning and networking opportunities. Save even more when you register for both the Training Days Conference and a University Session. You will automatically receive a 10% discount on your University Session registration fee of $125.00. So take advantage of early registration at the RMOUG web site and don’t forget to mark your calendars for February 16 – 18, 2010.

Our third gift is coming in March 2010. RMOUG is partnering with ISACA, formerly known as the Information Systems Audit and Control Association, a nonprofit organization that is dedicated to serving the needs of IT governance professionals to provide a joint workshop addressing the application development process and the security, auditing governance issues involved. More details about the workshop will be posted to our web site soon.

And one final gift is my personal wish for everyone -- that this coming year brings you peace of mind, good health, prosperity, happiness that just keeps multiplying, fun around every corner, and the confidence to achieve your dreams!

Have a wonderful and safe holiday…
The Rocky Mountain Oracle User’s Group (RMOUG) is dedicated to helping others. We give our time and put forth effort not for financial gain, but to improve the professional, technical and personal lives of our colleagues. Stan Yellott personified this dedication by helping not only his professional colleagues, but by extending the effort to high school and college students, the next generation of IT professionals. The Stan Yellott Scholarship Fund is founded on the principle of going above and beyond to assist our current and future colleagues.

RMOUG started to award annual college scholarships to high school students. The Stan Yellott Scholarship Fund continues the educational mission by assisting deserving students.

Scholarships of $1,000 each are awarded semi-annually in November and March to students interested in pursuing studies related to Information Technology.

By contributing to the Stan Yellott Scholarship Fund, you are joining with RMOUG and the Information Technology Community to assist deserving students to achieve their IT educational goals. For contributions of $100 or more, you will receive a commemorative Tie-Dyed T-Shirt in honor of Stan.

Applications for scholarship awards may be downloaded at www.rmoug.org

If you are interested in contributing to the scholarship fund, please contact Heidi Kuhn
HeidiKuhn@rmoug.org

Congratulations to our most recent recipient

Kyle Schultz

From Kyle’s Application

In the year 2002 I had the fortune of participating in a midnight highway accident. While I lost my car, job and educational funding as a result, I was blessed with the opportunity to re-evaluate my life and goals. During the following interleave I came to understand the importance of a solid education while struggling through various desktop publishing jobs. It took living in the real world for me to truly understand what was at stake.

I have learned and seen first hand the purpose and value of a degree. After more than five years of job experience in the field, I am ready to finish my masters degree, only this time with a newfound sense of urgency. I have ambitious goals of applying a Master of Computer Science to my artistic and multimedia orientation in the entertainment industry and only a financial need hinders me.

Using my professional experience and my previous schooling, I know I am going to be successful in the pursuit of a Master of Computer Science and my multimedia programming career endeavors!

Kyle came highly recommended by Scott Strong, an instructor at the Colorado School of Mines, Department of Mathematical and Computer Sciences. Mr. Strong describes Kyle as an intelligent, mature student who is committed to understanding both the concepts and mechanics of the course material. He describes Kyle’s attendance as impeccable and believes Kyle will be the type of student who is unafraid to seek out interesting interdisciplinary work. Per Mr. Strong,

My expectation is that when Kyle Schultz completes his undergraduate degree he will possess a unique total package of talents, which will make him highly adaptable and marketable in technological industries. I am excited to see his full development as a student and the outcome of his academic training. It is important to support such a unique student and to do so, I HIGHLY RECOMMEND Kyle Schultz to the Stan Yellott Scholarship Fund.

“We can never replace all the great things you do but we will help carry on all the things you started”
George T
Tuning SQL with Function-based Indexes (FBI)

by Donald Burleson

With each release of Oracle we see dozens of new features, and it’s hard to understand that tuning features are right for you. For SQL tuning, the function-based indexes ranks among one of the most powerful Oracle features.

When function-based indexes (FBI’s) were quietly introduced in Oracle 8i, many Oracle professionals still failed to fully utilize function-based indexes as one of the single most powerful Oracle tuning tools.

Oracle DBA’s known that there are many “silver bullet” techniques, single changes that have a profound impact on the entire SQL workload. These include any global setting that affects the execution plans of SQL statements:

- Parameter settings (optimizer_mode, optimizer_index_cost_adj)
- New CBO statistics
- New patches/ new releases
- Indexes (especially function-based indexes)
- Materialized views

Out of these powerful tools, the most readily available is the adding of a “missing” index, especially a function-based index.

If the goal of an index is to fetch the desired row-set with a minimum of databases “touches” (e.g. consistent_gets), then a function-based index is a Godsend because anything that you see in the where clause of a SQL statement may have a matching function-based index.

In almost all cases, the use of a built-in function like to_char, decode, substr, etc. in a SQL query may cause a full-table scan of the target table. For example, consider this query:

```sql
-- select March sales
select sum(sales) from mysales
where to_char(order_date, 'MON') = 'MARCH';
```

In this case, without a specific index on (to_char(order_date, 'MON'), the above query will be forced to read every row in the sales table, even though an index already exists on the order_date column.

To avoid this problem, many Oracle DBAs will create corresponding indexes that make use of function-based indexes. If a corresponding function-based index matches the built-in function of the query, Oracle will be able to service the query with an index range scan thereby avoiding a potentially expensive full-table scan.

The following is a simple example. Suppose that you have identified a SQL statement with hundreds of full-table scans against a large table with a built-in function (BIF) in the where clause of the query. After examining the SQL, it is simple to see that it is accessing a customer by converting the customer name to uppercase using the upper BIF.

Running the explain plan utility confirms your suspicion that the upper BIF is responsible for an unnecessary large-table full-table scan.
The table access full customer option confirms that this BIF is not using the existing index on the `customer_name` column. Since a matching function-based index may change the execution plan, a function-based index can be added on `upper(customer_name)`.

It can be risky to add indexes to a table because the execution plans of many queries may change as a result. This is not a problem with a function-based index because Oracle will only use this type of index when the query uses a matching BIF.

```sql
create index upper_cust_name_idx
on customer
(upper(customer_name))
tablespace customer_idx
;
Exec dbma_stats.gather_index_stats('scott', 'upper_cust_name_idx');
```

Now, the SQL can be re-explained to show that the full-table scan has been replaced by a index range scan on the new function-based index.

For this query, the execution time has been decreased from 45 seconds to less than two seconds.

```
create index case_index
as
(case SOURCE_TRAN
when 'PO' then PO_ID
when 'VOUCHER' then voucher_id
else journal_id
end = '0000000001'
end);
```

As we now know, even though the `ship_date` column has an index, the `trunc` and `to_char` built-in functions will invalidate the index, causing sub-optimal execution with unnecessary I/O. However, in these cases you can re-formulate the where clause to make the `ship_date` index work, without having to create a function-based index:

```
-- Will not use ship_date index
where trunc(ship_date) > trunc(sysdate-7);
-- Will not use ship_date index
where to_char(ship_date,'YYYY-MM-DD') = '2004-01-04';
```

```
-- Uses ship_date index
where ship_date >= trunc(ship_date-7) + 1;
-- Uses ship_date index
where ship_date = to_date('2004-01-04','YYYY-MM-DD');
```

Remember, whenever possible you always want to use an existing index before creating a function-based index.

**Using Case With a Function-Based Index**

Using function-based indexes (FBI) you can create an index on any built-in function, including a case expression. Here we use case within the create index syntax:

```
create index case_index
as
(case SOURCE_TRAN
when 'PO' then PO_ID
when 'VOUCHER' then voucher_id
else journal_id
end = '0000000001'
end);
```

Once created, you need to create CBO statistics, but beware that there are numerous bugs and issues when analyzing a function-based index.
As a final step, run the execution plan for the query and ensure that your SQL with `case` is using the appropriate index.

Index on Complex Functions

After considering the power of the function-based index we might ask if there is any limit to the ability to index into complex expressions. Jonathan Gennick shows a great example where we use Oracle regular expressions to extract “acreage” references from inside a text string, ignoring important factors such as case sensitivity and words stems (acre, acres, acreage):

The only problem with this query is that it will always perform a large-table full-table scan on the `michigan_park` table, causing unnecessary overhead for Oracle.

However, using the powerful function-based indexes we could eliminate the unnecessary overhead by using the regular expression directly in the index.

The rules for choosing a function-based index on a complex expression (regular expression, decode) is a trade-off between several factors:

- **The number of blocks in the table** - A full-table scan of a super-large table can cause I/O contention.
- **The percentage of rows returned** - If the regular expression returns only a small percentage of the total table rows, a regular expression index will greatly reduce I/O.
- **The frequency of the query** - If the query is executed frequently, Oracle may do millions of unnecessary full-table scans.
- **The tolerance for slower row inserts** - Parsing the text column at insert time (to add the row to the regular expression index) will slow-down inserts.

It’s the age-old quandary. If we build the regular expression once (at insert time) it can be used over-and-over again with little overhead. Conversely, using regular expressions in SQL without a supporting index will cause repeated full-table scans.

Statistics and Function-Based Indexes

A common issue with Oracle DBA’s and developers is “Why does Oracle not using my function-based index?”

Because a function-based index applies a function to a data column, special steps must be taken for the optimizer to be able to utilize a function-based index. The Oracle documentation notes that collecting statistics (and histograms) is an important prerequisite:

“You should analyze the table after creating a function-based index, to allow Oracle to collect column statistics equivalent information for the expression. Optionally, you can collect histograms for the index expressions by specifying for all hidden columns size
Here are some methods for analyzing a function-based index, but beware that they may not work on all releases of Oracle for reasons noted below:

```sql
-- Gather index stats for function-based index
exec dbms_stats.gather_index_stats('OWNER', 'CASE_INDEX');

-- Gather table stats for function-based index
exec dbms_stats.gather_table_stats(
    ownname=>null,
    tabname=>'CASE_TAB',
    estimate_percent=>null,
    cascade=>true,
    method_opt=>'FOR ALL HIDDEN COLUMNS SIZE 1');
exec dbms_stats.gather_table_stats(
    ownname => 'OWNER',
    tabname => 'CASE_TAB',
    cascade => TRUE);
```

Check For Bugs

As noted, function-based indexes are “tricky” to get working, and you must carefully verify your SQL execution plans to ensure that they are working before migrating them into production.

There are reports of function-based indexes not being used by the CBO, some due to bugs and others due to issues with the `dbms_stats` analysis routine.

- See bug 2782919, titled “GATHER_TABLE_STATS FAILS ON TABLE WITH FUNCTION-BASED INDEX”.

Conclusions On Function-Based Indexes

In sum, function-based indexes are one of the true “Silver Bullet” techniques, a single action that can have a profound impact on your entire SQL workload. There are several points to remember about function-based indexes:

- A single FBI might optimize hundreds of SQL statements, reducing system-wide I/O.
- Function-based indexes can match anything in a where clause, even derived data like that from a decode statement.
- It’s important to collect CBO stats (and possible histograms to ensure that the FBI is used properly by the optimizer.

Even though using a function-based index is tricky, it is well worth the effort since the function-based indexes allow you to go directly to the target data blocks that contain the rows that you desire.

About the Author

Donald K. Burleson is one of the best-known names in Oracle technology, a full-time DBA with over 25 years of full-time experience working on some of the world’s largest and most complex databases. Author of more than 30 books, including the bestselling “Oracle Tuning: The Definitive Reference”, Burleson runs www.db-a-oracle.com, America’s largest independent source of Oracle information and www.remote-dba.net, a remote DBA Oracle provider.
Overview

I used a product in the early 90’s called, “Oracle For 123.” It was an application from Oracle Corp. allowing data to be retrieved directly to Lotus 123, the spreadsheet application of choice by businesses at the time. Not that Lotus still isn’t popular software (now owned by IBM), but Microsoft Excel seems to have taken the front seat in most businesses. I believe that businesses have enjoyed, and continue to enjoy, access to their data in big centralized databases. However, they still want to use their spreadsheet applications for their own flavor of analysis, crunching numbers, reporting, charting, and a host of other features. I venture to say this trend will most likely continue.

I know, it’s hard to hear from an Oracle point-of-view, but business users love spreadsheets. At some point we Oracle-techies just need to accept and embrace that fact. I’m not saying users don’t like those cool-looking reports that are served up from your state-of-the-art application. However, from time to time, users just want raw data dumps so they can control the data in their beloved spreadsheet program. When you come across this scenario and want to simplify the “data-dump, export, copy, paste, reformat” issues, here are some ways to get data from the database, directly from Excel, into either a spreadsheet or written to files) directly on the user’s workstation.

Introduction

Being Oracle developers, we want to use all the standard tools Oracle provides so the business can do their job more efficiently and get just the data they need. Right? Come on, is anything else really necessary? Being an Oracle technocrat, I used to pride myself on being able to produce any dataset, in any format, with nothing more than some ornery SQL and SQL*Plus column format masks & break points. As Oracle matured, its tool-sets (Oracle Forms, Reports, JDeveloper, Discoverer, and a host of “cube” crunching technologies), we (the IT folks) welcomed them and were able to provide our users the data in new and creative ways, making really pretty, build-it-only-once, and be done with it modules. Right? Not quite. Regardless of how much time spent on those applications, it always seems the datasets would find their way back into Excel on our users’ workstations.

The problem is that a lot of business users like Excel; it’s their comfort zone. They want to manipulate the data, using their own formulas, and see the information the way they want to see it, with all the pretty colors and graphs too! Again, not that IT couldn’t do that with these standard tools, but users want the ability to, in their words, “... do other things with the data too.” Like what? Good question, maybe they don’t trust us with the logic behind their secret macros!

When the time comes to have data put directly onto a user’s workstation, regardless of the reason, it’ll be nice to have a simple solution ready, using a tool in that special “comfort zone.” What’s a good model for this look like? A few points to include:

• Give the user an interface, right in Excel, prompting them for necessary parameters and creating a spreadsheet with the dataset requested.
• Ability to run directly from a user’s workstation and NOT have to maintain local tnsnames.ora files.
• From a security perspective, accept usernames & passwords (optional of course) at the database level so control remains tight with object access.
• And, maintain control of the queries so they are well tuned and don’t kibosh performance.

What follows are some pretty simple steps allowing you to “empower” your user-base to grab their data directly from the Oracle database and bring it into an Excel worksheet, or file, to be used as needed. Brace yourselves. Visual Basic (VB) macros are used to accomplish this! Don’t get turned off yet. VB has some pretty powerful stuff, and if you’ve been coding with PL/SQL or any other language, you should be able to follow this example and syntax pretty easily.

I can’t tell you how embarrassed I would have been if any of my peers caught me in the bookstore picking up a VB book when I had to do something similar for a client – it all turned out okay and I have a new appreciation for some of the VB features. You will too.

Technical Stack

In my examples, below, I essentially used two pieces of software:

1. An Oracle 10g Database.

   Grant yourself access (and create some private synonyms) to the SCOTT tables Dept and Emp if you want to use these examples exactly as defined, below.

   Oracle was running under a GNU/Linux OS:
2. Microsoft Excel

Microsoft® Office Excel® 2007 (although I’ve successfully used Excel 2003 with the same macros; however, some tweaks may be necessary depending on your version and service pack):

Below are two sections that detail how to query the Oracle database from a Microsoft Excel spreadsheet, and:

1. Create a new worksheet, in Excel, with the data results (this article, Part 1 of 2), and
2. Using similar techniques, use Excel as the user interface to create multiple data files that can be used for other purposes (email, archiving, uploading to other external applications, etc. - next article, Part 2 of 2).

Creating an Excel Worksheet

The first thing to do is open Microsoft Excel using a new, blank spreadsheet. Click on the Developer tab [A], choose the “Visual Basic” icon [B] to open up the Visual Basic Editor:

NOTE: If you don’t have the Developer tab showing, right-click the ‘ribbon’ next to the View tab and select ‘Customize Quick Access Toolbar’, select ‘Popular’ in the menu on the left and check the ‘Show Developer tab in the Ribbon’ check box, and then click [OK].

This will launch a new window:

You’ll need to create a new “module” to hold your visual basic (VB) macro code. Simply click “Insert -> Module”:

A new window will appear to the right, that’s more or less a text editor to start coding:

Now it’s time for some code! First, I like to setup a little procedure that I can use throughout the program in case I need to exit for some reason. Simply click in the “Module” window and create the Sub-module (FYI, in VB, lines starting with a single quote are comment lines):

```
' Define Procedure to EXIT if Cancel or ESC is pressed
Sub Back_2_Spreadsheet()
    MsgBox (“Pressed Cancel, Exiting … “)
End ' Cancel out of Program
End Sub
```

The nice thing about using the “MsgBox” syntax is that it gives a pretty nice out-of-the-box, standard pop-up, and waits for the user to click OK before continuing. For example, the “Exiting Program … message-box command, above, will appear as:
The next block of code will define the main sub-module called “Process_Oracle_Request”. Then, it will ask the user which database they want to connect to, DEV or PROD, and store their input in a variable for future use. It will also request to enter the “Extract-Date”, for future use:

```vbnet
Sub Process_Oracle_Request ()
' Seede the Variable with default DB Name
Oracle_DB_Name = "PROD"
' Ask User which DB
Oracle_DB_Name = InputBox(“Enter Database Name (PROD or DEV), "Oracle Database Name", Oracle_DB_Name)
' Check to make sure a value was selected and ESC was not pressed. Pressing ESC or pressing [Cancel] means NULL is returned
If Oracle_DB_Name = "" Then
Back_2_Spreadsheet
End If
' Set Default to Today
Extract_Date = Format(Date, "mm/dd/yyyy")
' Then ask with default entered
Extract_Date = InputBox("Date (format MM/DD/YYYY) or Leave Blank For Today", "Enter Billing Date", Extract_Date)
' Okay to leave blank - default in system date
If Extract_Date = "" Then
Extract_Date = Format(Date, "mm/dd/yyyy")
End If
End Sub
```

The “Input Box” syntax, above, will accept the “Oracle_DB_Name” variable and it uses three parameters (there are more, like X&Y coordinates, if you want to use them, but not needed in this example):

1. Prompt (The information you want to tell the user)
2. Title (The title of the pop-up window)
3. Default Value (shows “PROD” because it was seeded prior to the call)

For example, the “InputBox("Enter Database Name (PROD or DEV), "Oracle Database Name", Oracle_DB_Name)" syntax will cause the following window to appear:

![Input Box Example](image)

Then an “If-Then” statement is executed to check if a value was entered or not. If the value was erased, or the [Cancel] button was pressed, a NULL value gets returned, effectively telling us the user wants to cancel, and go back to the spreadsheet by calling “Back_2_Spreadsheet” procedure previously defined.

Similar to the Database Name, the next parameter asked for is the “Extract Date.” Before the user is asked for the value using the InputBox command, the parameter is seeded with today’s system date, using the VB syntax, “Format(Date, "mm/dd/yyyy").” Then, when asked for the parameter, today’s date will be defaulted as the value. The “If-Then” statement that follows does a little different logic, and represents what was asked for in the InputBox, “Date (format MM/DD/YYYY) or Leave Blank For Today” so if the user clears the value, today’s date will be defaulted back in. This is meant as an example of how to use the logic. You may prefer to exit the program anytime the user chooses “Cancel”, therefore calling the “Back_2_Spreadsheet” procedure as we did above.

Now that the parameters are accepted, the next thing is to declare the connection objects to the database. I found it very simple to use ActiveX® Data Objects (ADO) to set up the necessary connection to the database (FYI, ADO provides updated functionality to DAO and RDO interfaces provided in releases of Excel prior to 2000). The first thing to do is set up some variables with the appropriate object type references:

```vbnet
Set objmyconn = New ADODB.Connection
Set objmycmd = New ADODB.Command
Set objMyRecordset = New ADODB.Recordset

objmyconn.ConnectionString = "Driver={Microsoft ODBC for Oracle}; " & _
"(CONNECT_DATA=(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP) & _
"(HOST=r moug_devbox.r moug_domain.com)(PORT=1522)) & _
"(CONNECT_DATA=(SID=DEV))); uid=scott;pwd=tiger;"

Else
End If
```

Next, determine which database to connect to based on the parameter the user entered:

```vbnet
If Oracle_DB_Name <> "PROD" Then ' Assume DEV
objmyconn.ConnectionString = "Driver={Microsoft ODBC for Oracle}; " & _
"(CONNECT_STRING=(DESCRIPTION=" & _
"(ADDRESS=(PROTOCOL=TCP) & _
"(HOST=r moug_devbox.r moug_domain.com)(PORT=1522))") & _
"(CONNECT_DATA=(SID=DEV))); uid=scott;pwd=tiger;"
Else
End If
```

In the above code, objmyconn.ConnectionString, is getting assigned to a string that looks remarkably similar to a tnsnames.ora entry. Specifics on the VB syntax:

- The “If-Then” statement uses a Not-Equal (<>l) condition to check the value the user entered. In this case, unless the user entered “PROD” as the database name, all other values will default to “DEV”. You can put in more error checking if you’d like, but this is for example purposes.
- The “Driver=” part of this string defines the “Microsoft ODBC for Oracle” driver. This is the only remotely difficult portion of setting this up on a workstation. You simply need to install that driver so it is available to use. I’ll discuss that later in the article.
- The “& _” at the end of the lines is the syntax for continuation characters stating to continue the string on the next line; therefore, the five lines of code are really one long string being stored in the variable.

It’s time to open the database connection and execute a database command, a PL/Sql block in this case:
The code above is an example of opening the database connection defined earlier, setting the active connection, setting the "CommandText" to be run (by supplying the string), and finally executing it. This piece of code shows how to execute a block.

Ok, now it's time to query the database and return the results into our spreadsheet. The intended data will come from our good friend Scott's schema, Dept & Emp. If executed from SQL*Plus the classic query would be:

```
SELECT lpad(' ', 4*(level-1)) || ename quick_org, 
  e.ename, e.job, e.hiredate, e.sal, e.comm, 
  e.deptno , d.dname, d.Loc 
FROM emp e, dept d 
WHERE e.deptno = d.deptno 
CONNECT BY PRIOR e.empno = e.mgr 
START WITH empno = (select empno from emp 
  where emp.job = 'PRESIDENT') ;
```

To make this example work, including the header columns, we need to open a worksheet, add our "header" columns, and then query the database. The code for that:

```
' Open new workbook
Set Wkb1 = Workbooks.Add
Wkb1.Activate

' Create headers in Row 1
Worksheets("Sheet1").Cells(1, 1).Select
ActiveCell.FormulaR1C1 = "Current Org Structure"
Range("C1").Select
ActiveCell.FormulaR1C1 = "Emp Name"
Range("D1").Select
ActiveCell.FormulaR1C1 = "Emp Job"
Range("E1").Select
ActiveCell.FormulaR1C1 = "Emp Hire Date"
Range("F1").Select
ActiveCell.FormulaR1C1 = "Emp Sal"
Range("G1").Select
ActiveCell.FormulaR1C1 = "Emp Commission"
Range("H1").Select
ActiveCell.FormulaR1C1 = "Emp Dept No"
Range("I1").Select
ActiveCell.FormulaR1C1 = "Emp Dept Name"
Range("J1").Select
ActiveCell.FormulaR1C1 = "Emp Location"

' Put Cursor back in A1
Range("A1").Select
```

You'll notice in the query string, I kept the SQL statement similar to what I'd use in SQL*Plus. I find it easier to read that way than having it all on a single line. Also, note there is no semicolon at the end of the SQL statement and is not required for passing the string. Once the "CommandText" is set, the "Execute" takes place.

Now we need to get the “Record Set” back and into the active worksheet:

```
' Open Recordset
Set objMyRecordset.ActiveConnection = objmyconn
objMyRecordset.Open objmycmd

' Copy Data to Excel
ActiveSheet.Range("A2").CopyFromRecordset (objMyRecordset)
```

By using the “objMyRecordset” object, setting it to the current connection and issuing the “Open” command against it will put the data in the RecordSet data buffer. Finally, a “CopyFromRecordset” command is used to copy the record set from the buffer into the spreadsheet, starting at position “A2” (remember the column headings are already in row 1).

Now, simply close the database connection and the submodule:

```
' Close DB Connection
objmyconn.Close

End Sub

'Tell the User that the Extract has been completed.
MsgBox ("Spreadsheet Has Been Created, Press [OK] To Finish.")
```

You'll notice, above, a final “MsgBox” command. I always like to tell the user that the process has been completed so they're not hanging around waiting for something to be done. This ties out the user interface.

Save the macro (Ctrl-S) and it is saved as part of the current spreadsheet. For example, give it name like sample_macro.xlsm.
Running the Macro to Create the New Worksheet

To test out the macro directly from the Visual Basic Editor, press the icon, or simply press F5 (run). Here's what should happen, based on the code above: Press [F5] to execute the macro. Notice that the first time you run this, you will be asked for which Macro you want to run:

Choose the “Process Oracle Request”, then click [Run]. Next, the first parameter box is displayed:

Click [OK] to continue. The next parameter box will be displayed; type in the date for your parameter:

Click [OK] to continue. The new spreadsheet is now getting created and filled in with the data from the database. When that is complete, the final “Message Box” will be displayed:

Click [OK] to continue.

A quick NOTE to those of you running Excel 2003. You may encounter the following error: Compile Error: Can't find project or library.

At this point you should notice a new Excel spreadsheet has been created and is open on your desktop. If you navigate to it, the results, at least for our example, are:

You can now save your new spreadsheet to a location for future processing. You can also automate the “save” feature. I show some naming conventions on how to do this later in the article.

Protect & Deploy – let's not forget that!

We're almost done. I'm a still a believer that supporting the users and giving them this kind of tool should be under IT's control due to performance concerns and support arrangements. So there are two final things that need to be done:

1. Prevent the user from updating (or even seeing) the macro and query, and
2. Give the user a nice user-interface, a “hot-key”, from the spreadsheet, to initiate the macro.

First, let's protect the macro by using a password. From the VB Editor, right click the “Module1” icon, choose the “VBAProject Properties...” option:

Click on the “Protection” tab:

library marked as MISSING: <referencename>.
Click the “Lock Project for viewing” tab, then enter and confirm the password to be used (i.e. “RMOUG”):

![Lock Project for viewing](image)

Click [OK], then save and exit your spreadsheet. Reopen the spreadsheet and go to the VB Editor again. This time when you click on the VBAProject (sample_macro.xlsm) area to open the VB Editor, you’ll be prompted for the password:

![Password prompt](image)

Type in your password (i.e. RMOUG) and press [OK]. Your project will now be opened for editing. NOTE: Passwords are case-sensitive so make sure you document it accurately.

Finally, let’s setup the spreadsheet to invoke the macro based on a hot key which will make it easy for the user to initiate. From the spreadsheet, click the “Macros” icon:

![Macros icon](image)

Choose the macro you want to assign. In this case, “Process_Oracle_Request”:

![Macro selection](image)

Click “Options” and enter a key-character you want to use, in this case “d” making the macro initiate upon [Ctrl+d]:

![Macro options](image)

Click [OK], then press [Cancel] on the “Macro” window to close it. To let the users know what to do when they open this spreadsheet, simply navigate to cell A1 and type, “Press “Ctrl+d” to start Dept and Emp Extract.”:

![Macro prompt](image)

Save the spreadsheet.

You now have a fully functional spreadsheet for the user and the code is secured via a password. All the user has to do is open the spreadsheet and when they’re ready, press [Ctrl+d] to fill in their parameters and produce the new spreadsheet. Distribute to your users! Keep in mind, if you have a shared network drive, you can copy the spreadsheet to that centralized location and replace with any future updates or versions that may be required.

**Microsoft ODBC for Oracle**

Without getting into the step-by-step details, use this link as a reference for installing this driver if it is not already on your workstation: http://msdn.microsoft.com/en-us/library/ms713590(VS.85).aspx.

Check back with us for Part 2 in the Spring issue of SQL>Update.

Mark grew up in the southern part of New Jersey and attended Rutgers University receiving a BA (Computer Science major, Business minor). He took his first Oracle class in 1989 and has been in the field ever since. After a few years on the East coast, then in Texas, Mark came to Colorado (1993) and quickly heard about, and joined, RMOUG. He gave his first “Training Day” presentation in 1995, and has previously served on the RMOUG board in different capacities. His most recent focuses have been with Oracle Applications, the DW/BI area, and integrating imaging solutions. Mark started MODE Technical Solutions, Inc. in 1997 and lives in Morrison, CO with his wife of 15 years and 5 children (11 yrs, 9 yrs, 9 yrs, 9 yrs, and 9 yrs ... no, that’s not a typo!).
Oracle9i has introduced an easy method to see if indexes are even being used by your application.

Amazing as it sounds, I am finding an average of 10 indexes per table object, particularly in larger applications. Oracle probably is not using most of these indexes or only uses them occasionally such as at month end/quarter end/year end processing. These unused indexes are maintained and can make batch cycles run twice/three times as long as needed.

My SQL Performance Tuning class (being offered frequently via the Web...or we can schedule this at your company) goes into great detail on how Oracle uses indexes, how the optimizer decides which index to use, as well as this index-monitoring topic...which is an excerpt from my SQL Performance Tuning course guide.

Oracle9 introduced the ability to tell if an index has ever been used in a SELECT or sub-query. This technology uses the MMON process and has to be tripped on to populate the V$OBJECT_USAGE view.

If the index then appears in a query explain plan, the USED column will be set to ‘YES’, otherwise it is null.

Consider flushing this information when re-running statistics.

Oracle 9+ can monitor Index Usage

- SQL> alter index <index name> MONITORING USAGE
- SQL> alter index <index name> NMONITORING USAGE
- Consider:
  DBMS_STATS.FLUSH_DATABASE_MONITORING_INFO

This script can be used to turn on index monitoring for all indexes for a particular schema owner.

The SELECT statements count various useful statistics.
1. Indexes that were used
2. Indexes that were not used
3. Number of indexes being monitored
4. Number of tables that have the indexes

Eliminating unused indexes is batch cycle tuning. Indexes and table structures have 5 Oracle operations applied to each DML:
1. Lock acquired
2. Before image journalled in rollback segment
3. Actual change applied
4. After image journalled to archive log
5. Lock released

...for each object associated with the DML. If the index is not being used, then it should not be there as it is being maintained by Oracle.

set headings off
spool index_monitoring.sql
select 'alter index '||index_name||' monitoring usage;'
from user_indexes;
spool off
start index_monitoring
select count(*) from v$object_usage where used = 'YES';
select count(*) from v$object_usage where used is null;
select count(*) from v$object_usage;
select count(*) from user_tables;
*** True Story *** A shop in Denver had 98 tables and over 1100 indexes! I tripped on index monitoring and when I left 3 days later, Oracle had not used 100 indexes yet.

*** Tip *** The programmer should use this technique when testing an application and build scripts that create/drop indexes before month end/quarter end/year end processing.

Summary

Index monitoring is a useful tool to eliminate unused indexes that will decrease batch cycle times and decrease insert/update times.

My SQL Performance Tuning class has a very interesting half-day lecture on indexes, how they are created, how Oracle uses them, when they are not used and why, and how 10g+ tablespace options has an adverse effect on index utilization.

Dan Hotka is a Training Specialist who has over 31 years in the computer industry and over 26 years of experience with Oracle products. He is an internationally recognized Oracle expert with Oracle experience dating back to the Oracle V4.0 days. Dan’s latest book is the TOAD Handbook by Pearson. He is also the author of SQL Developer Handbook by Oracle Press, Oracle9i Development By Example, and Oracle8i from Scratch by Que and has co-authored 7 other popular books including the Database Oracle10g Linux Administration by Oracle Press. He is frequently published in Oracle trade journals, and regularly speaks at Oracle conferences and user groups around the world. Visit his web site at www.DanHotka.com. Dan can be reached at dhotka@earthlink.net.

Dan Hotka - Author/Instructor/Expert
www.DanHotka.com
DHotka@Earthlink.net
515 279-3361

Instructor-Led Training At Your Desk!

Classroom style training over the web is available today. This is a new and exciting training format which fits into your busy schedule.

Web-Based Oracle Training by Dan Hotka:
• Morning Lecture Sessions (half days)
• Hands-on Labs
• Cost effective – includes all course materials

Advanced PL/SQL Tips and Techniques
• January 18 – 22, 2010
• April 5 – 9, 2010

SQL Performance Tuning
• March 1 - 5, 2010

See Dan’s web site for course outlines or email Dan with your level of interest. Dan also provides all of his course offerings for your company both over the web or on-site at a low course-fee (on-site course guides priced per student).

Contact Dan at:
www.DanHotka.com
Dhotka@Earthlink.net - (515) 279-3361

REGISTER TODAY!

Take advantage of the Advance Registration Rate. Register on or before January 12, 2010.

CONFERENCE FEATURES:
Oracle ACE Directors Keynote Panel
University Sessions
More than 100 Technical Sessions
Vendor Presentations
Exhibits
Oracle Campground

RMOUG TRAINING DAYS
February 17-18, 2010
Denver, Colorado
Colorado Convention Center

www.RMOUG.org
910.452.0006

THE COLLEGIATE PEAKS - PHOTO COURTESY OF JOHN KING, RMOUG MEMBER

SQL>UPDATE • Winter 2009 17
**Introduction**

In this document we will look closer at the keyword CASE in SQL. While it often is treated just as a readable replacement for decode, it has a few more uses. Readability is a feature in itself, but the most misunderstood part of case among developers is how flexible it is.

This article will only look at the basic use and not try to dig into every aspect of it. Thus it is geared more towards the intermediate SQL user than to the experts.

**Select**

The CASE keyword was introduced in 8i (9i for PL*SQL) and essentially replaces DECODE. I confess to never liking decode, though it is powerful, it is also ugly and sometimes close to unreadable. Case introduces a few things decode cannot do, but it makes it more accessible to those who write SQL occasionally. It is easier to get a case formatted nicely than it is to do the same with decode.

The basic syntax for case is:

```sql
select empno ,ename ,case when sal < 1500 then 'low' when sal between 1500 and 3000 then 'mid' when sal > 2500 then 'high' end sal_class from emp;
```

Here we have classified the people by their salary. Those earning less than 1500 will be tagged as low, those earning 1500-2500 will be mid, and those earning more will be high. This kind of case construct is called searched and is the kind I find most often in SQL. There is however a simpler version that can be used when the comparison is on just one value and against just one value per condition. This has the creative name of simple case.

```sql
select empno ,ename ,case deptno when 10 then 'Dept 10' when 20 then 'Dept 20' else 'unknown' end dept from emp;
```

This basic example just rewrites the value of deptno to have "Dept" in front of it. It shows the use of the simple case. It is often forgotten and rewritten as a searched case even when the simple case would do.

One thing to note here is that the else construct that can be used to give a value when a situation not covered in the conditions is encountered is not mandatory in SQL the way it is in PL/SQL. In SQL it defaults to null while it ends up with an exception in PL/SQL. The last example could be enhanced to use else like this:

```sql
select case when empno  = 7839 then 1 when ename  = 'KING' then 2 when sal    = 5000 then 3 end all_true from emp where empno = 7839;
```

You recognize a simple case on the expression between the case and when keywords. The expression is the department number (deptno). A searched case has nothing between case and when.

In this example, department 30 was not know at the time the SQL was written and it is now showing up as "unknown" when the SQL is executed. Note that the syntax is "else <value>". One common mistake is to write it as "else then <value>". It is easy enough to fix, but Oracle will not help you more than to inform you that you have a missing expression. The time I've lost looking for an error just to find the extra then is embarrassing. It looks correct since all the previous lines has it, but the compiler is uncompromising on the requirement to remove it.

One thing to remember is that multiple conditions may be true for a single row. In this case all three conditions are true. So does it stop after the first true value or does it run through all, and the final result is the result from last true condition?

```sql
select case when empno  = 7839 then 1 when ename  = 'KING' then 2 when sal    = 5000 then 3 end all_true from emp where empno = 7839;
```
The result is 1 and the reason is that it takes the first true value and then leaves the case statement. The above statement also shows that a searched case does not need to have any correlation between the different conditions. This is one of the strengths of case, while simultaneously being one of the most underused features.

**Update**

All the power of case in the select clause can also be used in other parts of the SQL language. For example, it is very common to see an update script that looks something like this.

```sql
update emp
set sal = sal * 1.15
where deptno = 10;

update emp
set sal = sal * 1.05
where deptno = 20;

update emp
set sal = sal * 1.10
where deptno = 30;
```

This is of course not much of a problem in a table with 14 rows, but say that each statement takes an hour or more to complete and updates just a small portion of the whole table. The time to run each statement can then be a problem if the conditions are such that the whole table has to be read each time. The above statements can be rewritten like this:

```sql
update emp
set sal = sal * (case when deptno = 10 then 1.15
                     when deptno = 20 then 1.05
                     when deptno = 30 then 1.10
                     end);
```

**Case** is here used to return the factor to use to adjust everyones salary. I find this easier to read and if there are many update statements it will give a better overview over what changes are intended to be made.

Another situation where case is useful is when there are many columns than needs to be updated, but the condition is not the same for each update.

```sql
update emp
set sal = sal * (case deptno
                     when 10 then 1.15
                     when 20 then 1.05
                     when 30 then 1.10
                     end);
```

Solving this with individual update statements would require eight individual statements if we assume that there could be more departments and more job titles than specifically listed here. There could also be more complicated situations causing more complex situations.

**Where**

Using case in a where is often useful when there are multiple combinations that all needs to be treated as true. This can be especially true if some additional tests should be performed for all such conditions. Consider this example.

```sql
select *
from emp
where case
      when job = 'PRESIDENT' and sal > 4500 then 1
      when job = 'MANAGER'   and sal > 2500 then 1
      when job = 'SALESMAN'  and sal > 1500 then 1
      else 0
end = 1
and sal / 10 = trunc(sal/10)
and ename like '%A%';
```

Here we will retrieve employees that have a salary above a certain threshold per title for just some of the titles an employee can have. For all those we want to make sure the salary can be divided by ten without getting a rest and their name should contain an A. This example may not be very realistic, but it shows the power of using case in the where clause. Note how the condition on the case compares the result of the whole case expression with a value. It may look very strange in the beginning, but it is very useful once you get used to that syntax.

**Order By**

Sometimes sorting needs to be done in a way that neither descending nor ascending will solve. Say that you want to sort the employees such that those in department 20 are listed before those in 10, with those in department 30 coming last, then something like this would do the trick.

```sql
select *
from emp
order by case deptno
      when 10 then 2
      when 20 then 1
      when 30 then 3
      else 4
end;
```
This translates 20 to 1, 10 to 2, 30 to 3, and anything else to 4 and those translated values are then what the rows is sorted by. This can be very powerful when specialized reporting needs have to be supported.

Another case is when you want to treat two groups as if they were one. Say that you want to return all employees ordered by deptno and salary, but department 10 and 30 should be treated as the same. That is employees within those departments should be grouped together and ordered by salary. This SQL shows one way to achieve that.

```
select *
from emp
order by case deptno
  when 10 then 10
  when 30 then 10
  else deptno
end
desc;
```

Note how all other departments just keep their deptno via the else. In fact, the "when 10" line could be removed as it doesn't change the value. I would keep it just for the declarative value in showing exactly what we want to achieve.

**Group By**

Using case when aggregating data can provide some unexpected power. Let us start with a simple case where we report total salary for management, workers, and sales.

```
select case job
  when 'PRESIDENT' then 'MGMT'
  when 'MANAGER' then 'MGMT'
  when 'SALESMAN' then 'SALES'
  when 'CLERK' then 'WORK'
  when 'ANALYST' then 'WORK'
end cat,
sal
from emp
group by case job
  when 'PRESIDENT' then 'MGMT'
  when 'MANAGER' then 'MGMT'
  when 'SALESMAN' then 'SALES'
  when 'CLERK' then 'WORK'
  when 'ANALYST' then 'WORK'
end

having case
  when sum(sal) < 9000 then 1
  when sum(sal) > 11000 then 1
  else 0
end = 1
order by case cat
  when 'PRESIDENT' then 3
  when 'MANAGER' then 3
  when 'SALESMAN' then 1
  when 'CLERK' then 2
  when 'ANALYST' then 2
end;
```

Is this an excessive use of case? Probably, and it could be more maintainable by using subquery refactoring (WITH clause) or an inline view (select in the from clause). That would allow the same functionality without repeating the same thing so many times. However, this article’s focus is to show case and adding in more SQL features would not help understanding the powerful features of case.

**Pivot**

Say that you want to produce a report showing each department on a row with a total salary per title. It can be done with a series of unions and summing the results, but it can be done with a single select using case. Here is one version of this.

```
select deptno
  ,sum(case job
        when 'PRESIDENT' then sal
        when 'MANAGER' then sal
        when 'SALESMAN' then sal
        when 'CLERK' then sal
        when 'ANALYST' then sal
    end) pres_sal
  ,sum(case job
        when 'PRESIDENT' then sal
        when 'MANAGER' then sal
        when 'SALESMAN' then sal
        when 'CLERK' then sal
        when 'ANALYST' then sal
    end) mgr_sal
  ,sum(case job
        when 'PRESIDENT' then sal
        when 'MANAGER' then sal
        when 'SALESMAN' then sal
        when 'CLERK' then sal
        when 'ANALYST' then sal
    end) sls_sal
  ,sum(case job
        when 'PRESIDENT' then sal
        when 'MANAGER' then sal
        when 'SALESMAN' then sal
        when 'CLERK' then sal
        when 'ANALYST' then sal
    end) clk_sal
  ,sum(case job
        when 'PRESIDENT' then sal
        when 'MANAGER' then sal
        when 'SALESMAN' then sal
        when 'CLERK' then sal
        when 'ANALYST' then sal
    end) anl_sal
from emp
group by deptno;
```
We group the data by `deptno` and use one case per title to report just salaries for that title in that column. This gives us a nice report showing how much salary each department spends in each job title.

This is a very powerful feature once you wrap your head around the basic construct. You will start recognizing areas where this technique can be used. You will also be able to stop co-workers from reinventing the wheel when a pivot need comes up.

### Conclusion

This article has reviewed the use of CASE in SQL. It is very powerful and this article has tried to show some of the many flexible ways it can be used to extend the power of SQL.

I hope this article helps you. Please send comments and questions to mathias.magnusson@gmail.com or visit the comment page linked to from http://mathiasmagnusson.com/oracle/articles.html.

---

**Is your company trying to consolidate servers?**

Are you trying to cut licensing costs, reduce power and cooling, free up datacenter floor space and save money?

If the answer is yes to any of these then you should consider a Total cost of Ownership (TCO) study from Paragon Solutions Group. With our proven methodology we can provide a quick, easy to read TCO document that allows you to make intelligent choices moving forward.

---

### IBM vs. SUN SERVERS *

*The Difference is Clear*

<table>
<thead>
<tr>
<th></th>
<th>IBM</th>
<th>Sun</th>
<th>Sun</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>p560</td>
<td>M8000</td>
<td>M9000</td>
</tr>
<tr>
<td>CPU Architecture</td>
<td>POWER6</td>
<td>SPARC64 VI</td>
<td>SPARC64 VI</td>
</tr>
<tr>
<td>Number of Cores</td>
<td>16</td>
<td>32</td>
<td>32</td>
</tr>
<tr>
<td>Processor Speed</td>
<td>3.6 GHz</td>
<td>2.4 GHz</td>
<td>2.4 GHz</td>
</tr>
<tr>
<td>Performance Estimate (RPE2)*</td>
<td>18,100</td>
<td>18,360</td>
<td>17,440</td>
</tr>
<tr>
<td>Memory</td>
<td>64 GB</td>
<td>128 GB</td>
<td>128 GB</td>
</tr>
<tr>
<td>HW &amp; SW Purchase Price (Estimated)</td>
<td>$283,899</td>
<td>$1,037,108</td>
<td>$1,380,990</td>
</tr>
<tr>
<td>HW &amp; SW (Op sys) Support (24x7, 3yrs)</td>
<td>$61,032</td>
<td>$190,659</td>
<td>$234,032</td>
</tr>
<tr>
<td><strong>Sub-Total</strong></td>
<td><strong>$344,931</strong></td>
<td><strong>$1,227,767</strong></td>
<td><strong>$1,615,022</strong></td>
</tr>
<tr>
<td>Oracle Database Cost</td>
<td>$320,000</td>
<td>$640,000</td>
<td>$640,000</td>
</tr>
<tr>
<td>Oracle Support Cost (3 Years)</td>
<td>$211,200</td>
<td>$422,400</td>
<td>$422,400</td>
</tr>
<tr>
<td><strong>Total 3 Year Projected Cost</strong></td>
<td><strong>$876,131</strong></td>
<td><strong>$2,290,167</strong></td>
<td><strong>$2,677,422</strong></td>
</tr>
</tbody>
</table>

*Performance Estimate is the Ideas International Relative Performance Estimate v2 (RPE2). This estimate is operating system neutral. Server, Software, Oracle and Support costs are estimates.*

---

See why over 45 Enterprises have chosen Paragon to help them reduce costs and improve over-all efficiencies.

To find out more, visit www.paragonsg.com or contact John Bussie jbussie@paragonsg.com 303-910-9412

*We Can Do IBM vs HP Comparisons Too*
I received my Bachelors Degree in Information and Computer Science in the Philippines, and a Post Graduate Diploma in Information Technology from UK. I came to the United States in 1997. Prior to coming to the US, I worked as a programmer analyst developing applications using GUPTA SQL Windows and SYBASE. My first job assignment in the US was in Monterey CA, where I was first exposed to Oracle. After eight years of working for a property and casualty insurance company, I decided to gain development experience in the health care industry in 2005, and landed a job with a non-profit organization in Scotts Valley that handles Medi-Cal claims for Santa Cruz county.

I moved with my family to Colorado Springs in 2007, and I am now currently working as a Sr. Applications Developer for INGENIX – a United Health Group company. I belong to a team called the Managed Gateway Deployment, and concentrate mostly on Oracle PL/SQL development to support processing of health care (medical and dental) claims of our vendors and payers.

**My First Exposure to Oracle**

Back in my home country (Philippines), my first exposure to a database was DBIII+ (offered in college) and FoxBase when I started working for a consulting company. I was heavily involved in maintaining the mail list and mail merging. The experience was not that exciting, and it was really plain and simple.

In 1993, I got a real programming job as a programmer trainee, and I was exposed to SYBASE 8. I wrote queries, procedures, and functions to support the front-end application developed in GUPTA SQL Windows. GUPTA had a dynamic library which made it easy to access the database while developing the GUI application.

In 1997, I was hired and sponsored by a US employer to work as a consultant for an insurance company in Monterey, CA. The company is in the property and casualty insurance business. The company’s goal at that time was to convert their legacy system to an Oracle database which gave me the opportunity to use and learn about Oracle. At first, I was involved in developing a front-end application for the claims department using GUPTA SQL Windows. After this project, I was involved in the conversion. I was a little bit hesitant, since I was coming from a SYBASE background. My IT manager affirmed that it is easy, it still uses SQL, and I would only encounter a few minor syntax problems. That was true, but it also takes getting acquainted with SQLPlus, learning all the other commands specific to Oracle, understanding the Oracle structure, and its dictionary tables.

My initial task was to develop SQL scripts that would populate tables considered as ADMIN (administration data) due to the data being CONSTANT,
and were used to support the business rules. At that time, I worked a lot with the DBA who was in charge of the data modeling. He gave me instructions on what tables to populate and where to get the data from. I never saw the data model when I started working on the SQL scripts. So, I asked for it. Oh boy, after seeing the data model, I had a better picture of how the tables were related. I believe that understanding the data, and the business, is a very important factor in every database development and administration.

Due to the amount of data, the plan was to convert one business at a time. The business refers to the type of insurance such as homeowners, personal auto, commercial, etc. This wasn’t a one year project. It took three years to finish the conversion.

During the conversion process, there was great emphasis on performance. The conversion was planned to be executed during the weekend to avoid downtime for the users and the business. There were only a few hours devoted to the actual data load, and all of it was automated from UNIX scripts. There were other processes that had to run before and after the conversion such as back-ups, and having QA look at the application.

This was a very exciting moment for me. The company hired a consultant from Oracle to show us tricks on how we could speed-up our PL/SQL codes. I learned how to run our code in a parallel mode using the rowids as a range, commits at a certain number of rows, what kind of cursors performed faster, disabling/enabling constraints before and after, etc. This was in Oracle 8i, so there were no bulk collections yet. The conversion was not just a simple insert; it involved computations in some areas, and a great deal of understanding of the parent and child relationship. I had to make sure that the GUI application was able to pull data from the tables based on the business rules defined on the front-end.

We had dress rehearsals on weekends. My PL/SQL codes were scheduled to run on a test server to get the actual timing. I worked a lot with the DBAs since they were developing the automation through the UNIX scripts, and they monitored the execution time as well. Anything more than an hour was not acceptable. However, exceptions were based on the amount of data loaded. The QA people were also busy with testing the new GUI application, and they had to deal with new and converted data.

After the conversion project, I had other projects that dealt with database security and database administration. One of my favorite projects was to change the database date to a past or future date. The reason for this is that policies have a maturity date. The QA people had test scenarios that are based on the maturity dates. For example, an auto policy would mature in six months. QA would not, of course, wait for six months to see the results. Changing the dates on a server level causes problems. One of my colleagues stated that it was not possible to change the date on a database level. Well, he was wrong. You actually can. I approached it with changing one of the database date parameters, and developed a UNIX script to change the time every minute. It worked great; problem solved.

Oracle is a great and powerful database. You can do a lot with it. It just depends on how you use it, and when the challenge or opportunity comes. I had a great learning experience with my previous projects, and really developed my PL/SQL programming skills.

Tell Us About Yourself

Join us in sharing your Oracle experiences with other RMOUG members! Tell us about your life, your job, or share your amusing Oracle anecdotes, tips and secrets!

Please submit all material to NewsletterDir@rmoug.org or phone Pat Van Buskirk (303) 621-7772
My name is Brad Blake and I am currently the SIGS Director (Special Interest Groups) for RMOUG. Here’s a little background on myself.

I grew up in Evergreen, CO – my parents still live in that general area which allows for my family to visit the mountains as often as we like. After graduating from Evergreen High School, I headed to Colorado State University in Fort Collins – GO RAMS! At CSU, I majored in Biological Sciences (pre-med coursework) with a minor in Spanish, and of course enjoying what college life had to offer. After graduating with my undergrad degree, I spent a few years working in the medical field, to see if I wanted to further pursue a career in medicine. I decided that wasn’t the path I wanted to follow, particularly because I wasn’t interested in another eight plus years of schooling. At the time I had several friends that had come out of CSU with Computer Science degrees. It seemed like they were all doing well and enjoying their jobs, plus the type of work they were doing sounded interesting to me, so I decided to head back to school. I went to Regis University and got a Masters Degree in Computer Information Systems, with an emphasis in Database Technologies.

When I first started the program at Regis, I really didn’t know much of anything about databases. I’ll never forget my first database course – it was a real eye-opener for me, and when we started using SQL to do simple queries and updates, everything really started to click. I also remember one of my instructors pulling up a monster.com job search on the projector, showing all of the open positions for database-related careers. I was sold – I took all the database classes offered at the time, and before I graduated from Regis, I started looking for my first “Junior DBA” job.

I was fortunate to quickly find a great job at a small dot-com company in downtown Denver, where I started out as a Junior DBA. It was an excellent learning opportunity for me, working under a Senior DBA who taught me how things work in the “real world”. After about four and a half years there, I started to feel like things were reaching a plateau, and was ready for a new challenge. I found a job as a DBA for a small pharmaceutical company based in Boulder, called Pharmion Corporation. This was a fantastic job – not only was it an excellent company to work for but before they hired me, they didn’t even have a full-time DBA on staff. I was basically able to mold my own position. With this came great opportunities, but also some struggles. One of the more challenging tasks was trying to learn how to manage a DB2 database – I migrated this over to Oracle fairly quickly! They also had an existing application, written in Oracle Forms, that was hosted in a datacenter in the UK, with a DataGuard standby site in Germany. The database
and application were developed by outside consultants, so I became the primary support person for this. I was fortunate to get experience with other software – I installed and implemented a worldwide Business Objects environment, primarily using Crystal Reports. With that, I also became the primary Crystal Report writer. Other applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForce.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage included JD Edwards for financials, SalesForge.com for applications I helped manage include...
Application For Membership

☐ Individual...$75  ☐ Student (must have Student ID)...$35

Name ____________________________________________
Company Name _____________________________________
Address __________________________________________
City, State, Zip _____________________________________
Phone ___________________________________________
Fax _______________________________________________
E-Mail ___________________________________________

Payment Method     Total $ ________________________

☐ Check/Money Order (Make payable to Rocky Mountain Oracle User’s Group)
☐ Visa Card # ________________________________
☐ MasterCard# ________________________________

Name on Credit Card __________________________________

Corporate:

☐ 1-5 Members ________________________________ $300
☐ 6-8 Members ________________________________ $450
☐ 9-12 Members ______________________________ $595

Additional members (over 12) $50 each

Please attach list of all members.

Corporate Contact:
Company Name _____________________________________
Contact Name _______________________________________
Address __________________________________________
City, State, Zip _____________________________________
Phone ___________________________________________
E-mail ___________________________________________

Please mail completed form and payment to:

Rocky Mountain Oracle Users Group
PO Box 621942
Littleton, CO 80162

FAX: (303) 933-6603

What Is RMOUG?

The Rocky Mountain Oracle Users Group (RMOUG) was established in 1984 with just a few members. Meetings were held twice-a-year to share ideas and information about Oracle. Today, RMOUG is one of the largest Oracle user groups in the world with over 1,000 members.

RMOUG offers general membership meetings, a professional magazine, an annual training event, and an information-packed web site. Members include professional analysts, project managers, database administrators, developers, and designers who work with Oracle products to produce high-quality business solutions.

RMOUG is an alliance partner with the International Oracle Users Group - Americas. RMOUG is a not-for-profit organization incorporated in Colorado.
Reach A Targeted Oracle Audience
Advertise Now!

A full page, full color ad in RMOUG SQL>UPDATE costs as little as 70 cents per printed magazine and even less for smaller ads.

RMOUG SQL>Update Advertising Rates

<table>
<thead>
<tr>
<th>Type</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>Business card or 1/8 page</td>
<td>$50</td>
</tr>
<tr>
<td>1/4 page</td>
<td>$350</td>
</tr>
<tr>
<td>1/2 page</td>
<td>$625</td>
</tr>
<tr>
<td>Full page</td>
<td>$1,000</td>
</tr>
<tr>
<td>Inside cover</td>
<td>$1,250</td>
</tr>
<tr>
<td>Back cover</td>
<td>$1,500</td>
</tr>
</tbody>
</table>

Discounts available for RMOUG Members and Full Year Contracts

See Media Kit for deadlines and mechanical requirements.
Submit ad materials to: Pat Van Buskirk, RMOUG Newsletter Director
38101 Comanche Creek Road • Kiowa, CO 80117
303-621-7772 • E-mail: NewsletterDir@rmoug.org • www.rmoug.org

SQL>UPDATE is mailed to over 1,000 RMOUG Members and distributed during Quarterly Education Workshops and Training Days each year
Meet Your Board

Peggy King, King Training
President
Business: (303) 798-5727
Business: (800) 252-0652
E-mail: President@rmoug.org

John Peterson
Vice President
E-mail: VicePresident@rmoug.org

Bill Schwartzkopf
Secretary
E-mail: Secretary@rmoug.org

Kathy Robb
Treasurer
Arisant, LLC
E-mail: Treasurer@rmoug.org

Ron Bich, SofTec Solutions, Inc.
Training Days Director
Phone: 303-650-6951
E-mail: TrainingDaysDir@rmoug.org

Barbara A. Lewis
Membership Director
Axia College of the University of Phoenix
Voice: 303-757-6709
Email: MembershipDir@rmoug.org

Allison Leech
Programs Director
E-mail: ProgramsDir@rmoug.org

Pat Van Buskirk
Newsletter Director
National Radio Astronomy Observatory
Voice: 303-621-7772
Email: NewsletterDir@rmoug.org

Brad Blake
SIGS Director
E-mail: SIGSDir@rmoug.org

Linda Seeley
Scholarship Director
E-mail: ScholarshipDir@rmoug.org

Heidi Kuhn
Administrative Assistant
Voice Mail: (303) 948-1786
Fax: (303) 933-6603
E-mail: Admin@rmoug.org

Not Pictured
Chris Ostrowski • IS Director • Email: ISDir@rmoug.org
Ann Horton • Oracle Liaison • Email: OracleLiaison1@rmoug.org

February 17-18, 2010  Denver, Colorado
Colorado Convention Center
www.rmoug.org

PRE-TRAINING DAYS UNIVERSITY SESSIONS
TUESDAY, FEBRUARY 16, 1:00-5:00 P.M.
Installing Oracle Database 11
RAC on VMWare
Robert Freeman, LDS Church
Hands-On Lab: Working with SQL Developer and SQL Developer Data
Modeler
Sue Harper and Kris Rice, Oracle Corporation
Understanding Explain Plans
Dan Hotka, Training Specialist
A Guide to Fusion Web Development with JDeveloper 11
Peter Koletzke, Quovera and Duncan Mills, Oracle Corporation
Oracle Database 11
R2 New Features
Daniel Liu, Oracle Corporation
Troubleshooting Oracle APEX
Scott Spendolini, Sumner Technologies
APPSPERF: Optimizing Your E-Business Suite Upgrade to
Minimize Downtime
Jerry Ireland and Mark Farnham, RIGHTSIZING
REGISTER TODAY!
ADVANCE REGISTRATION
OPEN UNTIL JANUARY 12.
RMOUG Member- $240
Non-Member - $300
New Member - $315
UNIVERSITY SESSION
REGISTRATION
$125
(10% discount if you register for Training Days)
HAVE QUESTIONS?
Call 910.452.0006 or Email rmoug@teamycc.com
RMOUG TRAINING DAYS
February 17-18, 2010 | Denver, Colorado
Colorado Convention Center
www.rmouug.org

THE COLLEGIATE PEAKS PHOTO COURTESY OF JOHN KING, RMOUG MEMBER

PRE-TRAINING DAYS UNIVERSITY SESSIONS
TUESDAY, FEBRUARY 16, 1:00-5:00 P.M.

- Installing Oracle Database 11g RAC on VMWare
  Robert Freeman, LDS Church

- Hands-On Lab: Working with SQL Developer and SQL Developer Data Modeler
  Sue Harper and Kris Rice, Oracle Corporation

- Understanding Explain Plans
  Dan Hotka, Training Specialist

- A Guide to Fusion Web Development with JDeveloper 11g
  Peter Koletzke, Quovera and Duncan Mills, Oracle Corporation

- Oracle Database 11gR2 New Features
  Daniel Liu, Oracle Corporation

- Troubleshooting Oracle APEX
  Scott Spendolini, Sumner Technologies

- APPSPERF: Optimizing Your E-Business Suite Upgrade to Minimize Downtime
  Jerry Ireland and Mark Farnham, RIGHTSIZING

REGISTER TODAY!
ADVANCE REGISTRATION OPEN UNTIL JANUARY 12.

RMOUG Member - $240
Non-Member - $300
New Member - $315

UNIVERSITY SESSION REGISTRATION
$125
(10% discount if you register for Training Days)

HAVE QUESTIONS?
Call 910.452.0006 or E-mail rmouug@teamycc.com
| November       | 11/2/09 | Newsletter - Call for Articles Winter Issue          |
|               | 11/3/09 | Training Days 2010 - Early Registration begins       |
|               | 11/8/09 | Training Days 2010 - Speaker confirmation due        |
|               | 11/19/09| Training Days 2010 - Schedule at a Glance posted to website |
|               | 11/20/09| Board Meeting @ 7:00am before first QEW session      |
|               | 11/20/09| Quarterly Educational Workshop                       |
|               | 11/20/09| Scholarship - Announce Winter Scholarship recipients at QEW |
|               | 11/24/09| Newsletter - Deadline for Articles Winter Issue      |
| December      | 12/3/09 | December Meetup - Oracle Enterprise Manager (OEM) Lab @ 6:00pm at Regis University |
|               | 12/15/09| Board Monthly Meeting - Conference Call @ 5:30pm     |
|               | 12/18/09| Newsletter - Mail Date Winter Issue                  |
|               | 12/3/09 | January Meetup - TBD @ 6:00pm at Regis University    |
| January       | 1/12/10 | Training Days 2010 - Speaker Presentation Papers Due |
|               | 1/13/10 | Training Days 2010 - Last day for Early Registration |
|               | 1/13/10 | Training Days 2010 - Standard registration begins    |
|               | 1/14/10 | Board Monthly Meeting - Corporate Offices @ 5:30pm    |
| February      | 2/8/10  | Newsletter - Call for Articles Spring Issue         |
|               | 2/9/10  | Training Days 2010 - Last day for Standard registration |
|               | 2/10/10 | Training Days 2010 - Late Registration begins        |
|               | 2/16/2010-2/18/2010| Training Days 2010 Conference              |
|               | 2/17/10 | Board Monthly Meeting @ Training Days                |
|               | 2/28/10 | Newsletter - Deadline for Articles Spring Issue      |
| March         | 3/11/10 | RMOUG / ISACA Workshop - details coming              |
|               | 3/11/10 | Board Monthly Meeting - Conference Call @ 5:30pm     |
|               | 3/15/10 | Scholarship - Accepting Applications for Spring Scholarship |
|               | 3/22/10 | Newsletter - Mail Date Spring Issue                  |
| April         | 4/5/10  | QEW - Call for Presentations                         |
|               | 4/16/10 | QEW - Deadline for Presentations                     |
|               | 4/22/10 | Board Monthly Meeting - Corporate Offices @ 5:30pm  |
|               | 4/23/10 | QEW - Preliminary Agenda                            |
|               | 4/30/10 | Scholarship - Deadline for Applications for Spring Scholarship |
| May           | 5/3/10  | Newsletter - Call for Articles Summer Issue          |
|               | 5/21/10 | Board Meeting - 7:00am before first QEW session      |

---

**Ever use tkprof?**

See three incredible new tools for your trace data, designed by Cary Millsap.

http://method-r.com/software/mrtools

---

**Yannick Computer Solutions, Inc.**

Yannick LeRolland
Software Support
Computer Consultant
Systems & Network Administration

720.840.1965
7152 Cerney Circle
Castle Rock, CO 80108-3402

Yannick_lerolland@live.com

---

**Method R*®**
Join us for our next Quarterly Education Workshop next May. RMOUG hosts quarterly workshops in May, August and November of each year with the fourth and largest educational event being Training Days in February. Learn about the newest technologies, gain more insight into Oracle techniques and enjoy the camaraderie of meeting with other Oracle professionals.

If you or your organization are interested in partnering with RMOUG to host an upcoming meeting, or to submit an abstract for presentation, please contact

Allison Leech, Programs Director at ProgramsDir@rmoug.org

Watch RMOUG’s Web Page for May Location and Topics

www.rmoug.org
Envisioning A Better Future and Designing Technology To Create It...

✓ Oracle
- E-Business Suite implementation, upgrades, migrations, and support
- Fusion Middleware and Open Systems Development
- Business Intelligence (OBIEE) development
- Hyperion Financial Performance Management
- DBA and Database tactical services

✓ IT Infrastructure
- Managed Services, Remote DBA support, and Hosting (Infrastructure & Applications)
- BI Rationalization (consolidation) services and tools
- Security and Identity Management
- IT Infrastructure Performance, High Availability, and Scalability implementation and support
- Strategic Global Sourcing

✓ Profitability and Cost Management
- Budgeting and Forecasting Solutions
- Operational Intelligence (BI)
- ERP Business Intelligence
- Enterprise Performance Management

✓ Rolta Software Solutions
- iPerspective™ - Rapid development of Enterprise Web Services
- Geospatial Fusion™ - Enterprise Spatial Integration and Visualization
- OneView™ - Business and Operational Intelligence