Steven Feuerenstein
Guarantee Application Success

John Krahulec
Enterprise Social Networking

Tim Mishek
ASM - The Next Generation

Benjamin Wood
Four Things To Know About MySQL

Board Focus - Heidi Kuhn
Member Focus - Dan Hotka

Change Service Requested

Non-Profit
Organization
U.S. Postage Paid
San Dimas, CA
Permit No. 410
Training Days has a long tradition of excellence.
The price is right.
The networking atmosphere is perfect.
Training Days is for Developers, DBAs, and Managers.

Early registration begins in November

Email TrainingDaysDirector@rmoug.org
On the Cover:
Glenn Goodrum has been working with Oracle since version 5 in 1988, as a database designer, DBA, and data architect. Presently he is the data architect at SquareTwo Financial in the Denver Technology Center. As a hobby, he enjoys photographing the fabulous scenery in the great state of Colorado. He took this picture of a barn on Jefferson County Road 73 near Conifer on March 20, 2010, a day of bright blue sky following a fresh snowfall in the foothills.

BE MORE THAN A TECHNOLOGY CONSULTANT.
BE AN INFLUENTIAL TECHNOLOGY CONSULTANT.

> We have the tools to position you for influence across industries. Are you ready to take your seat at the table?

Regis University’s School of Computer & Information Sciences not only equips adult learners with the theory and technical skills needed to compete with the rapid changes of the technology landscape. But it also delivers a rigorous core curriculum that provides the critical thinking and creative problem-solving skills that will empower you to make bold choices, blaze new trails, and lead a company to where it’s never been before.

A Regis Business Technology Management degree combines the technical aspects of the IT field with project planning, budgeting, leadership, and management—a hybrid skill set highly sought after by today’s business technology leaders. And with our seasoned practitioner faculty, access to major commercial software, and strong business and government relationships, we’ll get you on the fast-track to influencing the boardroom more quickly than you imagined. Experience the power of influence.

BE INFLUENTIAL.
As this year comes to a close I want to thank everyone who has helped make 2010 a great success!

Over this past year RMOUG has received continual support from both the local and worldwide IT community for many member events:

RMOUG kicked off the year with the Training Days conference February 16th – 18th at the Colorado Convention Center with over 120 sessions.

- Training Days 2010 sponsors included Oracle, Oracle Technology Network (OTN) and Quest Software.
- Over eighty speakers from RMOUG and around the world came together to present what has become known as one of the top Oracle conferences. This year over 25 Oracle Aces/Ace Directors attended; these individuals have been recognized by the Oracle Technology Network as Subject Matter Experts (SMEs).
- This was the first year for the Oracle Ace Panel at RMOUG Training Days providing attendees the opportunity to have their questions answered by OTN's SMEs. The panel Thursday morning was moderated by Duncan Mills (Oracle Corporation) with Ace Directors Alex Gorbachev (Database Management & Performance), Cary Milsap (Database App Development), Debra Lilley (Applications & Apps Technology), Dan Morgan (Database Management & Performance), Scott Spendolini (Database App Development) and Ace Chen Shapira (Database Management & Performance) as panelists.

2010’s three Quarterly Educational Workshops (QEWs) offered a mix of topics including: database tuning, design, application development, and security technology.

- Thank you to Comcast, Regis University SCIS, and Oracle Corporation for hosting our workshops.
- Thanks to all speakers who shared knowledge at these workshops.
- A special “Thank you” to DBAK for sponsoring our August workshop.

In March, RMOUG and Information Systems Audit and Control Association (ISACA) partnered for their first joint meeting -- hopefully this will be the first of many opportunities for RMOUG to partner with other user groups to provide related training and networking opportunities to RMOUG members in the future.

Database Lab (Meetups) continued in 2010 with five meetups (March, May, July, September, and November). These Database Labs are hands-on learning opportunities that are hosted at Regis University SCIS (School of Computer & Information Sciences) using the latest Oracle and other database technologies, products, features, and utilities. Check out the RMOUG homepage for details on the next Database Lab.

SQL>Update_RMOUG newsletter continued its fine tradition of excellent articles, insightful reviews, and interesting interviews throughout 2010. Share a copy with a co-worker today!

Well, there it is --- RMOUG’s year “2010” in a nutshell.

In closing: I would like to extend a very special “Thank you” to those who have helped to organize all of these events. As Pablo Picasso once said, “Action is the foundational key to all success.” These volunteers’ actions, dedication, and new ideas have contributed greatly to the continued growth and success of RMOUG.

So consider taking action yourself by making a commitment to get more involved with RMOUG in 2011. We need your ideas and/or time to ensure RMOUG’s continued success and growth. So consider getting involved by writing an article for the newsletter, speaking at our QEWs and/or Training Days, volunteering at Training Days, helping Directors with their duties, or running for the Board of Directors in 2011.

I look forward to seeing you at our scheduled events next year.

Peggy King
2009/2010 President
Be Sure To Attend The
Spring Quarterly
Education Workshop
May 20, 2011

RMOUG is seeking abstracts for this meeting. Presentations may include overviews, tips, techniques, testimonials, and lessons-learned. For abstract submission, we are seeking the equivalent of a proposal for the presentation.

If you or your company are interested in sponsoring breakfast for the workshop, please see page 11 for rates.

Sponsor the next QEW and receive an ad in SQL>Update at HALF the normal price.

Please contact
Carolyn Fryc
cfryc@orsportal.com

Check www.rmoug.org for the location, times and featured speakers

Stan Yellott Scholarship Fund
Application
Call for Spring Applications
Submissions Accepted February 15, 2011

The Rocky Mountain Oracle User's Group (RMOUG) is committed to supporting others in their pursuit of technical knowledge.

Since 2001, the RMOUG Scholarship Fund has provided funds to encourage future IT professionals in their efforts to broaden their knowledge.

In 2007, RMOUG voted to rename the scholarship fund to honor the memory of Stan Yellott. A long time member of RMOUG, Stan supported the user community by serving on the Board with a focus on expanding educational opportunities. Stan took this commitment beyond colleagues by including high school and college students, the next generation of IT professionals.

The Stan Yellott Scholarship Fund will honor Stan's good work by assisting deserving students to attain their IT educational goals. Scholarships are awarded to students interested in pursuing studies related to Information Technology.

Eligibility Requirements:

- Registered or the intent to register at an accredited post secondary educational institution in the United States
- Minimum GPA 2.5 from a current official transcript
- Currently enrolled in a computer science/information technology class and/or intends to enroll in a computer science the following term

For details visit the RMOUG website at www.rmoug.org and click on the Scholarship tab.

If you are interested in contributing to the scholarship fund, please contact Heidi Kuhn

HeidiKuhn@rmoug.org

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.
Guarantee Application Success

by Steven Feuerstein

Who’s Guaranteeing What?
The lawyers at Quest Software asked me to clarify something right up front: using our software will not guarantee that you will be successful.

Having said that, I do believe that if you follow the ideas in this paper and my presentation you are likely to improve the chances of delivering a successful application.

Key Criteria For A Successful Application
I’ve been writing software for thirty years. Wow, that sounds like a long time. In fact, it is a long time. And I’ve been involved in my share of successful projects and failed projects. I certainly know which kind I would prefer to be part of.

In the last few years, I’ve been focusing my work in PL/SQL on the challenge of improving code quality through automated testing. That’s really hard to do - hmmm, I guess that’s why we’re not all yet doing it. And as I have worked through the challenges, I have also taken a step back and asked myself some fundamental questions about software.

One of those questions is: what does it take for an application to be considered a success (and us, as the developers of that application, successful)?

After giving it some thought, I have come to believe that there are three fundamental criteria for any successful application:

1. The application must be correct. We build an application for a community of users. That application must meet the requirements of those users.
2. The application must be fast enough. It doesn’t have to run as fast as theoretically possible, but it has to run fast enough to avoid user frustration (and outright anger).
3. The application must be maintainable. If it is very difficult (and resource intensive) for developers to fix and enhance the application over time, the application will come to be seen as a failure.

I hope you agree with these criteria. Certainly, I don’t think I have uncovered any great secrets of software here. This is all common sense, but I do think that it also does cover the key elements of successful software.

If, by the way, you think that I missed anything important, let me know: steven.feuerstein@quest.com.

I will now explore each of these in a bit more detail and, for each, ask the obvious question: how do you achieve correct, fast enough, and maintainable?

A Successful Application Is A Correct Application
It seems rather obvious, doesn’t it? How can an application be a success if it is not correct? If the application doesn’t meet user requirements or is full of bugs, it will be considered a failure.

Hmmm. Think about your applications. How many times have you actually met all user requirements and delivered bug-free code? The likely answer is “never.” Yet our software is still often considered successful. How can this be?

Incomplete and buggy applications can still be considered a success, because we have “educated” our users to reduce their expectations of our products! “Everyone knows” that all software has bugs. And our users have gotten used to only getting a portion of what they ask for.

Still, even with reduced expectations, we need to make sure that we deliver working code, and the only way to be sure that our programs work, the only way to prove that they work...is to test our code.

The bottom line: a correct application is a tested application.

Why Developers Hate To Test Code
There are some very good reasons that developers hate to test code.

First of all, most testing is done “by hand” - that is, manually. To test a program, I must write a test script. I have to make sure my tables have the right code, my package variables have the right values, etc. Then I run my test script and verify the results. Did I get the right answer? Do I even know what the right answer is? And if it’s the wrong answer, is the problem in my program or in all test code I wrote?

Experts in unit testing say that you should expect to have to write at least ten lines of test code for every line of application code you need to test.

So if you have a 500 line program, you should fully expect to write 5,000 lines of test code to fully exercise your application logic. Wow! 5,000 lines of test code - that must be written, debugged, tested (that’s right - test your test code) and maintained alongside the program it is testing.

No wonder we don’t test our code.

And it gets worse from there. Writing software is a fun, creative process. We get to solve logical puzzles, play around with
interesting technologies - and get paid pretty well to do all that. It's a good deal.

But testing isn't much fun. It’s boring. It’s...insulting. “What, you think my program has bugs in it? How dare you!”

So from a psychological standpoint, we do everything we can to avoid testing, and concentrate most of our efforts in implementing those user requirements.

**Manual Testing Is A Dead End**

Almost all the testing we do (and we can’t avoid it completely) is a manual process. I write the test code, I run the test code, I stare at the results and ask myself: “Is that right?” And of course I am fairly desperate and strongly incented to say “Yes!” so I will find a way to rationalize saying “My program works.” even if it doesn't really do everything it should in the way that it should.

And so my programs go into production still filled with a lot of bugs. The users find those bugs (they seem to be really good at doing that, don’t they?), get frustrated and irritated with us, and we spend more and more of our time fixing bugs, putting out fires, and have less and less time to implement requests for new functionality, and much less time to “do things right.”

It's a vicious cycle, and it's one that we will never break out of - as long as we rely on manual testing. The only way to change the dynamic, to reduce the number of bugs in our code and, consequently, give us all some breathing room to improve our process and deliver more successful applications, is to automate the testing process.

**Automated Testing Options For PL/SQL**

The more you can automate testing, the more thoroughly you can test your code, covering more test cases and uncovering more bugs. You can test more often and more quickly, so there will be fewer opportunities for bugs to creep into your code as you work on your programs. And you can produce and run regression tests, which are a key ingredient for maintainable applications.

These are the areas of automation that are needed in the testing process:

- Define test cases: make sure each requirement is covered by a test case and all boundary conditions (“weird values”) are tested.
- Test code construction: the less test code a developer has to write, the better. This is one of the major bottlenecks with testing.
- Test execution: it should be easy to run your tests, either on demand or through scheduled execution via a command line interface.
- Results verification: manual verification is a slow and error-prone affair. A testing tool should automatically figure out which of our tests failed and which succeeded.
- Reports: reports should be available that automatically tell us about the state of our testing. How thoroughly have I covered my requirements? Which of my programs succeed and which do not?

Regarding automated testing options for PL/SQL programs, the situation in 2010 is much better than it’s been at any time in the past.

All of the following frameworks or tools automate some portion of the testing process. You should check out each one and evaluate its strengths and weaknesses against your requirements.

- utPLSQL: the “granddaddy” of the open source testing frameworks, utPLSQL is, essentially, “JUnit for PL/SQL.” It is built on Extreme Programming principles, just like all the other members of the “JUnit” family of tools. Unfortunately, you still have to write the test code yourself.
- PLUnit: similar to utPLSQL with a reduced number of automated tests available.
- PLUTO: similar to utPLSQL, but built using object types.
- dbFit: a testing platform for database programming languages build on top of the Fitnesse framework.
- DBUnit: a testing framework that is designed to make it easier for you to set up and restore the states of your database tables when testing your backend code.
- Quest Code Tester for Oracle: a commercial product from Quest Software, Code Tester stores your program’s expected behaviors in a repository and then generates test code for you. It offers a wide variety of automated tests.
- SQL Developer 2.1 (as of November 1, 2009 still an “Early Adaptor” version): repository-based integrated unit testing that allows you to specify setup, teardown and validation operations, as well as libraries of reusable test elements.

**A Successful Application is “Fast Enough”**

An application will not be regarded as successful if it runs so slowly that the users get frustrated and want to throw their monitors out the window.

The application has to be fast enough to keep users productive and happy.

Having said that, very few PL/SQL developers write code that requires every single line to be optimized to the fullest extent possible. In fact, committing the resources to optimize your code to run as fast as theoretically possible is most likely a big waste of your time and development resources.

Most of the code you write will never be part of a bottleneck, and if it’s not causing a problem for the user, then optimizing the code is unlikely to be of much benefit to the user, either.

I suggest the following general orientation towards making sure your application is fast enough:

- Make sure you take advantage of the most important performance optimization features of PL/SQL.
- Other than that, focus on writing code that is correct and maintainable.
- When you have verified that the code is correct, perform stress and load testing on the code to identify bottlenecks.
- Apply more specialized tuning techniques to get rid of the bottlenecks.

**A Successful Application Is A Maintainable Application**

We certainly learned one important lesson from Y2K: the code we write today is going to be around, and running, for a long time.

And you’d think company management, from CIO to CTO to development managers, would have learned a critical corollary lesson:

We have to pay attention not only to delivering an application on time, but also to building it in such a way that we can maintain it (fix bugs, add new features) efficiently over its long life.

Most developers follow a tried and true “methodology” called “Quick and Dirty.” We want to do things the right way, but our
manager is pressing us daily to hurry up and get this program finished so we can work on the next program, and the next, until we run out of time and “go production.”

So we don’t take the time to set clear standards, we don’t build or take advantage of much in the way of reusable code, we take shortcuts and hard-code things left and right, we copy-and-paste and ignore the nagging of our consciences - and we “get it done.”

The application goes production, the users find and report bugs - and at the same time request new features and changes to existing features.

After a while, unfortunately, we are so busy putting out fires from the last release of the application that we have little or no time to build anything new. The cost of maintenance has overwhelmed our team. No one is happy, everyone is frustrated, but there doesn’t seem to be a way out.

We really should not let this happen! So...how can we write more maintainable applications that avoid this terrible trap? I suggest the following:

- Managers must protect their developers.
- We are lost without regression tests.
- Write code based on standards
- Make standards as active as possible.
- Verify that standards are followed.

I explore each of these topics below.

Managers Must Protect Their Developers

Software development managers (I’ll refer to them as DMs in this section) must get a lot of pressure from their managers to deliver product now, or better yet yesterday. After all, those managers (a CTO perhaps or CIO) are getting calls from the actual users: “Where’s that new screen we need? When will I be able to run that report?”

That doesn’t sound like much fun and I certainly do not want to be a manager.

Unfortunately, I get the impression that the way many DMs deal with this pressure is to pass it down the line: to the developers. And that is the worst thing a DM can do if she wants to deliver quality software to her users.

When was the last time your manager asked you, “When you will be done”? Probably just before you read this document.

But can you remember the last time your manager asked you any of the following:

- How maintainable is that code?
- How thoroughly have you tested that code?
- How can I help you do your job more effectively?

How differently would you feel about your DM if she did ask you such questions?

I suggest that if a manager wants to get the best results from her developers, she should protect them from the pressures of the users. Of course they want that application yesterday. Of course it needs more features than were delivered. But passing on that pressure to developers won’t get you there faster or better. Your developers will simply take even more short-cuts, get even quicker and dirtier. You’ll just end up with software that is (more) full of bugs and less likely to satisfy users.

Everyone would be much better served by having managers protect and defend their developers, give them more breathing space, and make it more likely that they will do things right and produce code with fewer bugs and higher readability.

Regression Tests Are A Necessity For Maintenance

We can talk about standards-based development and best practices all we want, but if you don’t have a regression test in place it will be all but impossible to maintain applications.

I bet you’ve heard of the term “regression test.” Did you ever wonder why it is called that? The point of a regression test is to make sure that our code does not regress or go backwards. In other words, if it worked in version 1.2, it should still work in version 1.3.

Well, obviously.

But if that is so obvious why does the opposite happen so often?

Our users ask for 100 features and we deliver 25. “Sorry,” we tell them. “We didn’t have time to finish the others.” The users are not happy, but what can they do? So they use those 25 features - every single day.

Then in the next version, we have a boatload of new features - and break several of the original 25. Howls of frustration and bewilderment rise from the cubicles of our users:

“How could it happen?” they cry out.

“How could it work yesterday and not work today?”

And if we tell them the truth, they will run screaming out of the room and never work with us again. Because this would be the honest answer:

“We make changes to our code and - hey, it’s really complicated stuff - we don’t really know what the impact of those changes are. And there are so many features it’s hard to test them all. So we just hope and pray that every one still works.”

That sounds totally insane, and yet that is the everyday reality of most software development shops.

How do you ensure that all features from the last version work in the latest version? You test each and every one of them. How do you do that without consuming all the time and resources of your team? With automated testing tools.

Write Code Based On Standards

So you’ve got a solid set of regression tests in place. Your manager is fighting off the users, giving you the space to write some excellent code.

How are you going to do that?

By setting standards across as much of your code base as you can. At a minimum, you need to agree on ways to write the following chunks of your code:

...if a manager wants to get the best results from her developers, she should protect them from the pressures of the users...
Basic naming conventions and language utilization: everyone should write code roughly the same way - and make sure to take advantage of critical features like FORALL, BULK COLLECT, etc.

How to raise, handle, log and communicate errors: ideally, a developer should do nothing more in an exception handler than call a pre-defined, shared procedure for logging and re-raising errors. Check out the Quest Error Manager (freeware) for one such utility.

How, when and where to write SQL statements: it never ceases to astonish me that many PL/SQL shops will have rules about how to name local variables, but not a single rule in place for writing SQL statements. Those statements are the cause of most performance issues, are constantly changing due to business model evolution, and raise many errors. Yet since SQL is so easy to write, we take it completely for granted. Instead, you should define a separate data access layer to manage underlying tables. High-level application code (on the front and back ends) should call procedures and functions in this layer, never hard-code SQL statements.

Generally NEVER REPEAT ANYTHING. Instead, hide hard-coded literal values behind constants or functions, put all your SQL into a data access layer, hide all formulas and business rules inside functions, use SUBTYPEs to avoid hard-coded declarations of constrained types like VARCHAR2(100), and so on.

Make Standards As Active As Possible

If standards consist of nothing more than documents full of recommendations, it is very likely that no more than a few really dedicated souls will actually read and remember all the rules. You should do everything you can to make the standards less passive and more active. Here are some ideas:

- Build a set of templates that incorporate the standards and essentially take the “carrot” approach towards compliance: make it easier and more productive for developers to follow the standards, rather than write it themselves.
- Pre-build as much of the generic application code as possible. That is, put in place a foundation of code on which everyone’s programs will be written. Generic error management, table and transaction APIs, extended string utilities, etc.

Verify That Standards Are Followed

If developers notice after a while that no one is checking to see if they are following standards, they will pay less attention to, and eventually forget about, those standards.

- Peer code review: look at and critique code. This process will never cover all your code, but it is a great way to share knowledge about the application and expertise on the underlying code foundation.
- IDE-based automated code analysis: Toad and SQL Navigator offer CodeXpert, certainly one of the best. PL/SQL Developer integrates code quality feedback into its editor window. Another utility that performs a similar analysis is ClearSQL. None of these tools allow you to specify your own rules.

- Home-grown queries against data dictionary views: ALL_SOURCE, ALL_ARGUMENTS, ALL_DEPENDENCIES, etc. You can perform endless sorts of data mining against your code.
- Oracle’s compile-time warnings: you can now ask Oracle to give you feedback on code quality by enabling warnings before compilation.
- Oracle11g PL/Scope: the newest and best code analysis feature of PL/SQL, PL/Scope gathers information about all identifiers (named elements) in your compiled code. You can use it to check for compliance with naming conventions and obtain detailed information about how subprograms and variables are used and manipulated in your code.

Guarantee Application Success
You Can Do It!

The software industry has certainly helped businesses, governments and individuals achieve new levels of productivity. Many factors contribute, however, to producing software that has too many bugs, that causes too much user frustration, and that is hard to fix, enhance and evolve over time.

It is certainly possible, however, to break out of this cycle, and provide higher quality software that will lead to higher levels of success for everyone, users and developers alike.

Steven Feuerstein is considered to be one of the world’s leading experts on the Oracle PL/SQL language, having written ten books on PL/SQL, including Oracle PL/SQL Programming and Oracle PL/SQL Best Practices (all published by O’Reilly Media). Steven has been developing software since 1980, spent five years with Oracle (1987-1992), and has served as PL/SQL Evangelist for Quest Software since January 2001. He is an Oracle ACE Director and writes regularly for Oracle Magazine, which named him the PL/SQL Developer of the Year in both 2002 and 2006. He is also the first recipient of ODTUG’s Lifetime Achievement Award (2009). In 2010, Steven started the PL/SQL Challenge (www.PLSQULChallenge.com) an online PL/SQL quiz that has over 1300 daily players. Steven’s online PL/SQL cyberhome is located at www.ToadWorld.com/SF; but you can also find out more about Steven at www.StevenFeuerstein.com.
Enterprise Social Networking
It’s Now Ready For The Workplace

by John Krahulec

The term Social Networking makes people think about sites like Facebook and the sometimes-inappropriate information that people post on it. Rarely does anyone think about how this new medium has enabled people to communicate in new ways. Enterprise Social Networking takes advantage of these communication methods to allow employees to share, collaborate, and stay in touch about business-related items. Enterprise Social Networking connects People to People and People to Information. Social Networking connects People to People.

Enterprise Social Networking has been a holy grail for many years. It had many names: Enterprise Social Media, Enterprise 2.0, Gov 2.0, Enterprise Collaboration, and so on. Regardless of the name, the objective has always been to improve communication and collaboration within an organization. This concept has been fueled by the success of Social Networking sites like Facebook and Twitter. Many organizations have tried to embrace these existing sites for their own Enterprise use, but have quickly found that the nature of these open (and very public) sites does not meet the Enterprise requirements of information security, policy enforcement, and application integration, while maintaining meaningful collaboration among other things.

In this article, I will cover the merits of Social Networking, the merits of applying it in the Enterprise, and how to do it all from an Oracle database.

The Merits of Social Networking

Social apps aim to provide value in three ways. First, they can bridge geographical and organizational information divisions by moving conversations out of email and hallways and into shared online spaces such as walls, forums, blogs and wikis. This way, information becomes searchable, serendipitous connections can be made, and ideas pollinate in ways they couldn’t before. Communities of interest spring up around subject matter rather than organizational hierarchies.

The second way social apps provide business value is by letting people add context to information stores, which helps others identify what’s useful to them and can make search results more relevant. Social bookmarks, for example, let people share links to a content source – a Web page, blog, or white paper – that they found useful. Tags, social bookmarks, and other social networking tools help bring order to the avalanche of information that employees have to sift through.

Third, Enterprise Social Networking helps people find and connect to co-workers through user profiles, subject matter forums, and social graphs that provide a visual map of an employee’s connections with co-workers. No longer are employees just a cog in the machine. They are people again.

Now, let’s do a quick review of the current social media elements that make up Social Networking:

- IM/Chat – AIM, Google Chat, Yahoo Messenger
- Blogs - WordPress, Drupal
- Microblogs - Twitter
- Collaborative Encyclopedia - Wikipedia
- Social Networks – Facebook, LinkedIn

Each has a specific purpose and offers different types of communication. You could certainly use all of them in the office, but that doesn’t make it Enterprise. Each one has its own security authentication (no single sign-on) managed by some other unknown IT organization. True, there is some level of integration (e.g. mash-up) capability such as posting Tweets to your Facebook page or displaying your Facebook page on your blog site, but that isn’t the only level of integration we want to achieve within the Enterprise. Also, these choices don’t necessarily allow you to restrict information to specific groups or individuals. That would be a direct contradiction to the intent of current Social Networking: to make and share information with the world!

Now let’s step back a moment. We like all the collaborative features afforded by social media, but it just won’t work in an environment where marketing forecast reports need to be shared with the Marketing & Sales teams (and hidden from the Finance
Bringing the Merits of Social Networking to the Enterprise

Why do we need an Enterprise social network?

Some of the merits of social networking are obvious, in an Enterprise context: Use Status Updates to let others know that you’ll be in a meeting down the hall. Search the online employee directory to find people with common skill sets in other locations of your company. Use embedded Chat windows to discuss marketing data in real-time, and be able to search and retrieve that information later. Dynamically create social groups based on project teams, departments, and/or common interests, and have it include features such as group calendars, file sharing, and news boards. Use ‘Search’ features to access information according to access control lists. The ways you can apply social networking to the Enterprise seem limitless.

We are starting to see some new Enterprise Social Networking products come to market (e.g. Salesforce Chatter) or old products being retooled (or just re-marketed) to have a social networking flair (e.g. SharePoint). As promising as these seem, each has its own inherent problems. Salesforce Chatter works great for those employees in the Sales organization that already have access to Salesforce, but it requires everyone else in the company to now login to yet another site to participate in the “chatter”. But, if there is no compelling reason for them to log in, then only a small portion of your company is participating. This hardly makes it an Enterprise Social Network.

As far as SharePoint is concerned, it started as a document management solution, then was pushed beyond its original intent when developers tried using it to create applications, and then Microsoft created a mish-mash by bolting on some wiki and blog features so they could jump on the Enterprise Social Networking bandwagon. SharePoint is trying to be everything without being able to do anything well. I don’t think the mish-mash can grow.

Road Blocks to Enterprise Adoption

So, it’s clear that while Enterprise Social Networking has merits, there is a lack of understanding of the Enterprise requirements in the current line of product offerings. In addition, you will find some other roadblocks to Enterprise adoption of social networking.

**Perceived Tomfoolery:** First and foremost, there’s the bad rap these apps get from consumer-focused cousins such as Facebook and Twitter, which are perceived as outlets for sophomoric self-aggrandizement. Not much more to say here.

**ROI:** What’s the ROI? Seems like a standard question for every new IT initiative. However, though email has been around for quite a while, I would have loved to hear the ROI justification for it. It’s probably hard to explain, but we know we need it. Just as with e-mail and Internet access 15 years ago, it is difficult to quantify social media’s utility. CIOs know that social networking will find its way into their companies if IT doesn’t provide it. Beyond Facebook and Twitter are scores of wiki, blog, and collaboration sites that are free, or cheap enough that a product manager can buy a half-dozen accounts with a credit card and upload next quarter’s product road map in five minutes. Therefore, even if the ROI is never clear cut, IT managers may be compelled to invest in these apps to keep employees from adopting public systems that are outside company control.

There are some other metrics that can serve to answer part of the ROI question. One metric is user adoption. A large company can have about 10,000 wiki documents and 3,000 blogs across 150 topical communities on its Enterprise Social Network. Of its 10,000 global employees, the platform may have about 1,000 active users and another 1,500 lurkers who are watching conversations or consuming content but haven’t posted anything themselves. Adoption doesn’t necessarily tie in to ROI; however, you should be able to point to a decrease in the volume of emails and attachments across the company’s network, and the resulting reduction in storage costs. Instead of a 20-MByte PowerPoint being sent back and forth among department or the staff in the mail room). Also, you want these features integrated and within context to your corporate applications. Oh, and you want your information to be searchable, but yet again, only to those with the appropriate permission. And just to make it all easier to manage, let’s keep it all on one unified platform... one that presumes, that as a reader of this publication, you are a user of the Oracle database. (That’s not necessarily an overarching requirement of Enterprise Social Networking, but as you’ll see, it makes our work much easier.) So what options are available to the Enterprise? The following table gives you a glimpse of the differences between consumer social networking sites and the type of social networking the Enterprise is looking for:

<table>
<thead>
<tr>
<th>Type of Social Networking Site</th>
<th>Need</th>
</tr>
</thead>
<tbody>
<tr>
<td>Consumer Social Networking</td>
<td></td>
</tr>
<tr>
<td>Enterprise Social Networking</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Features</th>
<th>Controls</th>
</tr>
</thead>
<tbody>
<tr>
<td>User authentication at login, but security breaches commonly found</td>
<td>Minor access control for groups</td>
</tr>
<tr>
<td>No integration with other apps</td>
<td>No messaging control</td>
</tr>
<tr>
<td>Granular access control based on defined groups</td>
<td>Everyone can view any message you post</td>
</tr>
<tr>
<td>Multi-layered security: database level, application level, and use of access control lists</td>
<td>Integrate with existing enterprise apps for real-time data updates across your suite of applications</td>
</tr>
<tr>
<td>Enforce security policies, regulatory compliance</td>
<td>Messaging control can be enforced based on groups</td>
</tr>
</tbody>
</table>

**Examples:**

- **Facebook** allows these choices:
  - Everyone
  - Friends Only
  - Friends of Friends

- **Example:**
  - Defining your access control based on your known friends, or someone you have nothing in common with.

---

**SQL>UPDATE • Winter 2010**
10 co-workers, it can sit in a common workspace where people can post comments about it.

Another way to measure the value of social networking tools is to relate them to process improvements rather than time savings or loose efficiency gains. Some process improvement areas are consolidation/integration, policy/content control, policy questions around content control, compliance, and moderating employee behavior.

**Why Not Just Bring a Facebook Platform Into the Enterprise?**

This has been done. Jive Software, Lotus Connections by IBM, Eureka Streams (www.eurekastreams.org) by Lockheed Martin. However, the problem is that it is still another website to log into. It is not integrated with the other corporate applications used on a daily basis. It loses Enterprise context if the social network is not integrated with the applications.

**The Easy Answer**

The easy answer is to use a platform that allows for custom application development, and has built-in features for Enterprise Social Networking, document management and collaboration, a calendar that supports individual and group events, etc., to create a truly Rich Web Application.

**Applying Social Networking to the Enterprise with the Oracle Database**

So, you're convinced of the merits of social networking, and you want to apply it to your Enterprise. You have seen consumer social networking sites fail to meet Enterprise demands. However, you have your Enterprise Social Networking strategy planned: you know what types of dynamic groups you'd want within your Enterprise, what security policies you'll need to adhere to, the regulations, if any, your Enterprise must follow, and so on. Now, you want to see what the best, most efficient way is to roll out your strategy.

In other words, the question is: How can you reap the benefits of social networking, meet stringent Enterprise requirements, and leverage your existing infrastructure? Let's consider these basic factors to evolving your strategy:

- How will you reconcile existing workplace policies with the new virtual environment?
- How will you maintain your Enterprise Social Networking tool of choice in a separate database?
- What resources will you allocate to maintain data, independent databases, and software?
- How will you monitor the stringent security policies your Enterprise demands?
- How will you maintain control of information while encouraging communication?

Rather than look around for expensive, stand-alone Commercial Off The Shelf (COTS) products, you can leverage your existing Oracle database to bring together all the great elements of Social Networking to improve how your Enterprise works. Just like Enterprise Social Networking is a new way to streamline and leverage the same vast pools of corporate information, using your Oracle database to develop your Enterprise Social Networking tool is a new way to leverage the same vast potential your Oracle database offers.

Let's take it one step further. What could be the easiest way to leverage your existing Oracle infrastructure, resources, training, man-power, and maintenance, without compromising on cutting-edge web technologies like HTML 5, JSON, AJAX, and FLASH that makes those consumer social networking sites so zesty?

By using PL/SQL. With PL/SQL, you can use an unlimited number of Application Programming Interfaces (APIs) that will work with your application and maximize PL/SQL's top performance for database-intensive applications. For example, integrating the instant messaging feature allows you to leverage all your database features as opposed to using a stand-alone product that will require building APIs to retrieve 'chat' logs from your Oracle database.

Let's look at how you can build an instant messaging feature integrated with your Oracle database as part of your social Enterprise networking strategy, eliminate the traditional 'middle-tier', secure your data with multi-level security, pare down development efforts, and provide the same rich user experience of consumer social networking sites:

Set up tables within your database to store chats, chat archives, buddies, chat sessions, and chat user information.

Use JSON to manage data to your chat's client components: the Buddy List and the Chat window input fields used for sending messages. For example, clicking on the Submit button to send a message activates AJAX which transmits the message and the name of the recipient directly to an Oracle stored procedure.

Use PL/SQL to transmit/receive the message and recipient variables, and insert the data into your database tables.

To fetch data, use a Polling System to make AJAX requests to the database. The Polling System invokes a procedure that checks the status of a user's buddies, for example, or looks for messages to be received. The JSON code block is sent to the client.
The following sample code shows how to write in-bound messages to the database:

```sql
PROCEDURE p_chat_write
  (i_rx  IN VARCHAR2,
   i_message IN turbo_chat.message%TYPE)
IS BEGIN
  INSERT INTO turbo_chat
    (message_id, tx_user_id, rx_user_id, tx_timestamp, message)
  VALUES
    (seq_chat_message.NEXTVAL, te_security.g_security_rec.user_id, te_crypto.f_decrypt(i_rx), CURRENT_TIMESTAMP, i_message);
  COMMIT;
EXCEPTION WHEN others THEN
  te_logging.p_error(SQLCODE, SQLERRM, te_logging.f_get_procedure, NULL);
END p_chat_write;
```

The following sample code shows how you can update the JSON object with messages to send to the client:

```sql
PROCEDURE p_chat_read
  (i_json  IN INTEGER)
IS CURSOR messages IS
  SELECT message_id, te_utils.f_get_full_name(tx_user_id) tx_name, tx_user_id,
       TO_CHAR(tx_timestamp, 'MM/DD/YYYY HH24:MI:SS') tx_time,
       message
  FROM turbo_chat
  WHERE rx_user_id = te_security.g_security_rec.user_id
  ORDER BY message_id;
BEGIN
  MERGE INTO turbo_chat_users a
  USING (SELECT te_security.g_security_rec.user_id user_id
            FROM dual) t
  ON (a.user_id = t.user_id)
  WHEN MATCHED THEN
    UPDATE
    SET last_ping = SYSDATE
  WHEN NOT MATCHED THEN
    INSERT
    (user_id, last_ping)
    VALUES
    (te_security.g_security_rec.user_id, SYSDATE);
  te_json.p_object_open(i_json, 'chat');
  te_json.p_array_open(i_json, 'messages');
END p_chat_read;
```

Here is a sample chat window and buddy list that effectively utilizes the database:

You can similarly add additional Enterprise Social Networking capabilities such as an integrated calendar, a personal wall, and a file sharing and management system, to provide seamless information flow. Let's take a quick look at how you can build a wall.

Set up tables within your database to store your wall posts, post images, make entries, and post comments for a personalized wall. The following is a sample table for blog entries:
Each time an entry is posted on the wall, the client-side sends it to your database using Javascript to save the data. You store the data in a table using PL/SQL. The database then creates and sends back a JSON object, whose contents are rendered by the client for display on the wall. The following sample code shows how you can write and read data:

### Write/store new post:

```sql
FUNCTION f_post_entry_merge
    (i_rowtype IN te_types.rt_turbo_blog_entries)
RETURN turbo_blog_entries.entry_id%TYPE IS
l_rowtype te_types.rt_turbo_blog_entries := i_rowtype;
BEGIN
    -- Create entry_id new posts
    IF l_rowtype.entry_id IS NULL THEN
        -- Get new blog id
        SELECT seq_turbo_blog_entries.NEXTVAL INTO l_rowtype.entry_id FROM dual;
    END IF;
    MERGE INTO turbo_blog_entries a
    USING (SELECT l_rowtype.blog_id blog_id,
            l_rowtype.entry_id entry_id
    FROM dual) t
    ON (a.blog_id = t.blog_id
        AND a.entry_id = t.entry_id)
    WHEN MATCHED THEN
        UPDATE
        SET
        blog_text = l_rowtype.blog_text,
        update_user_id = te_security.g_security_rec.user_id,
        update_dt = SYSDATE
    WHEN NOT MATCHED THEN
        INSERT
        (blog_id, entry_id, blog_text, load_user_id, load_dt)
        VALUES
        (l_rowtype.blog_id, l_rowtype.entry_id, l_rowtype.blog_text,
        te_security.g_security_rec.user_id, SYSDATE);
RETURN l_rowtype.entry_id;
EXCEPTION
WHEN OTHERS THEN
    te_logging.p_error(SQLCODE, SQLERRM, te_logging.f_get_
        procedure, NULL);
RETURN NULL;
END f_post_entry_merge;
```

### Read post/wall data and return as JSON object to be rendered by the client:

```sql
PROCEDURE p_blog_json
    (i_blog_id IN VARCHAR2)
IS
CURSOR blogs IS
    SELECT *
    FROM turbo_blog_entries
    WHERE blog_id = te_crypto.f_decrypt(i_blog_id)
ORDER BY load_dt desc;
```

Here is a sample personal Wall:

**Bringing It All Together**

Social networking is coming to the Enterprise as surely as email did in the '80s and '90s. However, the challenges and roadblocks to Enterprise adoption only seem to have fueled a mish-mash
of products that all claim to be the ‘holy grail’ that will make social networking work for the Enterprise. The result is an avalanche of half-baked products that have been thrown together hoping to fill the very present need for a robust Enterprise Social Networking solution, without addressing the requirements of the Enterprise: information security, policy enforcement, and application integration, while connecting people to people and people to information. Instead of installing yet another overly complex, burdensome set of applications on already overloaded employees, try an Enterprise Social Networking solution that is already connected to your Enterprise data. Use your PL/SQL resources and your already stable Oracle-based infrastructure to get started now.

John Krahulec is Executive Vice President and COO for TURBO Enterprise, LLC and also for ConceptSolutions, LLC, a Deloitte Technology Fast 500 IT and management consulting firm. John has led development of TURBO Enterprise, a Rich Internet framework for PL/SQL developers, that delivers Enterprise-class applications from the Oracle Database. He has spent the past several years focused on fusing Web 2.0 technologies (HTML5, AJAX, FLASH, JSON, CSS, XML, etc.) and Enterprise-class social networking features with Enterprise back-end data to produce Rich Web Applications (RWA) in corporate, government, and military environments. With over nineteen years of Enterprise IT experience, John brings a unique perspective to RWA development, having developed databases for both desktop and Web applications.
ASM - The Next Generation

by Tim Mishek

Introduction

It is an exciting time to be an Oracle database professional with the recent release of 11gR2. Before Oracle 10g, there was no other alternative but to use raw devices in supporting Real Application Cluster implementations. In the early days of 10g, Automatic Storage Management (ASM) was introduced to get around the issues related to shared storage in a clustered environment. Using earlier versions of ASM proved to be a challenge, especially when trying to move data between ASM and OS file systems. With 11g, copying files can be accomplished easily with the new cp (copy) command. Earlier versions did not allow the OCR and voting disk used in Oracle RAC to be stored on an ASM volume. Included in release 2, ASM now supports storing OCR and Voting disks directly on ASM disk groups. In fact, Oracle no longer supports storing these files on raw devices unless the database is upgraded from a previous version. Moving forward, ASM has matured into a really powerful storage solution. With 11gR2, ASM is not limited to just database files. ASM Dynamic Volume Manager (ADVM) and ASM Clustered File System (ACFS) are new features that allow the storage of regular operating system files. The file structure is mounted to look just like a file system mount point. With ACFS, an alternative is now available for sharing Oracle homes on an ASM volume where OCFS used to be the only other option.

Installing The Grid Control Infrastructure

With previous versions of Oracle, the ASM configuration and installation was part of the database install process. ASM configuration was done using the database configuration assistant as part of the database setup. This has all changed. The ASM installation is now part of grid control infrastructure in addition to being installed in a separate home. Even though previous versions of Oracle recommend a separate home for ASM, it is no longer possible to install ASM directly in the database home location. It is important to mention that the Oracle base location needs to be different for the grid and database home if doing a RAC installation.

As part of the grid control installation, there are several options for installing ASM, which include RAC and standalone database. With either installation option, the cluster ready services are installed. This is somewhat different in the previous versions only requiring the CSS service to be installed for ASM on standalone databases. In previous versions, it is easy to forget that the cluster services were even running. ASM would be started just like any database instance. This is a big change in the way the cluster services function in a standalone environment in the new release. The cluster services are now used to startup and shutdown ASM and database instances. On a standalone system, Oracle uses Oracle Restart to start and stop cluster services. The installation, although somewhat different in appearance, is pretty straightforward and looks similar to previous ASM installations.

Once the grid control application is installed and the ASM instance is created, the database installation has to be installed in a separate home. As part of the installation, the database configuration assistant asks what type of storage is required for the newly created database files. This should look similar to previous ASM installations.

In Figure 1, the initial installation screen shows the options available when doing the grid infrastructure installation for the first time. In this example, the installation is chosen for a standalone server.

Figure 1. Initial Installation Screen

Figure 2. Create ASM Disk Group
Creating the disk group, redundancy, and assigning candidate disks is shown in Figure 2. This screen should look similar to previous versions of Oracle ASM installations.

ASM Configuration Assistant (asmca)

Even though the ASM installation process takes care of the disk layout and disk groups, there is a new tool used for ASM administrative functions. In the grid control home, the ASM configuration assistant (asmca) is now used to replace the database configuration assistant. There are several additional functions introduced with the new release including three main tabs shown in the new configuration assistant. Disk groups are created at the top level and everything else is built under it. The volume manager, located under the Volumes tab, is used to create ASM volumes that enable ASM disk groups to support non-database file storage. Once the volume is created, file systems are associated with volumes and are created and registered under the ASM Cluster File Systems tab. This is a really big enhancement in that an ASM volume can be mounted as a regular file system and even formatted with a native OS file system. Even though this can be done, it is recommended by Oracle to format ASM volumes using the ASM cluster file system (ACFS).

This initial screen as seen in Figure 4 shows the disk group created during the installation portion as shown in Figure 2. Disk groups can be created, deleted, mounted and dismounted using this screen.

Under the Volumes tab, volume ACFS1 is shown to be a part of the DATA disk group and mounted as /acfs. An ASM volume has to be created before a cluster file system can be mounted.

In this example, the ASM Cluster File Systems tab shows the specifics of the mounted file system. The cluster file system is created on this screen and needs to be associated to a volume.

Going back to a terminal window, the ASM cluster file system is showing up as a mounted file system using the df command in Linux. If this node were part of a cluster, the /acfs file system would look the same across all nodes and would allow file sharing.

Asmcmd

Even though ASMCMD (ASM Command Line Interface) has been around since 10g release 2, the functionality has increased substantially. In earlier releases, basic functionality included commands that a system administrator would do normally at the command line to emulate an operating system environment. This would avoid having to start a SQL*Plus session and run a database query...
within the ASM instance. The commands in earlier releases were limited to listing database files within the disk group and included basic commands for changing directories, copying files within a disk group, and creating aliases. There were a few more commands for checking disk space and connected databases but not much more. In 11gR2, the ASMCMD tool now has a full set of commands which closely mirror UNIX style commands. This functionality gives SAN and system administrators the ability to administrate ASM instances in a familiar command set. In addition to the newly added commands, there is now a rich command set available for administrating ASM volumes.

Integration of Cluster Ready Services

The integration of cluster ready services with ASM is now used for both standalone and clustered databases. In some ways it makes the administration easier, but can be difficult if one is not familiar with Oracle clustering. With previous Oracle versions, the cluster synchronization service (CSS) had to be installed for ASM to work but was transparent to the DBA. In database terms, a standalone ASM instance was managed like any other database instance. Now everything including the database is managed as a cluster. On a standalone installation, this is known as Oracle Restart.

By using the “crs_stat” command, Figure 11. shows the output on a standalone database server. In some ways, this is quite a paradigm shift from what a DBA is used to seeing on a single node database instance.

Additions and Enhancements

The copy (cp) command was introduced in Oracle 11gR1 as an alternative to move data between ASM and OS file systems within the ASMCMD command line interface. Introducing this capability has changed the dynamic of managing Oracle database files. It is no longer necessary to use RMAN to get database files on and off an ASM disk group. The introduction of the SYSASM role in 11gR1 changed the administration capabilities for ASM management. Prior releases, including 11gR1, the SYSDBA privilege would provide a database user full access to an ASM instance. The new release still allows a user the ability to log on to an ASM instance with the SYSDBA role, but full administrative access is not granted. Oracle 11gR2 only allows users that are granted SYSASM to do all administrative tasks. In addition, 11gR2 offers a new set of security options not available in previous versions.

Snapshots

The ability to snapshot a volume is now provided in the new release. If a volume is created as an ASM Cluster File System, the volume manager has the ability to create a snapshot to a point in time by using OEM or the acfsutil command. Having this ability gives administrators an alternative to running snapshots on SAN’s or other storage devices.

Rebalancing Disks

The biggest advantage in using ASM is the ability to perform disk administration functions without taking the database offline for maintenance. In effect, disks can be added or removed without any impact to the database. The rebalance rate can be adjusted to minimize performance problems and maximize data movement rates. The rebalance function puts the data over equal portions of the disks in a given disk group. By eliminating hot spots on disk drives, better performance can be achieved. In addition, hot and cold regions can be defined on disks where more throughput is needed.
**Conclusion**

Oracle has really done it right. Not only is ASM an absolute necessity for database clustering technologies, but is now a better option for general database storage. With the introduction of new administration tools and enhancement of others in addition to the new cluster file system, ASM has become a full featured storage solution. With disk management seamless to the database user, full integration of the clustering technology takes database administration and availability to a new level.

Tim Mishek is currently a database administrator and developer for a government contractor in Colorado Springs. He has over fourteen years of DBA experience of which twelve years has been devoted to Oracle technologies. Tim has presented at numerous conferences including RMOUG Training Days and IOUG Collaborate. He is an avid outdoorsman and runner and enjoys the Colorado Mountains.

---

**Be Sure To Attend The**

**Spring Quarterly Education Workshop**

**May 20, 2011**

RMOUG is seeking abstracts for this meeting. Presentations can include overviews, tips, techniques, and testimonials, and lessons-learned. For abstract submission, we are seeking the equivalent of a proposal for the presentation.

**Please contact**

Carolyn Fryc

cfryc@orsportal.com

---

**REGISTRATION OPENS November 4, 2010**

Take advantage of the Early Registration Discount

Register on or before January 12, 2011

www.RMOUG.org | 910.452.0006

---

**RMOUG TRAINING DAYS**

February 15-17, 2011 | Denver, Colorado

Colorado Convention Center

THE COLLEGIATE PEAKS PHOTO COURTESY OF JOHN KING, RMOUG MEMBER
Four Things To Know About MySQL

by Benjamin Wood

As an Oracle sales consultant representing MySQL, I recently had the opportunity to speak at the St. Louis Oracle Users Group. Many of the Oracle database professionals that I spoke with before my session asked what MySQL was doing at an Oracle event. That’s a great question. Here are four things you should know about MySQL.

1. MySQL is Now an Oracle Product
MySQL, founded in 1995, was an independent company until it was acquired by Sun Microsystems in 2008. In January 2010, Oracle acquired Sun Microsystems, thus becoming the new owner of MySQL. Beyond the familiar Oracle Database 11g, Exadata, Berkeley DB, TimesTen, and MySQL are all offerings from Oracle. Oracle maintains a portfolio of database products to cover a wide range of customer requirements.

MySQL now benefits from Oracle Corporation’s resources and commitment to the product, as seen in the upcoming MySQL 5.5 release. Version 5.5 pushes MySQL scalability even further.

2. MySQL Powers the High Volume Web
Take a look at the twenty of the most highly trafficked websites in the United States and you will find that the majority of them heavily or primarily utilize MySQL to power their online presence. MySQL powers all of the relational database transactions for the customer experience at Facebook. The same holds true at many other sites, from Wikipedia to YouTube. MySQL shines in its ability to easily horizontally scale to deal with the crushing workloads that come with applications used by hundreds of millions of people.

3. MySQL Powers Critical Infrastructure
Did you know that MySQL is responsible for call routing with most cell phone networks? How about that MySQL is powering flight operations on aircraft carriers? Not only is MySQL a highly reliable database in stand-alone configurations, there are a wide variety of high availability and disaster recovery options for mission critical applications that require up to five 9’s of availability.

4. Oracle 11g and MySQL Work Together
Customers from Ebay to TicketMaster use Oracle Database 11g and MySQL together to meet challenging deployment requirements. TicketMaster, for example, deals with the ‘look to book’ business case, where customers are looking at multiple options before making their purchase. When popular attractions go on sale, this generates enormous amounts of queries which are handled by farms of MySQL servers. Master data and final purchases are recorded in Oracle 11g.

There are many ways to get started with MySQL, from new application development to integrating MySQL with your existing database apps. To start using the product, download the MySQL Enterprise Edition from edelivery.oracle.com which includes the MySQL server, MySQL Enterprise Monitor, MySQL Workbench, and other great tools. Leverage your Oracle knowledge and get started with the world’s most popular open source database – now an Oracle product!

Benjamin Wood is a Sales Consultant with Oracle Corporation. His fifteen years of database experience includes a decade as a hands-on Oracle production DBA. The last three years have been spent representing MySQL as a Sales Consultant, first with Sun Microsystems and now with Oracle Corporation.
RMOUG Database Labs

If you haven’t heard about it, RMOUG and Regis University have teamed up to offer hands-on lab sessions on various topics. This is a great way to learn new skills, get some hands-on experience in a lab environment, and network with other users and members! Past topics have included OEM, Oracle on Linux, RMAN, and Oracle database security. The next session is planned for Thursday, January 13, and will cover the tool DBClassify from Zettapoint. There might be an additional third party tool added, so this next session could cover two different tools in one lab session! Future topics in 2011 will include MySQL, OEM11g and more. To stay up-to-date and informed on upcoming sessions, be sure to join the group here:

http://www.meetup.com/RMOUGLabs/

Also, be sure to “friend” us on Facebook to get updates on the labs, as well as other RMOUG events: http://www.facebook.com/RMOUG

SPONSOR A QUARTERLY EDUCATION WORKSHOP AND RECEIVE A HALF PRICE AD IN SQL>UPDATE

<table>
<thead>
<tr>
<th>Breakfast</th>
<th>Discounted Ad Rate</th>
<th>Total Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/4 Page</td>
<td>$350.00</td>
<td>$525.00</td>
</tr>
<tr>
<td>1/2 Page</td>
<td>$350.00</td>
<td>$662.50</td>
</tr>
<tr>
<td>Full Page</td>
<td>$350.00</td>
<td>$850.00</td>
</tr>
<tr>
<td>Inside Cover</td>
<td>$350.00</td>
<td>$975.00</td>
</tr>
<tr>
<td>Back Cover</td>
<td>$350.00</td>
<td>$1,100.00</td>
</tr>
</tbody>
</table>

Contact Carolyn Fryc - Programs Director - 720-221-4432 - cfryc@orsportal.com
My name is Dan Hotka, I've been a regular at RMOUG as long as I can remember.

I live in Iowa but I simply love Colorado. This picture is of me enjoying a trail ride in the Estes Park area. I enjoy the mountains and hiking. My hobbies include the old air-cooled VW beetle restorations. If you have been in on any of my presentations, you have seen most or all of my VW fleet (currently at 2). I purchased a motor home so that I can visit Colorado as often as I can. I've been known to play golf and I enjoy hiking and water sports.

I have been self-employed now for eight years. This adventure started out rather slow but I have been as busy as I want to be since the end of 2004. I enjoy working with the Oracle database. I have been involved with Oracle since the version 4.0 days (remember the six 5 1/4" diskettes! ...I still have the three-book doc set too). I have been a developer; I come from the Burroughs COBOL world. I got a lucky break where a hardware vendor (Sperry Univac, yes, I know Mapper!) hired me and trained me in both Unix and Oracle. This career move has fueled my success since 1986! Many of you might remember when I worked for Platinum Technology (probably my best job to date) and Quest Software (a very close 2nd!). I enjoyed my vendor positions.

My family makes its home in Des Moines. I had the opportunity to marry my best friend 32 years ago. My three children are all grown and live in Des Moines as well. Life moves on and they grow quickly! My travels have taken me away often when they were growing up, but I made it home to as many events as I could. Sometimes they got to travel with me. The Platinum Technology job took me to Europe every month (I really liked this job...). Summer of 1998, while on the job, the whole family came with me! We visited five countries. I kept my appointments; they got to tour Venice Italy, Paris & Versailles in France, Belgium, Austria, and Germany. The actual travel part of my work is what I consider the “work” associated with my business.

My home is decorated with many novelties from my travels. The giraffes are from S. Africa. On my last trip there, my wife discouraged me from bringing home more (there are seven ranging from three feet to taller than me), I responded with how we have to keep the gene pool mixed. I found wooden zebra’s...of all sizes...ranging from one foot to several feet tall; I brought home many of these instead! I have the Matrushka (stacking dolls from Moscow). I bought so many that I wasn’t sure if there was a limit as to what I could take with! I threw out clothes to make room for them all in my suitcase.

For the most part I enjoy the world-wide travel (it would be easier to list the places I have not been!!!). I have traveled to some cool places. I do trip summaries, which I email to friends and post to my website. Most find them educational. The travel has its downside but I can’t imagine myself doing anything else.

I enjoy the old stuff of Italy the best... Rome, Pompeii, etc. I got to know S. Africa well. I have been all over the Far East. Sydney, Australia is a place I didn’t want to return from! I know most of Europe quite well. My ancestry is from the Czech Republic and I’ve been to Prague twice now, finding a small town with my name ‘Lhotka’. Two of my children have been to this little town. I have made many friends along this path as well.

I enjoy what I do. When picking something to focus on in my self-employment, I picked training because I like public speaking and I always enjoyed training. I used to do the new hire training at Platinum Technology. Over the years I have been the ‘train the trainer’ where I attended training and then provided similar training to the staff in the office. I also wrote Unix courseware for my Honeywell position to train the staff that recently purchased Unix hardware. I feel it is training that allows me to contribute the best to the IT community.

I started my IT career at the bottom, a computer operator for a small eastern Iowa utility. I printed bills and such from a Burroughs 1700. The processor was about the size of a four drawer file cabinet, 96 column cards, and two 100KB platter disk drives. The processor had 32K (not megs, but 1024's) of memory...programs written in RPGII. I moved up to COBOL. I really only consider my first two jobs ‘job hops’. I spent years at an insurance company where I ground out COBOL and was a programmer manager. I had good friends who worked for IBM fixing the keypunch
machines (we used to fill people's desk drawers with these punch chips!!! ...practical jokes...they made such a mess), and I was always interested in the folks who came to fix the hardware or sell something. I liked how they came and went and saw many sites and different people.

This led me to my first vendor job: Sperry Rand (Univac)...yes...the dinosaur of computers! Everyone learned their famous Mapper (almost a relational database). This job trained me in both Unix and Oracle. I remember the person who was to learn Oracle...(Oracle v4.1). He didn't want to go to class and said, “Why don't you send Hotka...he always wants to learn something new...”...and away I went. Training was typically in Chicago, a five hour drive, and I would stay with family friends in the area. I made a deal with my manager that I'd go to classes he wanted me to go to, if I could go to additional classes I wanted to go to...NO travel expenses! I got to twelve Unix internals, programming, and all the Oracle classes in 2 ½ years...then...they laid me off when my son was born.

From there, I worked for Honeywell. My joke about that job was I wasn't paid very well but I didn't have to work very hard...I considered it a good trade at the time. Then to Oracle Corp. This is where I met the famous (RMOUG board member) Tim Gorman on my first consulting gig for them...bringing me to Denver every week through the summer of 1991. Oracle was an interesting company to work for at that time...many were there for two years or less or five years or more...at the time...this was a tough bridge to cross.

I then went on to Platinum Technology. I started writing books about this time period too (remember the Oracle Unleashed series?)...1996. This job lasted about three years then I left for Quest Software when Computer Associates bought Platinum and they wanted me on this 'fire fighting' brigade...where I'd fly out Sunday night to patch up some failed sales attempt...appeared to me that I'd be the classic scapegoat...and this was after being a successful director for Platinum Technology, assisting sales and support across Europe. No thank you. I enjoyed being the ‘trade show’ speaker for Quest Software, where I worked closely with the marketing team. When the 911 attack happened, software sales plummeted and Quest Software jockeyed me around different groups for a while and I felt it was a good time to break out on my own. I've had short consulting jobs between some of these stops. I still have good friends from many of these jobs. I learned something from each job as well.

I want to thank the RMOUG staff for their continued support. I have been a regular speaker at RMOUG and now I'm also a regular University Session trainer at Training Days as well. I try to speak at your events as often as I can.

Dan Hotka
Dhotka@Earthlink.net
Eons ago, I began a career in recreation by leading outdoor activities like rafting, mountain biking, and ski trips from a little shop called Outdoors Unlimited in Utah. A move away from the mountains to the east coast meant a shift in focus to the more readily available jobs centered on organized sports like soccer, softball, and basketball. I worked for the Augusta YMCA in Georgia, the Savannah River Site Operations Recreation Association in South Carolina, and the Town of Elkton in Virginia.

All of these jobs focused on service to members, with the program budgets set to barely cover the expenses. Being in recreation, however, meant working nights and weekends. With a growing young family, the hours didn’t work very well. So it’s no surprise that after moving back to the beautiful Rocky Mountains that I love, the transition to being an administrative assistant for RMOUG went fairly smooth. Taking care of membership wasn’t that much different from registration for sports teams.

At the time, RMOUG had completely blossomed from a small trade group to a State of Colorado Not-For-Profit Corporation. There were still growing pains, however, since we continued to use the tax identification number and insurance from the International Oracle Users Group. Even though I had classes in how to manage non-profits, my work experience had always been with hands-on programs. Getting our own tax identification number and going through a full audit the first few months on the job was a completely new experience. These events helped me develop a system that could easily be tracked by both me and hopefully anyone else who might come after me.

When I started working, the membership was just over nine hundred total members, a combination of Students, Individuals, and what the Membership Tracking System refers to as “Reps,” which are people that belonged through a corporate membership. As you can tell from the Total Membership chart, the numbers have changed quite a bit over time and are currently at their lowest point in the last thirteen years.

With membership under 1,000, the next challenge is to find ways to cut costs. Even though we are now a full non-profit 501(c)3 organization, most of our expenses have remained the same. That probably means we won’t be having any board meetings in Hawaii. Seriously, though, we need to evaluate why the membership has decreased. To do so, it might help to look at the changes that have occurred over the years.

The structure of membership for RMOUG is unique in that it has both individual members and company members. Student and Individual memberships always belong to the individual, no matter whom they work for or where they live. A company membership always belongs to the company, can be updated at any time, and unused slots can be used by other members of the company even if they aren’t on the roster. Company memberships can consist of one person on up. The highest number of “reps” for any company is currently 34. A “Company” membership usually has 1-3 members; a “Corp 1” membership is for up to 5 people; a “Corp 2” membership is set at 6-8 people; a “Corp 3” level is for 9-12 members; and a “Corp 4” is for any company with 13 or more participants.

When I began working in 1998, of the membership categories, 47% were individu-
Over the years, I’ve watched the membership demographics change somewhat with the economy. Throughout it all, our wonderful volunteers and the annual Training Days is the base that makes our user group so strong. The period around Training Days also marks the largest number of updates I make to the database as companies align their members to whom they plan to send to the conference.

Since Training Days is just around the corner, you can always check with me about your current membership status. If you have any suggestions on how we can increase our membership base, please let me know.

I really enjoy being a part-time contractor for RMOUG since most of the work can be done during the day, which leaves evenings and weekends free to enjoy my family. Of course, I try to sneak in a bike ride through Bear Creek Lake Park during lunch. Thursdays are my soup writing group day, and I am almost always working on some project, whether it’s my latest romance novel, scrapbooking, or reading a fun book. So if you need to reach me, emails usually work the best.

My husband, Darl, and I recently celebrated our 25th anniversary and we have two beautiful daughters; a nineteen-year-old working towards being a paramedic, and a fourteen-year-old in 8th grade. We all love the outdoors, especially when it comes to seeing wildlife. When we’re not tromping the back hills of Colorado looking for mountain lions, bobcats, bears, or coyotes, we’re at the gym watching our youngest participate in all those league and tournament basketball games that I used to organize.

Heidi Kuhn
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)棉花
☐ 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

Or Join Online:

http://www.rmoug.org/member.htm

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>Ad Format</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 all 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

Allison Leech
Secretary
E-mail: Secretary@rmoug.org

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

Kathy Robb
Board Member Emeritus
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

Carolyn Fryc
Programs Director
E-mail: ProgramsDir@rmoug.org

Tim Gorman
IS Director
Email: ISDir@rmoug.org

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

Thomas Green
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

Pat Van Buskirk
Newsletter Director
E-mail: NewsletterDir@rmoug.org
The PL/SQL Challenge

"Destined to Become World Famous"

Why? Just listen to what people are saying:

◆ "The quiz is fantastic – great learning opportunity – and now you can see past quizzes and their results even better. well done." - From Andre
◆ "Cool quiz to tickle the brain!" - From A.M.
◆ "The PL/SQL quiz is one of the highlights of my workday. Thank you for creating a professional culture for us PL/SQL developers!" - From Lauren
◆ "The PL/SQL Challenge has quickly become my caffeine companion in the mornings to help get the grey matter moving and the synapses firing. Even if I don't feel very intelligent after seeing some of the correct answers..." - From Matt G

But wait! There's more!
- Win fabulous prizes!
- Become the envy of your peers!
- Did we mention the fabulous prizes?

Play today! Play tomorrow. Heck, if you travel fast enough you can play yesterday!

The Challenge is simple: you play the quiz - daily. We keep score. You get ranked. Every three months, the top-ranking players compete in a quarterly championship to award first, second and third prizes. But that's not all! Every week and month, players win prizes based on the correctness of their answers and from participation raffles. The more you play, the greater your chance of winning.

So what are you waiting for?
It's time to take the PL/SQL Challenge!
To register and play, or for more information, visit www.plsqlchallenge.com.

www.plsqlchallenge.com
<table>
<thead>
<tr>
<th>Date</th>
<th>Event</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>12/14/10</td>
<td>Board Monthly Meeting</td>
<td>6:00pm</td>
</tr>
<tr>
<td>12/17/10</td>
<td>Newsletter - Mail</td>
<td>Winter Issue</td>
</tr>
<tr>
<td>1/11/11</td>
<td>Training Days</td>
<td>Training Days 2011 - Speaker Presentation Papers Due</td>
</tr>
<tr>
<td>1/12/11</td>
<td>Training Days</td>
<td>Training Days 2011 - Last day for Early Registration</td>
</tr>
<tr>
<td>1/13/11</td>
<td>DBLabs</td>
<td>DBClassify from Zettapoint @ 6:00pm - Regis University Denver</td>
</tr>
<tr>
<td>1/13/11</td>
<td>Training Days</td>
<td>Training Days 2011 - Standard registration begins</td>
</tr>
<tr>
<td>1/19/11</td>
<td>Board Monthly Meeting</td>
<td>Conference Call @ 6:00pm</td>
</tr>
<tr>
<td>2/1/11</td>
<td>Newsletter - Call</td>
<td>Articles Spring Issue</td>
</tr>
<tr>
<td>2/9/11</td>
<td>Training Days</td>
<td>Training Days 2011 - Last day for standard registration</td>
</tr>
<tr>
<td>2/10/11</td>
<td>Training Days</td>
<td>Training Days 2011 - Late Registration begins</td>
</tr>
<tr>
<td>2/15/11</td>
<td>Scholarship</td>
<td>Scholarship - Accepting Applications for Spring Scholarship</td>
</tr>
<tr>
<td>2/15-17/2011</td>
<td>Conference</td>
<td>Training Days 2011 Colorado Convention Center</td>
</tr>
<tr>
<td>2/17/11</td>
<td>Board Monthly Meeting</td>
<td>@ Training Days</td>
</tr>
<tr>
<td>2/15/11</td>
<td>Newsletter - Deadline</td>
<td>Articles Spring Issue</td>
</tr>
<tr>
<td>3/20/11</td>
<td>DBLabs</td>
<td>TBD - Regis University Denver</td>
</tr>
<tr>
<td>3/16/11</td>
<td>Board Monthly Meeting</td>
<td>Conference Call @ 6:00pm</td>
</tr>
<tr>
<td>3/18/11</td>
<td>Newsletter - Mail</td>
<td>Spring Issue</td>
</tr>
<tr>
<td>4/5/11</td>
<td>QEW - Call for Presentations</td>
<td></td>
</tr>
<tr>
<td>4/10-14/2011</td>
<td>Conference</td>
<td>Collaborate 2011 - Orange County Convention Center - Orlando Florida</td>
</tr>
<tr>
<td>4/22/11</td>
<td>Board Monthly Meeting</td>
<td>Corporate Offices @ 6:00pm</td>
</tr>
<tr>
<td>4/22/11</td>
<td>QEW - Preliminary Agenda</td>
<td></td>
</tr>
<tr>
<td>4/29/11</td>
<td>QEW - Deadline for</td>
<td>Presentations</td>
</tr>
<tr>
<td>4/29/11</td>
<td>Scholarship</td>
<td>Deadline for Applications for Spring Scholarship</td>
</tr>
<tr>
<td>5/20/11</td>
<td>DBLabs</td>
<td>TBD - Regis University Denver</td>
</tr>
<tr>
<td>5/1/11</td>
<td>Newsletter - Call</td>
<td>Articles Summer Issue</td>
</tr>
<tr>
<td>5/20/11</td>
<td>Board Meeting</td>
<td>7:00am before first QEW session</td>
</tr>
<tr>
<td>5/20/11</td>
<td>QEW</td>
<td>Quarterly Educational Workshop</td>
</tr>
<tr>
<td>5/20/11</td>
<td>Scholarship</td>
<td>Scholarship - Announce Spring Scholarship recipients at QEW</td>
</tr>
<tr>
<td>5/15/11</td>
<td>Newsletter - Deadline</td>
<td>Articles Summer Issue</td>
</tr>
<tr>
<td>6/15/11</td>
<td>Board Monthly Meeting</td>
<td>Corporate Offices @ 6:00pm</td>
</tr>
<tr>
<td>6/18/11</td>
<td>Newsletter - Mail</td>
<td>Summer Issue</td>
</tr>
<tr>
<td>7/20/11</td>
<td>DBLabs</td>
<td>Oracle Webs with Ron Bich - Regis University Denver</td>
</tr>
</tbody>
</table>

Please note dates are subject to change. For the most current events calendar visit our website at www.rmoug.org.

---

Tired of the I-25 Corridor?  
60 Acres In Gorgeous Bijou Basin  
Less Than An Hour From DTC

- Grassy Meadows & Pines
- Excavated Road and Building Site
- Power & Phone At Property Line
- Adjudicated Water Rights
- One Hour From DIA
- Possible Owner Carry W.A.C.

30777 Magic Dog Circle, Kiowa, CO  
$270,000

Pat Van Buskirk • Coldwell Banker, Parker Office  
(303) 243-0737 Cell (303) 841-5263 Office  
www.patvanbuskirk.com pat@patvanbuskirk.com
Join us for our next Quarterly Education Workshop in August at the Regis University Science Amphitheatre. 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 Carolyn Fryc, Programs Director at ProgramsDir@rmoug.org

Watch RMOUG’s Web Page for May Training Topics

www.rmoug.org
COST-EFFECTIVE ENTERPRISE CONTENT INTEGRATION
WITH ORACLE E-BUSINESS SUITE

With DocSavi, your users can access any ECM content right from Oracle EBS. Integrating with leading ECM suites, DocSavi works with any EBS module. The most exciting part is that DocSavi can be easily integrated into your environment. By leveraging existing Oracle infrastructure, no new hardware is required. DocSavi utilizes Oracle workflow so there are no systems to learn – ensuring high use adoption. It is also fully configurable: no coding is required!