Official Publication of the Northern California Oracle Users Group

NoCOUG JOURNAL

Vol. 27, No. 1 · FEBRUARY 2013

$15

Knowledge Happens

Be Very Afraid

An eye-opening interview with the CTO of McAfee.

See page 4.

We Don't Use Databases

Dream of freedom from the RDBMS.

See page 16.

Integrating Oracle and Hadoop

When an RDBMS is not enough.

See page 20.

Much more inside . . .

Thanking the Team

Take a moment to think about the huge amount of effort that goes into this publication. Your first thought might be about the care and attention required of the authors. Yes, writing is hard work. Now consider each author's years of hard-won experience; then add it up. The cumulative amount of time spent to acquire the knowledge printed in each issue is decades—maybe even centuries.

But let's take a moment to thank the people who make it possible for us to share this knowledge with you. Without the dedication and skill of our production team, all that we'd have is a jumble of Word files and a bunch of JPEGs. Copyeditor Karen Mead of Creative Solutions transforms our technobabble into readable English. Layout artist Kenneth Lockerbie and graphics guru Richard Repas give the *Journal* its professional layout.

Finally, what really distinguishes this *Journal* is that it is actually printed! Special thanks go to Jo Dziubek and Allen Hom of Andover Printing Services for making us more than just a magnetically recorded byte stream. ▲

—*NoCOUG Journal* Editor

Table of Contents

Publication Notices and Submission Format

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

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

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

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

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

2013 NoCOUG Board

ADVERTISING RATES

The *NoCOUG Journal* is published quarterly.

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

Personnel recruitment ads are not accepted.

journal@nocoug.org

Be Very Afraid

with Slavik Markovich

Slavik Markovich

Slavik Markovich is vice president and chief technology officer for Database Security at McAfee and has over 20 years of experience in infrastructure, security, and software development. Slavik co-founded Sentrigo, a developer of leading database security technology that was acquired by McAfee in April 2011. Prior to co-founding Sentrigo, Slavik served as VP R&D and chief architect at db@net, a leading IT architecture consultancy. Slavik has contributed to open-source projects, is a regular speaker at industry conferences, and is the creator of several open-source projects like Fuzzor (an Oracle fuzzer) and YAOPC (Yet Another Oracle Password Cracker). Slavik also regularly blogs about database security at **www.slaviks-blog.com**.

Is my financial and medical information safe from the bad guys? After watching* Die Hard 4*, I'm not so sure, because it seems that bad guys can access, change, or erase anybody's information with a few keystrokes.

Although life is not a movie, and the situation is not quite as bad as *Die Hard 4*, it is not that good either. You can read about breaches with varying degrees of severity every week. While the "bad guys" require a bit more than a few keystrokes to access/change information, they have very sophisticated tools at their service. World-spanning global botnets, automated hacking tools, a flourishing underground market, and a strong financial incentive all motivate the "bad guys" to continue breaking into systems.

On the flipside, there have been many significant changes and improvements to the applicable regulations associated with protection of PHI and ePHI healthcare information. In addition, the enhanced enforcement of HIPAA, and the newer HITECH, regulations has increased the visibility of—and, arguably, attention to—affected organizations complying with these regulatory mandates. SOX, GLBA, and other financial regulations are intended to address the integrity and authenticity of financial records. So, the organizations keeping your records are forced to think about security.

I would also add that it isn't always "the bad guys" that cause data compromise—sometimes it's caused accidentally, either by human, or system(s), error. To summarize, if you are being targeted, I'd say that there is a pretty good chance that the hackers will succeed in compromising your details. On the other hand, your liability is limited, at least on the financial front.

Why is information security so poor in general? Is it because administrators and users—me included—are clueless about information security, or is it because the operating systems, databases, networks, languages, and protocols are inherently vulnerable, which makes our task much harder than it really ought to be?

Indeed, there is a big awareness issue when it comes to security. Users, developers, and administrators generally lack deep understanding of security and, as everybody knows, security is only as strong as your weakest link. The "bad guy" just needs one successful try on a single attack vector, while the security protections need to cover all bases, all the time. It's an asymmetric game where currently the "bad guys" have the advantage.

When specifically talking about "database security," the reality is that the overall risk posture for these systems, and the often highly sensitive and/or business-critical information they contain, is most often grossly underestimated by the respective organizations. A comparison can be made to what the famous 1930s bank robber Willie Sutton was often quoted as saying, when asked by a reporter why he robbed banks: "Because that's where the money is." The "bad guys" often target these databases, and the valuable data assets they contain, because they know that's where they can get the biggest bang for their buck (i.e., the highest return for their exploit efforts).

Also, the associated risk to them of being caught and subsequently penalized is very often quite low combined with the associated payoff (return) being quite high. So from an ROI perspective, their motivating rationale is abundantly clear.

Finally, if you were indeed "clueless" about security, you probably wouldn't be asking these types of targeted questions.

> *"It is important, for security as much as for regulatory compliance reasons, to monitor and audit DBA activity. If you work in a bank vault, you know there are CCTV cameras on you. You want those cameras on you."*

The analogy is that certain cars are the favorites of car thieves because they are so easy to break into. Why are salted password hashes not the default? Why are buffer overflows permitted? Why was it so easy for China to divert all Internet traffic through its servers for 20 minutes in April 2010? Why is Windows so prone to viruses? Is it a conspiracy?

My motto is "always choose stupidity over conspiracy." It goes back to the issue of lack of awareness. Developers that are not constantly trained on security will introduce security issues like buffer overflows or passwords stored in clear text or encrypted instead of hashed with a salt, etc. Some protocols were not designed with security in mind, which makes them susceptible to manipulation. Some targets are definitely softer than others.

At an absolute minimum, measures should be taken to harden the respective systems, as per the individual vendors' guidelines and instructions. Unnecessary system services and processes should be disabled to reduce the attack surface, appropriate access control mechanisms should be properly configured, critical system patching should be done on a regular basis, etc.

But, unfortunately, these minimal security measures are often insufficient to address the rapidly expanding threat landscape. System visibility, in as near real time as possible, is required. Automated user process monitoring, vulnerability assessment, event correlation, and accompanying security policy notifications/alerting for these systems needs to be provided.

Is the cloud safe? Is SaaS safe?

I do not believe that the cloud or the SaaS model is inherently *more* or *less* safe—it is just a different *kind* of safe. Depending on the organizations' risk appetite, they can be provided with the appropriate safeguards and controls to make implementation of private and public cloud-based services correspondingly "safe." Technological controls, as well as organizational and administrative controls, need to be tailored for these types of deployments.

It's also critical that the database security model be extensible and scalable to accommodate virtual and cloud-based environments.

Do we need better laws or should we trust the "enlightened self-interest" of industry? Enlightened self-interest—the mantra of Fed chairman Alan Greenspan—didn't prevent the financial collapse. Will it prevent the digital equivalent of Pearl Harbor?

"Enlightened self-interest," by itself, is usually insufficient. At least it has been proven to be up to now. On the other hand, over-regulation would not be a good alternative, either. There has to be a happy medium—where government and private industry work together to promote a more secure environment for commercial transactions to occur, and where consumers' privacy is also protected. But, unfortunately, we're not there yet.

If not laws, how about some standards? Why aren't there templates for hardened operating systems, databases, and networks? Or are there?

There are numerous standards for applying security controls to these systems, including Center for Internet Security (CIS), which includes "hardening" benchmarks for a variety of different systems and devices, as well as the NIST 800 Series Special Publications that offer a very large set of documents addressing applicable policies, procedures, and guidelines for information security. In addition, most of the more significant IT product vendors provide specific hardening guidelines and instructions pertaining to their various products.

The problem is how to consistently measure and make sure that your systems do not deviate from the gold standard you set. Unfortunately, systems tend to deteriorate with use—parameters are changed, new credentials and permissions are introduced, etc. An organization without a consistent, proven way to scan systems is going to have issues no matter how close it follows the standards. A recent scan we did with a large enterprise discovered over 15,000 weak passwords in their databases. In theory, they followed very strict federal policies.

Who will guard the guards themselves? As an administrator, I have unlimited access to sensitive information. How can my employer protect itself from me?

There's a fundamental tenet in information security called "principle of least privilege," which basically says that a user should be given the necessary authorization to access the information they need to perform their tasks/job—but no more than that level of privileged access. In addition, there's another concept called "separation (or "segregation") of duties," which states that there should be more than one person required to complete a particular task, in order to help prevent potential error or fraud.

In the context of databases, this translates to not allowing users and administrators to have more access than is required for them to do their jobs—and for DBAs, that the DB administrative tasks will be monitored in real time and supervised by a different team, usually the information security team. A security framework that enforces these database access control policies is critical, because the inconvenient fact is, many compromises of DBs involve privileged access by trusted insiders.

While there is a much higher probability that someone who is not a DBA would try to breach the database, the DBA is in a much better position to succeed should he or she really want to do that.

If risk is the arithmetical product of the probability of an incident happening and the potential damage that incident could cause, then due to the latter factor, DBAs as well as other highly skilled insiders with access privileges pose a significant risk.

In 2007, Computerworld and other sources reported that a senior DBA at a subsidiary of Fidelity National Information Services Inc. sold 8.5 million records, including bank account and credit card details, to a data broker. An external hacker would find it very difficult to achieve this kind of scale without insider cooperation.

It is important, for security as much as for regulatory compliance reasons, to monitor and audit DBA activity. In fact, this should be done for all users who access the database. DBAs are the first to understand this. If you work in a bank vault, you know there are CCTV cameras on you. You want those cam-

> *“What DBAs should not accept are solutions that hinder or interfere with the DBA’s daily tasks—DBAs are primarily concerned with running databases efficiently. Any solution that jeopardizes this primary objective is counter-productive and doomed to fail anyway, because DBAs and other staff will find ways to circumvent it.”*

eras on you. DBAs are in a similar situation, and they understand this requirement completely.

What DBAs should not accept are solutions that hinder or interfere with the DBA’s daily tasks—DBAs are primarily concerned with running databases efficiently. Any solution that jeopardizes this primary objective is counter-productive and doomed to fail anyway, because DBAs and other staff will find ways to circumvent it.

At the risk of getting lynched by* Journal *readers, I have to ask your opinion about certification. Information Technology is the only profession whose practitioners are not subject to licensing and certification requirements. Can we really call ourselves “professionals” if we are not subject to any rules? Doesn’t the cost-benefit analysis favor licensing and certification? Even plumbers and manicurists in the state of California are subject to licensing and certification requirements but not IT professionals. Do you advocate security certification?

Well—while there’s certainly value in conducting user security awareness training and in promoting and achieving professional security certification, there are some issues. Like who would the accrediting body be? Who exactly needs to be certified? Will there be different levels of certification? Will each OS, DB, network device, application, etc., require its own distinct cert? It can quickly get very complicated.

But a shorter answer could be yes—I advocate security certifications.

In the novel* 1984*, George Orwell imagined that a device called a “telescreen” would allow “Big Brother” to listen to everything you said. The reality in 2013 is much worse since so much is digital, including my every message, phone call, and commercial transaction, and the cell phone is everybody’s personal electronic monitoring bracelet. What steps should we take to protect ourselves in this brave new digital world?

One possible answer might depend on how much security an individual is willing to trade for a potential reduction of features and functionality. For example, when “location services” are enabled on your phone, a variety of enhanced proximity-based services are then available, like several kinds of mapping services, driving directions and conditions, identification of nearby retail outlets, restaurants, gas stations, etc.

In addition, you can also locate your phone if it gets lost, wipe it of its contents, and/or have emergency services find you to provide help. But you also potentially get location-based advertisements, and there’s the specter of the device and application vendors (browser and service providers, too) aggregating and mining your various voice/data transmission location(s), for their own commercial purposes. The ongoing “privacy vs. commerce” battles involved in the “Do Not Track” discussions are good examples of these often-conflicting forces.

My personal assumption is that anything I publish on any network (text message, Facebook, Twitter, etc.) is public, no matter what settings it is published with. If I want to keep something private, I encrypt it. But, I’m willing to make privacy sacrifices in the name of convenience. I do use GPS; I do use Facebook and LinkedIn, etc.

Thank you for spending so much time with us today. Would you like to tell* Journal *readers a little about today’s McAfee? What are your current products? What is in the pipeline?

Well, I’m glad you asked. The **McAfee Database Security** solution comprises a core set of three products that serve to scan, monitor, and secure databases:

- **McAfee Vulnerability Manager for Databases**, which automatically discovers databases on the network, detects sensitive information in them, determines if the latest patches have been applied, and performs more than 4,700 vulnerability checks.
- **McAfee Database Activity Monitoring**, which provides automatic, non-intrusive, and real-time protection for heterogeneous database environments on your network with a set of preconfigured security defenses, and also provides the ability to easily create custom security policies based on configurable, and very granular, controls. In addition, it has the capability to deliver virtual patching updates on a regular basis to protect from known vulnerabilities.
- **McAfee Virtual Patching for Databases (vPatch)**, which protects unpatched databases from known vulnerabilities and all database servers from zero-day attacks based on common threat vectors, without having to take the database offline to patch it. Additionally, vPatch has been accepted as a “compensating control” in compliance audits.

The McAfee Database Security solution is also tightly integrated with McAfee’s centralized security management platform, **ePolicy Orchestrator (ePO)**, which consolidates enterprise-wide security visibility and control across a wide variety of heterogeneous systems, networks, data, and compliance solutions.

At McAfee, we do not believe in a silver bullet product approach. No security measure can protect against all attacks or threats. However, **McAfee’s Security Connected** framework enables integration of multiple products, services, and partnerships for centralized, efficient, and effective security and risk management. ▲

All Indicators Are Green

by Naren Nagtode

Naren Nagtode

According to U.S. News & World Report, four of the top ten U.S. jobs for 2013 are in the IT and database fields. Number 4 in the list is Software Engineer, number 6 is Database Administrator, number 7 is Software Engineer, and number 9 is Web Developer. If you are reading this journal, you are part of the lucky bunch!

The economic recovery is being led by the technology sector. Those who keep up with technology will be the ones who will benefit. NoCOUG offers high-quality education and training in Oracle- and database-related technologies at an unbelievably small membership cost. You can keep updated with the latest features, current trends, and best practices; mingle with other Oracle professionals; and learn from gurus and peers.

NoCOUG made it through the tough economy and is emerging stronger. While many user groups closed, we not only survived, we came out stronger and delivered a series of successful conferences and training days. So what is the secret sauce of NoCOUG? It is the committed and strong board and volunteers.

I would like to thank the board for the enormous amount of effort and success achieved in the past year. All indicators are green, including memberships, attendance, and finances. In addition to organizing four conferences and two seminars and publishing four issues of the *NoCOUG Journal*, the board successfully implemented a membership management system and a conference management system.

I would also like to thank the board for placing trust in me and electing me as the new president. I will be helped by Vice President Hanan Hit; Secretary and Treasurer Dharmendra (DK) Rai; Membership Director Alan Williams; Conference Director Ben Prusinski; Track Leaders Jimmy Brock, Abbulu Dulapalli, and Nasreen Aminifard; Vendor Coordinator Omar Anwar; Training Director Randy Samberg; Meetup Coordinator Gwen Shapira; Webmaster Eric Hutchinson; Journal Editor Iggy Fernandez; Marketing Director Scott Neely; and IOUG Liaison Kyle Hailey.

The upcoming conference at Oracle Conference Center on February 21 will feature an entire track on NoSQL and Big Data topics. Dave Rubin will kick off the conference with a keynote address, titled *"Oracle NoSQL Database and Oracle Database: A Perfect Fit."* I'll see you there! ▲

The new Board of Directors. Back row: Abbulu Dulapalli, Jimmy Brock, Dharmendra (DK) Rai, Kyle Hailey, Alan Williams, Hanan Hit, Omar Anwar. Front row: Iggy Fernandez, Randy Samberg, Eric Hutchinson, Nasreen Aminifard, Naren Nagtode. Not in picture: Gwen Shapira, Scott Neely, Ben Prusinski.

Thirteen Ways to Make Your Oracle Database More Secure

Mike Dean

by Mike Dean

In my experience, database security is often overlooked, misunderstood, and generally ignored. Corporations will spend tons of money and time to address database design, performance, scalability, and availability, but security always seems to be an afterthought with not enough resources devoted to it.

The reason for this apparent lack of concern seems to fall into four schools of thought:

- *"I just paid a huge amount of money for Oracle, and it should already be secure."*
- *"My database is behind a company firewall, so I am not worried."*
- *"Nobody knows or cares enough about my data to bother stealing it."*
- *"I pay my DBAs a lot of money. I assume they are taking care of it."*

In reality, none of these could be further from the truth. Database security is not automatic, and a firewall will only provide a thin layer of defense. A misconfigured firewall will provide no layer of defense. From common thieves looking for credit card numbers to industrial spies looking to steal corporate secrets to international spies looking to steal government secrets, there are people out there actively searching for information to steal. Don't be so naïve as to think no one will find your data interesting or valuable. Or, maybe they don't want to steal your data but wouldn't mind knocking your website off the Internet for a while. If a hacker manages to crash the database that supports your e-commerce website, then that will certainly cost you time, money, and customers. On the final point, I think most DBAs do not spend very much time actively trying to secure their databases. This is usually through no fault of their own, as it is management that determines priorities, tasks, and funding that do not allow adequate time or resources for this type of work. It is, however, the DBAs' responsibility to educate themselves about security issues and how they may impact the databases for which they are responsible. It is also the DBAs' responsibility to make management aware of security issues and their potential impact.

This paper specifically addresses Oracle database security, but many of these ideas are applicable to any type of database and even to IT security in general. This is not intended to be a complete guide to securing Oracle databases but just some high-level ideas and examples. For a complete Oracle security checklist, you can download the latest from the Center for Internet Security at **http://www.cisecurity.org**. These suggestions are in no particular order of importance.

I make no claims that I came up with all of these ideas by myself. There are many Oracle professionals out there that have done years of research into this subject, and this paper is a compilation of my experience along with the ideas of many others. I have spent more than 15 years working with Oracle databases; the majority of these years have been spent as a production DBA on mission-critical, highly classified databases for the United States Department of Defense.

Disclaimer: This paper contains my own opinions, which I believe to be valid; however, I make no guarantee that implementing them will prevent your database from being stolen by bad guys.

Insist on strong passwords that are changed on a regular basis

Individuals should have their own accounts protected by strong passwords, and the sharing of accounts should be forbidden. The password strength and expiration policy needs to be enforced by the database in the form of password profiles and the password-verify function (available as of 10*g*). As of 11*g*, Oracle can enforce case sensitivity in passwords. Passwords should expire on a regular basis and be locked after a certain number of failed login attempts. The one exception to this rule would be for accounts used by applications to connect to the database. You can leave yourself vulnerable to a denial-of-service attack if someone repeatedly tries to connect with a valid username but an invalid password, and manages to lock that account.

I recommend that you go the extra step to enforce this policy by using a password cracker on a regular basis to identify weak passwords. One such free tool that I have used with success is called "woraauthbf"—written by Laszlo Toth—and is available at **http://www.soonerorlater.hu**.

I realize that changing database passwords on a regular basis can be a nightmare. With multiple accounts in multiple databases, it can be such a challenge that I think most organizations fail to accomplish it. Implementing a centralized user management system seems to be the ultimate solution. Oracle does this in the form of Enterprise Users that are managed via Oracle Internet Directory, which is Oracle's LDAP directory.

For more details about Enterprise User Management, refer to the Oracle Database Enterprise User Administrator's Guide at **http://docs.oracle.com/cd/B19306_01/network.102/b14269/toc.htm**. You can also check out a white paper that I wrote on the subject: *Implementing Oracle 11*g *Enterprise User Security*. (**http://www.dbspecialists.com/presentations.html#implementing_oracle_11g_enterprise_user_security**)

Beware of plain-text passwords

You can have a super-strong password, but if it is sitting in an unprotected text file, it isn't very secure. Make sure that you understand and document all of the configuration files that contain passwords so they can be checked for proper permissions. Make sure that they are not world-readable and exist only where needed. You will also need to know where these files are so you can change the passwords on a regular basis. In addition, be aware that passwords can sometimes be visible to certain OS commands.

For example, if you run a SQL script at the command line like this: "sqlplus system/oracle@orcl @scriptname.sql" then the entire command will be visible to anyone that happens to be logged onto the server when the script is run. In Windows, "tasklist -v" will display the username/password, and on Unix, the "ps" command will do the same. (It doesn't seem to be an issue on Linux.) According to Oracle Support document 557534.1, this behavior *"is not related to sqlplus. It is how the shell interprets the command and provides the details about the process."* So perhaps some flavors of Unix have fixed this issue as well. There are other executables (expdp, exp, sqlldr, etc.) that can take passwords at the command line, so they may or may not be vulnerable as well.

One way to make sure that you don't have this problem is to run the SQL script like this: "sqlplus /nolog @scriptname.sql" and have "connect system/oracle@orcl" in the SQL script itself; then the username and password will not be visible. (Make sure you have proper permissions on the script.) An even more secure solution would be to use OS Authentication or Secure External Password Store, and then you can run the script without specifying a password at all.

Secure the perimeter and everything that leads to the database

In order to have a secure database, you need a secure database server. For that, you need a secure application server and network and firewall, etc., etc.—you get the point. Your database itself can be rock solid, but if someone hacks the server and logs in as "oracle," they pretty much own your database. A vulnerable web server can allow a hacker to gain access to the network. Once inside the network, they may be able to poke around long enough to find a way into your database server. Make sure that every path that leads to the database is just as secure as the database itself.

A properly configured firewall is the first line of defense to keep hackers out of your network. You can go a step further by implementing Validnode Checking, which is a feature of Oracle that acts as an Access Control List for IP addresses that are allowed (or denied) access to your database. While this is not a substitute for a good firewall, it will add an extra layer of protection.

Never expose your database directly to the Internet. I know that it is really convenient to be able to SQLPlus directly into the database from your home computer, but it is very insecure and just asking for trouble. You can (and should) use a Port Scanning tool such as Nmap (**http://www.nmap.org**) to detect open SQLNet network ports (1521, 1522, and 1526, among others). If possible, avoid using the standard SQLNet ports.

Practice the principle of least privilege

Users should be granted the privileges to access the data that they need in order to do their jobs and nothing more. This will invariably mean more work for the DBA to determine access requirements and do individual grants, but it is truly necessary. As a DBA, you should know which users require which access, and it should be documented. I have frequently seen new applications come out of development destined for production where one of the "requirements" is that the application user has DBA privileges. This is horrible security practice and should never be allowed in production.

Hire trustworthy people

This may seem like an obvious point, but its importance cannot be overstated. DBAs, system administrators, network administrators, and various other people inside and outside of the IT department all have access to sensitive information. Thorough background checks should be a standard part of the hiring process, and this goes for both employees and contractors. When I worked at a large defense contractor, all employees were subject to rigorous criminal and financial background investigations and drug testing. Contractors, on the other hand, could basically come in off the street and start working almost immediately. The assumption was that the contracting companies were doing the background investigations. This turned out not to be true in the case of a contractor that was hired as a software engineer and then later fired when it was discovered that he had a previous conviction for theft and fraud.

Use database auditing

The auditing functionality that is available in Oracle is vast and very powerful. Auditing is an absolute must for every Oracle database in which security is a concern. Simply put, if you are not even auditing your database activity, then you are really not serious about database security. Auditing won't always prevent an intrusion or data theft, but it can certainly be used to detect one and provide valuable forensic evidence after the fact. In my experience, there is minimal overhead imposed by auditing, although you will need to manage the size of the audit trail, as it can grow quite large.

For the most part, you want to audit every command that fails and only a handful of commands that succeed. For example, you would want to audit all DDL statements, logon/

logoff activity, and "alter" commands. You certainly don't want to audit every successful select statement! In some cases, however, you may want to monitor all activity against certain tables that contain sensitive information.

Once you start auditing your database, you should monitor it on a regular basis to look for anomalies. Look for things like failed login attempts, "Insufficient Privileges" errors, and "Table or View Does Not Exist" errors. These can all indicate that someone is poking around in your database looking for trouble.

For complete details on how to implement auditing, refer to the Oracle Database Security Guide, available at **http://docs.oracle.com/cd/B19306_01/network.102/b14266/toc.htm**. You can also check out a blog that I wrote about auditing: *Overview of Oracle Auditing.* (**http://www.dbspecialists.com/blog/best-practices/overview-of-oracle-auditing/**)

Protect your backups

Stealing a backup of your database is as good as getting the database itself. Encrypt your backups and make sure they are stored securely.

Stay current with Oracle versions and apply Critical Patch Updates on a regular basis

Oracle has been improving the security of its software for many years, and you should try to stay relatively current. I realize that many people are running older versions of Oracle and are hesitant to upgrade, but you are truly vulnerable to many different attacks if you have an older version. Oracle releases CPUs every quarter to address known security problems. In order to protect yourself from these vulnerabilities, you should apply these patches on a regular basis. These vulnerabilities are real and can be exploited to gain unauthorized access.

Use bind variables

In addition to the importance of using bind variables for performance and scalability, they are critical for database security by preventing SQL injection. There is a great discussion about this issue by Tom Kyte at **http://asktom.oracle.com/pls/apex/f?p=100:11:0::::P11_QUESTION_ID:23863706595353**.

Only install and configure software that is actually needed

This will reduce the overall attack surface for your database. For example, unless you actually run external procedures from within the database, you should disable the EXTPROC listener configuration. If you are not using XML Database, then don't install it, and you won't have to worry about the network ports that it opens.

Monitor your database security on a regular basis

Over time, things can change, sometimes without your awareness, that will leave your database vulnerable. A user that once had a really secure password may have changed it to "password" and put it on a sticky note on their desk. A firewall that was once airtight may now have huge holes in it. Privileges that you locked down last year may have all been unlocked by the application of a patchset or a mistake by another DBA. It is important to regularly audit your database security posture in order to know that you are still secure.

Know your data and use encryption when necessary

Within your database, there is likely some data that is extremely sensitive, and you should consider using encryption to make sure it stays secure. Being able to identify the data that needs this extra level of protection is the first step. Financial information, personnel data, and classified information are all good candidates for encryption.

Oracle offers many different ways to encrypt data, both inside and outside the database, using the separately licensed Advanced Security Option. Transparent Data Encryption (TDE) can be used to encrypt data in the database and will automatically decrypt it when queried by anyone with appropriate privileges, making it transparent to the application. This will protect the data as it sits in the datafiles but won't protect it from users. In other words, if someone steals your actual datafiles, they wouldn't be able to read the data because it is encrypted, but if someone finds a DBA username/password lying around and logs in, they will be able to read the data.

You can also encrypt SQL*Net traffic to and from your database with the Advanced Security Option. This can be set up via the NetManager GUI or by setting various SQLNET.ENCRYPTION* and SQLNET.CRYPTO* parameters in the sqlnet.ora on both client and server. More details on all of this can be found in the "Advanced Security Administrators Guide" at **http://docs.oracle.com/cd/B19306_01/network.102/b14268/toc.htm**.

Use a security checklist

When it comes to actually hardening your database, it is important to have a methodical, repeatable approach by which you can accurately assess your overall database security posture. The Center for Internet Security (**http://www.cisecurity.org**) publishes and maintains security checklists for Oracle versions 8, 9, 10, and 11. These documents cover a wide variety of database security issues, from default init.ora parameters that should be changed to PUBLIC grants that should be revoked.

Read these documents and start to think about whether your database is in compliance. I would be willing to bet that it isn't. Some of the items seem impractical and perhaps even impossible to implement in every database, but it is certainly a worthy goal to try to come as close as possible to 100% compliance. This document (or one like it—there are others out there) should serve as your baseline for database security. ▲

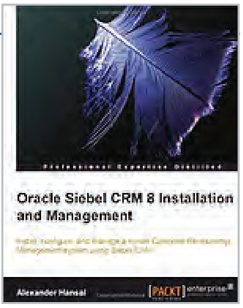
Mike Dean is an Oracle Certified Professional who has been working with Oracle databases since 1996, mostly as a production DBA on government and commercial projects in the Washington, D.C., and the Northern Virginia area. Since 2011, he has worked with Database Specialists as a senior staff consultant, helping customers with a wide variety of issues on their mission-critical systems. Mike can be reached by email at **mdean@dbspecialists.com**.

HELP

Oracle Siebel CRM 8 Installation and Management

A Book Review by Brian Hitchcock

Details



Authors: Alexander Hansal

ISBN: 978-1-849680-56-1

Pages: 560

Year of Publication: 2010

Edition: 1

List Price: $70

Publisher: Packt Publishing

Overall Review: A thorough introduction to the architecture and management of Siebel CRM 8.

Target Audience: Anyone new to Oracle Siebel CRM 8.

Would you recommend this book to others: Yes.

Who will get the most from this book? Administrators.

Is this book platform specific: No.

Why did I obtain this book? My group at work was preparing to support Siebel CRM 8 for the first time.

Overall Review

Everyone brings a context to everything they do. For this book review, my context is that I've never worked on a Siebel CRM system before. I've heard some things about it over the years, but I've never actually worked with such an environment. Given that, I'm highly qualified to review this book from the viewpoint of someone that needs to have a high-level overview and come up to speed on the basics as fast as possible. By the same token, this also means that I can't offer a review from the perspective of someone who has many years of experience administering Siebel CRM.

Given my context, this book delivered exactly what I needed. After reading this book, I can tell you about the components of a Siebel CRM system and some things I would look for if I were asked to troubleshoot such a system. This leads to another important point: It is always good to get some exposure to new things—in this case, software products that I haven't seen before. I may not end up working with Siebel full time, but learning how the product works has given me a fresh perspective on the other Oracle products that I do work with all the time.

Because I'm not a Windows Administrator, I can't comment on the sections of the book that discuss Siebel on Windows. My experience is with Linux/UNIX systems. I was expecting to see some discussion of Siebel CRM on virtual machines (VM) and specifically on Oracle Virtual Machines (OVM), but it never came up. Since I am working with OVM more and more often, I would have liked to have seen this discussed. As each aspect of Siebel was discussed, there was a section with information about how to verify the installation. This is very good information that is not often included. I was able to find related information using the index, which is also a good thing. Similarly, at the end of many sections of the book, a URL was given linking to the relevant official Siebel documentation. One negative observation is that some of the figures were primitive. In two cases it is difficult to read the labels of the system components in the figure because the text is obscured by the lines of the figure.

Chapter 1—Introducing the Siebel Web Architecture

The high-level architecture is discussed. Since I'm new to Siebel and the discussion was all about Siebel, I wasn't sure if I should assume that this meant Oracle Siebel CRM or not. It became clearer as I got through more of the book, but initially, I wasn't really sure that "Siebel CRM" was the same as "Oracle Siebel CRM." Because I have been working with Oracle Fusion Applications recently, I would have liked to see the author compare and contrast Siebel CRM with the CRM components of Fusion Applications. I was also unclear about how the licensing for Oracle Siebel CRM works. Several components of Siebel CRM are Sybase products, for example. Can I assume that a license for Oracle Siebel CRM includes all the needed licenses for the Sybase software products? Overall, I would have liked to see some discussion of the timeline for Siebel CRM as it moved from a Siebel product to an Oracle product, and the various changes that were made to the product along the way, including licensing issues. I was fascinated to learn that Siebel does not use the RDBMS for constraints; these are all handled in the application software. I had heard this before but wasn't sure if it was true until I read about it here. That seems like a throwback to a time long ago. The diagram of the Siebel Web Architecture was good. I have long been confused about the "Siebel Enterprise Server" versus the "Siebel Server," and now I know that the Enterprise Server is a logical term for the entire system, which can have one or more Siebel Servers executing.

The discussion also covers configuration parameters that have been stored in files in the past but have been moving into the database. The various components of the Siebel Enterprise are presented. I learned that the Siebel Web Templates contain

HTML tags that are proprietary. I found this fascinating. For the end user to see all the data inside Siebel CRM, the user must use the HI mode, which stands for the "High Interactivity mode, and this mode is only supported using Microsoft Internet Explorer. I would have liked to see a discussion of the plans for the Siebel product. I assume it will move away from proprietary HTML commands and away from supporting only one web browser, but this isn't made clear.

The chapter ends with a summary wherein the author tells us that *"installing Siebel CRM is a complex endeavor that involves multiple professionals."* It seems to me that software is always getting more complex. After all these years, why isn't it becoming less complex?

Chapter 2—Planning and Preparing the Installation

This chapter covers the steps needed to prepare and execute the installation of Siebel CRM, the various components involved, and various license key issues.

The author begins this chapter with the following: *"Implementing Siebel CRM for thousands of users in corporations that do business across the globe is not something a single person would do on a single day."* This reinforces the point that Siebel CRM is a complex product. We are then warned that many issues with Siebel CRM projects can be traced back to a lack of planning. I wondered if this is unique to Siebel, but that isn't discussed. My experience is that all large systems have issues that go back to insufficient planning. Another way this happens is that the project requirements change while the planning is happening, and at some point something has to get built.

Steps are presented for the reader to set up a Siebel development system to be able to follow along with the discussions in the text. This development environment requires a host machine with a VM and fully licensed MS Windows. Since I don't have anything like this available to me, I didn't attempt to set up this development environment.

Chapter 3—Installing Siebel CRM Server Software on Microsoft Windows

Here we learn how to install the software on MS Windows servers. This can be done in a GUI mode and a console mode. The GUI installer is used to install most but not all of the Siebel components. The specific inputs needed by the GUI for each component are discussed. Specific advice is presented, such as the need to install the database server utilities on the same machine as one of the Siebel servers. The level of detail provided for each component is good. To integrate Siebel CRM with other products, such as MS BizTalk and Oracle Enterprise Business Suite (EBS), requires installing EAI connectors. The Siebel Web Server Extension is installed using a separate installer and must be installed on each individual server where a web server will run. The Environment Verification Tool (EVT) is provided to check that everything, including patches for the Siebel server software, is in place.

Chapter 4—Configuring Siebel Server Software on Microsoft Windows

Having discussed installation on MS Windows, this chapter covers how to configure the installed software. This is done with the Siebel Software Configuration Wizard, which also validates various configuration parameters. This wizard runs in two phases, gathering the inputs and then configuring the software. The Siebel Gateway Name Server must be fully installed before any other part of the Siebel Enterprise Server can be set up. You can set up multiple Enterprise Servers to use a single Name Server, but this isn't supported. The Name Server is the heart of the Siebel CRM installation. It can use either the database or LDAP to authenticate users. By default, the database is used, but MS Active Directory is another option. Configuring the database requires executing a SQL script and other utilities to set up the needed tablespace names, user passwords, and additional user accounts. This process also creates tables, indexes, functions, and procedures, and imports seed data.

Chapter 5—Installing and Configuring Siebel CRM Server Software on Linux

This chapter covers the same installation and configuration issues as the previous two chapters, only this time on the Linux platform. There are many specific details that are different for Linux. We are told to use a non-root user account for the install steps, and I'm curious about what most installations use. My guess would be "siebel," but we aren't told. The Siebel Gateway Name Server is also known as the name daemon on Linux. Configuring the Siebel Enterprise on Linux creates the ODBC data source, which is defined in a hidden file .odbc.ini.

Details are given for installing the database schemas and seed data, configuring the Siebel Servers, the Siebel Web Server Extension (SWSE), and starting all the various components on Linux.

Chapter 6—Installing Siebel Client Software

Most business users will connect to Siebel CRM using the Siebel Web Client that uses files downloaded by the user's browser. All users have access to the Siebel Web Client when connected to the corporate network. For users that don't have a full-time Internet connection, there is the Siebel Client Software, which requires installing software and configuring other components. The differences between—and the confusion about—Developer Web Client and Mobile Web Client are covered. Siebel Client Software is only supported using MS Internet Explorer. I would hope this limitation is going away soon, but that isn't discussed. A Siebel sample database must be installed to support the Client Software. This sample database is a Sybase Adaptive Server Anywhere database, which makes me wonder again about license issues; it is installed not by the Oracle Universal Installer (OUI) but by using InstallShield. The application of Client Software patches, which does use OUI, is discussed.

Chapter 7—Installing Ancillary Siebel Server Software

The "ancillary" software turns out to be Visual Mining's NetCharts server and Business Intelligence Publisher (BI Publisher). I have not worked with Visual Mining's NetCharts before. This product creates charts to help visualize the data from web-based applications. We are told that NetCharts is a lower-cost option compared to BI Publisher. I would have liked some comments on how this will be handled in the future.

Will NetCharts continue to be an option, or will BI Publisher be the only choice? The author also mentions that many customers choose to export data into MS Excel and points out that this is a lengthy, insecure, and error-prone process. I agree.

The steps to install and configure both NetCharts and BI Publisher are covered. This includes setting up needed Siebel Enterprise parameters.

A useful history is presented that explains how BI Publisher came to replace Actuate Report Server. I found the summary of what BI Publisher actually does to be very good. Among other things, BI Publisher supports multiple heterogeneous data sources and is built using pure Java and XML.

Chapter 8—Special Siebel Server Configurations

The Siebel Enterprise Server is a logical construct of multiple software components, one of which is the Siebel Server. Each Enterprise Server will have one or more Siebel Servers to handle the processing. This chapter covers the installation and configuration of multiple Siebel Servers. Setting up multiple Siebel Servers is usually done in production environments to support scalability and failover, and to minimize downtime during deployments. The author recommends that we only install one Siebel Server on each server. Since virtual servers are not specifically addressed, I don't know if that would include multiple VMs on a single physical server, with a single Siebel Server on each VM.

Three different kinds of load balancing are supported in Siebel. Single Siebel Server, Siebel Native Load Balancing, and Third-Party Load balancing are all described, along with the configuration details for each. The SWSE also needs additional configuration to support load balancing. This chapter ends by describing the process of installing additional language packs. I found it interesting that not all system messages are translated, so we must always install the English language pack first before installing any other languages.

Chapter 9—Siebel Server Management

Once the Siebel CRM environment is installed and configured, it needs to be maintained. This requires understanding all of the different servers and components, and all of the parameters affecting them. The various server management screens that appear in the Siebel client are shown as well as the command-line utilities available that perform the same functions. The various software components are grouped to simplify their management, and the most important groups are identified. There are many parameters in Siebel, and collections of parameters are called "profiles." The parameters are organized into the enterprise hierarchy. Parameters in one level inherit all the parameters from the lower levels. This means that it is important to be careful when changing parameters, as the change can propagate and affect other parameters.

Two standard administrative user screens are shipped with each standard Siebel application, one for server configuration and one for server management. The data displayed in these screens is stored in the Siebel Gateway Name Server, and we can query the Name Server to retrieve specific information. However, the Name Server is not an RDBMS, and we are cautioned not to user SQL wildcards when retrieving data from the Name Server.

Chapter 10—User Authentication

Siebel CRM offers several ways to authenticate users. The default is database authentication. This requires that every Siebel user also have a database user created. Other options are LDAP and Web SSO. For LDAP, MS Active Directory is also supported. Siebel CRM also provides a software development kit (SDK) that supports custom authentication solutions. The author advises that *"avoiding redundant user accounts across various systems is the most crucial aspect of professional user administration,"* without further discussion. I would ask if this is more important than user password security. The main authentication component is the Security Adapter, which is configured for the chosen authentication. If database authentication is chosen, you can use the same RDBMS that contains the Siebel data or a separate database. Whatever the configuration details may be, the user accounts and passwords are always managed outside of Siebel CRM.

Chapter 11—User Authentication and Access Control

Having discussed user authentication in the previous chapter, we move on to look at how access control works in Siebel CRM. This refers to restricting access to Siebel views, customer data, and master data. Siebel views are the actual web pages a user sees. Views, along with groups of users, are associated with responsibilities. For mobile users, views that allow access to large data sets should not be available when the user is working offline to prevent performance issues.

Note that controlling access to customer data is not handled by the database. The data is in the database and the details of the access controls are stored in tables in the database, but the access control is not based on any mechanism built into the database itself. Data that is relatively static, such as product information, is referred to as "master data." Whole sets of data can be set up as catalogs and categories, which makes it easier to grant access to one or more users. It is confusing that Siebel can control access to views based on conditional expressions using the Siebel Query Language. Yes, that would be "SQL" but not the SQL you are used to thinking of.

Chapter 12—Managing User Accounts

Managing users in Siebel CRM requires understanding divisions and organizations and how to set them up. I'm not clear what the impact would be of a major reorganization. Someone has to update all of the information to move people around. Is there a process for this? This isn't addressed. This chapter covers setup and management of the position hierarchy; setup of user and employee accounts; and setup of divisions, organizations, and user and employee accounts.

Since the authentication of users is handled outside Siebel, you also need to create user accounts in the database or LDAP. If database authentication is used, this means creating a database user for each Siebel CRM. If LDAP is used, we are told that *"info entered in Siebel screens will be propagated to the directory server and a new directory server account will be created automatically."* Really? What is the mechanism for this? Does this happen in real time or due to some batch update? Finally, if a user is removed from LDAP, does that change get propagated back into Siebel? These subjects aren't addressed.

Chapter 13—Siebel Remote and the Siebel Development Environment

Siebel Remote is a module that synchronizes local databases with the central Siebel database. This is used even if a customer doesn't deploy mobile clients, because developers still need it to set up local workspace for development work. Synchronization involves exchanging transaction information between the Siebel database and the local database. The database stores information about Siebel transactions in tables. Don't get confused—these are Siebel transactions, not the database transactions. The central and local databases are synchronized by exchanging files. These files are generated by reading the transaction information from each database. There is no discussion of conflict issues or how they are resolved. If the same data is changed in both the central and the local database, how does the synchronization process handle this? Perhaps this issue doesn't come up. I would also like to have seen some comments on the size of these transaction files and how long the synchronization process typically takes. For a large organization, I would expect the files would be large and synchronization would take a long time.

Chapter 14—Installing and Configuring the Siebel Management Server Infrastructure

The Siebel Management Server Infrastructure consists of the Siebel Management Server and the Management Agent. This infrastructure is currently used by two Siebel modules: the Application Deployment Manager (ADM) and the Siebel Diagnostic Tool. ADM is used to migrate configuration changes from one Siebel CRM environment to another. The Diagnostic Tool is discussed in a later chapter. This chapter provides an overview of the infrastructure as well as installing and configuring the components. The prerequisites for the Siebel Management Server are presented, as is installation on MS Windows. Finally, the steps to install and configure the Siebel Management Agent are discussed. The Management Agent software must be installed on each machine that runs Siebel Server.

Chapter 15—Migrating Configuration Changes between Environments

This topic comes up when, for example, you need to transport configuration changes from a development environment to test and production environments. First we learn how to deploy the Application Deployment Manager (ADM). Then we see how to extract administrative data which are grouped into data objects. Examples include data for List of Values, responsibilities, positions, and organizations. The major administrative objects are described as well as the ADM architecture. To extract changes from the source environment, we generate the ADM package, which is then copied and applied to the target. Other migration utilities are explained.

The author ends this chapter with the following statement: *"The complete, flawless, and reliable deployment of configuration changes from one Siebel enterprise to another is a complex endeavor that cannot be accomplished with a single tool."* I'm becoming convinced that Siebel administration is not for the faint of heart!

Chapter 16—Monitoring Siebel Applications

This chapter describes how complex the Siebel environment is, due to interwoven architecture, heterogeneous hardware, software from various vendors, and large numbers of users. The Server Component Event Logging is presented, which includes Siebel Application Response Measurement (SARM). Siebel software components include instrumentation points that gather timing information as requests flow through the system. Once SARM is enabled, sarmquery is used to examine the log files that are generated. A command-line utility is provided to select information from the log files. The sarmquery utility uses syntax that is unique but looks very much like SQL. The Siebel Diagnostic Tool takes SARM data and outputs charts and tabular data. SARM only provides data on the server side. Siebel also provides client-side logging, which is used to gather performance data for the client, and the client log file is discussed.

"Implementing Siebel CRM for thousands of users in corporations that do business across the globe is not something a single person would do on a single day."

Conclusion

This book provided what I needed. Since I did not have any previous experience with Siebel CRM, I needed a high-level review of all of the major components along with a brief description of what each one does. I was struck by the complexity of the product, but that may be because it is new to me. The other products in the Oracle Applications Unlimited product suite are complicated. Imagine trying to take all the pieces and parts of Siebel CRM and Oracle EBS and all the other products, combining them all, and rewriting them all in Java to use a common interface. That's the challenge of Fusion Applications. ▲

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

We Don't Use Databases; We Don't Use Indexes

Iggy Fernandez

by Iggy Fernandez

Whenever salespeople phone Mogens Norgaard, he puts them off by saying that he doesn't use the products that they are calling about.

When the office furniture company phones, he says "*We don't use office furniture.*" When the newspaper company phones, he says "*We don't read newspapers.*" When the Girl Scouts phone, he probably says "*We don't eat cookies.*"

Once he got a phone call from the *phone* company.

You can only imagine how that conversation went. Read the whole story at **http://wedonotuse.com/stories-and-answers.aspx**.

If a database vendor phoned, I can imagine Mogens saying, "*We do not use databases. We do not use indexes. We store all our data in compressed text files. Each compressed text file contains one year of data for one location. There is a separate subdirectory for each year. We have a terabyte of data going back to 1901, so we currently have 113 subdirectories. The performance is just fine, thank you.*"

On second thought, that's just too far-fetched.

You see, back in the early days of the relational era, the creator of relational theory, Dr. Edward Codd, married relational theory with transactional database management systems (a.k.a. ACID DBMS), and the Relational Database Management System (RDBMS) was born. He authored two influential *Computerworld* articles—"Is Your DBMS Really Relational?" (October 14, 1985) and "Does Your DBMS Run by the Rules?" (October 21, 1985)—that set the direction of the relational movement for the next quarter-century. Today, the full declarative power of "data base sublanguages" such as Structured Query Language (SQL) is only available within the confines of a transactional database management system.

But it shoudn't have to be that way.

Consider the running example of "big data" used in *Hadoop: The Definitive Guide*. The National Climatic Data Center publishes hourly climatic data, such as temperature and pressure, from more than 10,000 recording stations all over the world. Data from 1901 onwards is available in text files. Each line of text contains the station code, the timestamp, and a number of climatic readings. The format is documented at **ftp://ftp.ncdc.noaa.gov/pub/data/noaa/ish-format-document.pdf**. The files are organized into subdirectories, one subdirectory for each year. Each subdirectory contains one file from each recording station that was in operation during that year. The individual files are compressed using gzip. All the files can be downloaded from **ftp://ftp.ncdc.noaa.gov/pub/data/noaa/**.

You might have already guessed where I am going with this.

Conceptually the above terabyte-sized data set is a single table. There are numerous questions that can be answered from this data set—and SQL would be a good tool for the job—but it should not be necessary to uncompress and load this huge mass of non-transactional data into a transactional database management system in order to get answers. The physical representation described above conserves storage space, and it is a technical detail that is irrelevant to the logical presentation of the data set as a single table; it is a technical detail that users don't care about. As Dr. Codd said in the opening sentence of his 1970 paper, "A Relational Model of Data for Large Shared Data Banks" (lovingly reproduced in the 100th issue of the *NoCOUG Journal*), "*future users of large data banks must be protected from having to know how the data is organized in the machine (the internal representation).*"

Why shouldn't we be able to query the above data set using good old SQL?

Well you can do just that with the Oracle query engine, and you don't have to load it into an Oracle database first. You can even take advantage of partitioning and parallelism. You can also write queries that mix and match data from the database and the filesystem.

The following demonstrations were performed using a pre-built developer VM for Oracle VM VirtualBox. The version of Oracle Database is 11.2.0.2. In the demonstrations, we only consider the years from 1901 to 1904. Here is the directory structure.

```
/home/oracle/app/oracle/admin/orcl/dpdump/noaa
/home/oracle/app/oracle/admin/orcl/dpdump/noaa/1901
/home/oracle/app/oracle/admin/orcl/dpdump/noaa/1904
/home/oracle/app/oracle/admin/orcl/dpdump/noaa/1902
/home/oracle/app/oracle/admin/orcl/dpdump/noaa/1903
```

We first need to create "directories" and define an "external table." The definition of this external table specifies a prepro-

> *"Future users of large data banks must be protected from having to know how the data is organized in the machine (the internal representation)."*

cessing script, which is the secret sauce that makes it possible for the query engine to traverse the subdirectories and uncompress the data.

```
create or replace directory data_pump_dir
  as '/home/oracle/app/oracle/admin/orcl/dpdump;

create or replace directory noaa_dir
  as '/home/oracle/app/oracle/admin/orcl/dpdump/noaa;

create or replace directory noaa_1901_dir
  as '/home/oracle/app/oracle/admin/orcl/dpdump/noaa_1901';
create or replace directory noaa_1902_dir
  as '/home/oracle/app/oracle/admin/orcl/dpdump/noaa_1902';
create or replace directory noaa_1903_dir
  as '/home/oracle/app/oracle/admin/orcl/dpdump/noaa_1903';
create or replace directory noaa_1904_dir
  as '/home/oracle/app/oracle/admin/orcl/dpdump/noaa_1904';

grant all on directory data_pump_dir to public;

grant all on directory noaa_dir to public;

grant all on directory noaa_1901_dir to public;
grant all on directory noaa_1902_dir to public;
grant all on directory noaa_1903_dir to public;
grant all on directory noaa_1904_dir to public;

create table temperatures
(
  station_code char(6),
  datetime char(12),
  temperature char(5)
)
organization external
(
  type oracle_loader
  default directory data_pump_dir
  access parameters
  (
    records delimited by newline
    preprocessor data_pump_dir:'uncompress.sh'
    fields
    (
      station_code position(1:6) char(4),
      datetime position(7:18) char(12),
      temperature position(19:23) char(5)
    )
  )
  location ('noaa')
);
```

Here's the tiny preprocessing script that makes it possible for Oracle to traverse the subdirectories and uncompress the data. It recursively traverses the filesystem beginning with the location specified by the query engine; that is, the location specified in the table definition. It uncompresses all zipped files it finds and sends the output to the "cut" utility, which cuts out only those column positions that we care about and writes what's left to standard output, not to the filesystem. The table definition specifies its location as data_pump_dir.

```
#!/bin/sh
/usr/bin/find $1 -name "*.gz" -exec /bin/zcat {} \; | /usr/bin/cut -c5-10,16-27,88-92
```

All the capabilities of SQL—including analytic functions and pivoting—can now be exploited as shown in the following example. For each month in the year 1901, we list the top three recording stations in terms of average monthly temperature.

```
select * from
(
  select
    month,
    station_code,
    dense_rank() over (partition by month order by average) as rank
  from
  (
    select
      substr(datetime,1,4)||'/'||substr(datetime,5,2) as month,
      station_code,
      avg(temperature) as average
    from temperatures
    where datetime >= '1901' and datetime < '1902'
    group by
      substr(datetime,1,4)||'/'||substr(datetime,5,2),
      station_code
  )
)
pivot(max(station_code) for rank in (1, 2, 3))
order by month;

MONTH  1        2        3
1901/01  227070   029600   029720
1901/02  227070   029070   029600
1901/03  227070   029070   029600
1901/04  029070   029500   029810
1901/05  029070   029500   029810
1901/06  029070   029810   029500
1901/07  029070   029500   227070
1901/08  227070   029070   029600
1901/09  029070   227070   029600
1901/10  227070   029600   029070
1901/11  227070   029600   029720
1901/12  227070   029600   029070
```

It's an epiphany—that's what it is.

We can also use "partition views" and take advantage of "partition pruning." For those who don't remember, partition views are a really old feature that predates "real" partitioning in Oracle 8.0 and above. Partition views continue to work just fine today, even in Oracle Database 11*g* Release 2.

Let's create a separate table definition for each year and then use a partition view to tie the tables together.

```
create or replace view temperatures_v as
select * from temperatures_1901
where datetime >= '190101010000' and datetime < '190201010000'
  union all
select * from temperatures_1902
```

> ***"The time is soon coming when the marriage of relational theory and transactional database management systems will be dissolved. We will be free to store structured non-transactional data outside a transactional database management system while continuing to exploit the entire universe of indexing, partitioning, and clustering techniques as well as the full power of relational languages, not only SQL."***

```
where datetime >= '190201010000' and datetime < '190301010000'
  union all
select * from temperatures_1903
where datetime >= '190301010000' and datetime < '190401010000'
  union all
select * from temperatures_1904
where datetime >= '190401010000' and datetime < '190501010000';

create table temperatures_1901
(
  station_code char(6),
  datetime char(12),
  temperature char(5)
)
organization external
(
  type oracle_loader
  default directory noaa_dir
  access parameters
  (
    records delimited by newline
    preprocessor data_pump_dir:'uncompress.sh'
    fields
    (
      station_code position(1:6) char(4),
      datetime position(7:18) char(12),
      temperature position(19:23) char(5)
    )
  )
  location ('1901')
);

-- the remaining table definitions are not shown for brevity
```

When we specify only a portion of the temperatures_v view, the query plan confirms that the unneeded branches of the view are filtered out by the query optimizer.

```
select count(*) from temperatures_v
where datetime >= '1901' and datetime < '1902';

Plan hash value: 907705830

SELECT STATEMENT
  SORT AGGREGATE
    VIEW
      UNION-ALL
        EXTERNAL TABLE ACCESS FULL TEMPERATURES_1901
        FILTER
          EXTERNAL TABLE ACCESS FULL TEMPERATURES_1902
        FILTER
          EXTERNAL TABLE ACCESS FULL TEMPERATURES_1903
        FILTER
          EXTERNAL TABLE ACCESS FULL TEMPERATURES_1904
```

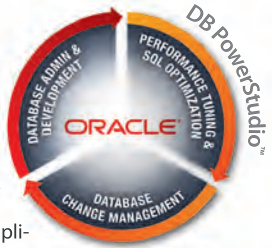
Finally, let's check whether query execution can be parallelized. And so it can.

```
select /*+ parallel(temperatures_v 4) */ count(*) from temperatures_v;

Plan hash value: 2698603534

SELECT STATEMENT
  SORT AGGREGATE
    PX COORDINATOR
      PX SEND QC (RANDOM)
        SORT AGGREGATE
          VIEW
            UNION-ALL
              PX BLOCK ITERATOR
                EXTERNAL TABLE ACCESS FULL TEMPERATURES_1901
              PX BLOCK ITERATOR
                EXTERNAL TABLE ACCESS FULL TEMPERATURES_1902
              PX BLOCK ITERATOR
                EXTERNAL TABLE ACCESS FULL TEMPERATURES_1903
              PX BLOCK ITERATOR
                EXTERNAL TABLE ACCESS FULL TEMPERATURES_1904
```

I predict that the time is soon coming when the marriage of relational theory and transactional database management systems will be dissolved. We will be free to store structured non-transactional data outside a transactional database management system while continuing to exploit the entire universe of indexing, partitioning, and clustering techniques as well as the full power of relational languages, not only SQL.

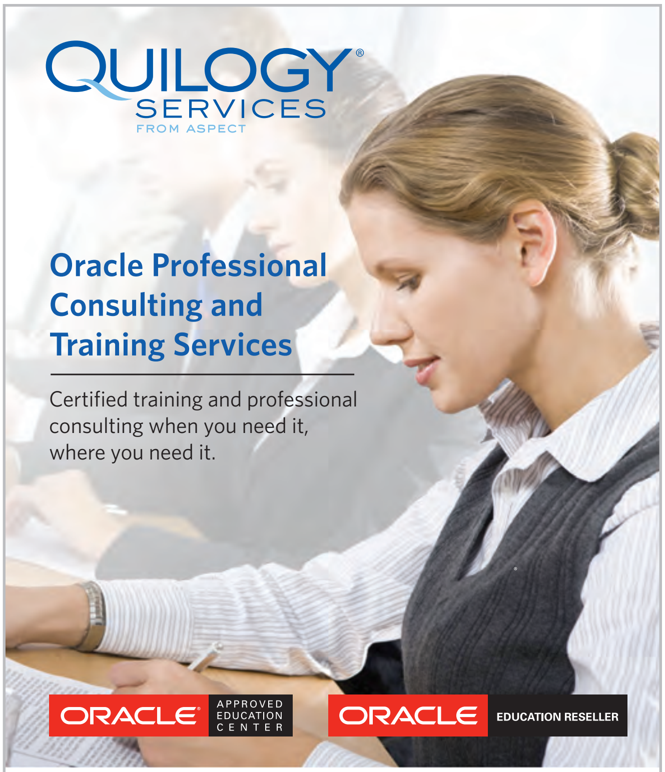
Over to you, Mogens. ▲

Speed Past OEM with
DB PowerStudio for Oracle


DB PowerStudio™
DATABASE ADMIN & DEVELOPMENT
PERFORMANCE TUNING & SQL OPTIMIZATION
ORACLE
DATABASE CHANGE MANAGEMENT

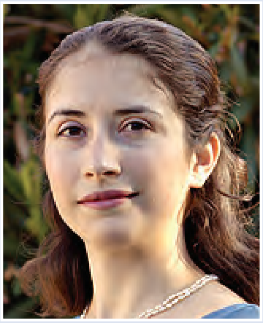
ignite8
Turn Up the Heat on Oracle Performance

DELPHIX



Integrating Oracle Database and Hadoop

by Gwen Shapira



Gwen Shapira

Modern data warehouses face challenges that are not easily resolved by traditional relational databases:

1. Storing and processing unstructured data—images, video, and PDFs.
2. Storing and processing large bodies of text such as email and contracts, especially when the requirement includes natural language processing—email and contracts.
3. Storing and processing large amounts of semi-structured data—XML, JSON and log files.

Traditional RDBMS data warehouses have limitations that make them less than effective for those new data types that are now required by the organization. They can be stored and processed in a relational database, but there is little benefit in doing so, and the cost can be high.

There are also some types of jobs in which a traditional data warehouse is not the optimal solution: ad-hoc analytical queries can be a bad fit for the existing schema and cause performance issues that will wreak havoc in the carefully scheduled batch workloads running in the data warehouse.

Advanced machine-learning algorithms are typically not implemented in SQL, which means that the developers will need to read massive amounts of data from the data warehouse and process it in the application.

Cleanup and transformation of data from different sources require large amounts of processing. While ETL jobs often run in the data warehouse, many organizations prefer to move these jobs to a different system that is more suitable to the task, rather than invest in larger data warehouse servers and the associated database license costs.

Hadoop and Map/Reduce

Hadoop is a platform designed to use inexpensive and unreliable hardware to build a massively parallel and highly scalable data-processing cluster.

It is designed to be a cost-effective and scalable way to store and process large amounts of unstructured and semi-structured data. This type of data processing poses some technical challenges: Clearly, large amounts of cheap storage are required to store large amounts of data, but large disks are not enough. High-throughput access to the data is required to allow timely processing of the data. In traditional storage systems, throughput did not keep up with the increases in storage space. On top of the storage system, a large number of processors are required to process the data, as well as applications that are capable of utilizing a large number of processors in parallel. And, of course, we also want the system to integrate cleanly with our existing infrastructure and have a nice selection of BI and data-analysis tools.

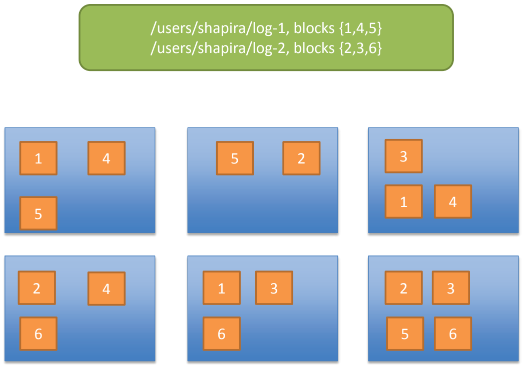
Hadoop was designed to address all of these challenges based on two simple principles:

1. Bring code to data.
2. Share nothing.

The first principle is familiar to all DBAs: we always plead with the developers to avoid retrieving large amounts of data for processing in their Java apps. When the amounts of data are large, this is slow and can clog the network. Instead, package all the processing in a single SQL statement or PL/SQL procedure, and perform all the processing where the data is. Hadoop developers submit jobs to the cluster that stores the data.

The second principle is due to the difficulty of concurrent processing. When data is shared between multiple jobs, there has to be locking and coordination in place. This makes development more difficult, imposes overheads, and reduces performance. Hadoop jobs are split into multiple tasks; each task processes a small part of the data and ignores the rest. No data is shared between tasks.

Hadoop is made of two components: HDFS, a distributed and replicated file system, and Map/Reduce, an API that simplifies distributed data processing. This minimalistic design—just a filesystem and job scheduler—allows Hadoop to complement the relational database. Instead of creating a schema and loading structured data into it, Hadoop lets you load any file onto HDFS and use Map/Reduce to process and structure the data later.

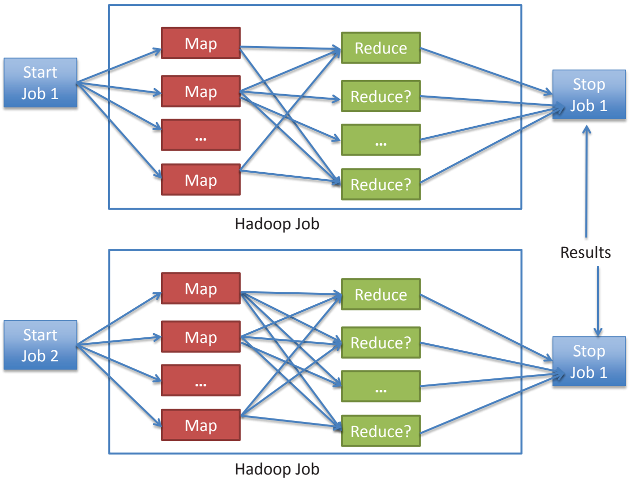

/users/shapira/log-1, blocks {1,4,5}
/users/shapira/log-2, blocks {2,3,6}
1
4
5
5
2
3
1
4
2
4
6
1
3
6
2
3
5
6

HDFS provides redundant and reliable storage for massive amounts of data, based on local disks of all nodes in the Hadoop cluster. File sizes are typically very large, and to reflect that, Hadoop's default block size is 64MB (compare this with Oracle's 8KB!). Sustained high throughput is given priority over low latency to maximize throughput of large scans. HDFS provides reliability by copying each block to at least three different servers in the cluster. The replication doesn't just provide failover in case a node crashes—it also allows multiple jobs to process the same data in parallel on different servers.

Map/Reduce is a method to distribute processing jobs across the servers. Jobs are split into small, mostly independent tasks. Each task is responsible for processing data in one block, and whenever possible it will run on a server that stores that block locally. The design maximizes parallelism by eliminating locks and latches.

As the name suggests, Map/Reduce has two phases: map and reduce. Map tasks filter and modify the data. This is analogous to the "where" portion of a query and to non-aggregating functions applied to the data. The reduce phase applies the data aggregation: group by and aggregating functions such as sum and average.

Since Map/Reduce jobs are limited to filtering and aggregating, more complex analytical queries do not translate well to Map/Reduce and are therefore difficult to implement in Hadoop. However, since Map/Reduce jobs are just Java code and can import existing Java libraries, Map/Reduce offers flexibility that does not exist in pure SQL. For example, OCR libraries can be used to allow Map/Reduce jobs to extract data from image files.



Don't imagine that Hadoop is the ultimate silver bullet. Hadoop has several well-known limitations: Presently there are not many enterprise products for Hadoop, and this means that building solutions on top of Hadoop is likely to require more resources than building the same solution on top of Oracle. Hadoop is best suited for batch processing, and low-latency real-time analysis is still in beta stages. In addition Hadoop can be tricky to debug, tune, and troubleshoot—it is not as clearly documented and instrumented as Oracle database.

One of the design goals of Hadoop is to allow loading data first and structuring it later. One of the main ways to provide structure for the semi-structured files stored in Hadoop is by using Hive, a data warehouse system on top of Hadoop that allows defining a schema for files in HDFS and then querying the data in the tables using a SQL-like language that is translated automatically into Map/Reduce jobs. This greatly simplifies the task of data processing in Hadoop and means that learning Java and writing Map/Reduce jobs from scratch is not mandatory.

ETL for semi-structured data

This is probably the most common use-case for Hadoop that I run into.

There is a system somewhere that is generating log files or XMLs, and these files include data that needs to get into the data warehouse. Of course, in order to extract the data and process it so it will fit into the star schema, some processing is required. It can be done on the relational database, but why waste the most expensive CPU cycles in the enterprise on text processing?

Instead, our customers use Flume to load log data into Hadoop and write Hadoop Map/Reduce jobs, or they use Hive to process the data and then use one of the techniques we'll discuss later to load the data into a relational database.

ETL for structured data

This looks like a strange use-case at first. The data is already structured—why bother with Hadoop? But anyone who's dealt with ETL jobs knows that one person's structured data is another's messy data source. OLTP data requires a lot of aggregation and transformation to fit into a data warehouse. Traditionally we did much of this work in the data warehouse itself, but data warehouses can be expensive and difficult to scale. As the amounts of data grow, so does the amount of time it takes run the ETL jobs.

Instead, load the data into Hadoop and add enough servers to make sure the job is split into small manageable chunks and finishes within the SLA.

Historical reports

Normally, searching for data within a relational database is easy enough. The only problem is when the data is not in the relational database. Since relational databases typically use expensive storage, and since performance typically degrades as the amounts of data grow, the amount of data stored in a relational database is typically limited.

Many online stores have been in business since the late 1990s. But how many of those businesses have detailed purchasing information for nearly 20 years of data? Marketing departments have learned not to ask for too-detailed information from the past, because it is unlikely the data was stored.

But with Hadoop using cheap storage, there is no reason not to store all the data your organization may need. And Hadoop clusters are not tape storage—they can process the data. So marketing departments can look at long-term trends, and IT departments no longer throw away potentially valuable data.

Needle in hay stack

Network equipment can generate huge amounts of logs. No one would dream of loading all the data into a relational database, but sometimes this data has to be searched.

For example, when it is determined that there was a security breach, many terabytes of log file data have to be scanned for the few lines indicating the source of the breach.

The solution is simple: just run grep. Hadoop was built to run grep fast and in parallel on huge amounts of data. It is easy to implement too.

Search unstructured data

I work for Pythian, a remote DBA company. Naturally, many of our customers feel a bit anxious when they let people they haven't even met into their most critical databases. One of the ways Pythian deals with this problem is by continuously recording the screen of the VM that our DBAs use to connect to customer environments. We give our customers access to those videos, and they can be replayed to check what the DBAs were doing. However, watching hours of DBAs just doing their job can be boring. We want to be able to find the interesting moments—catch the DBA just as "drop table" is being run or when a difficult issue is solved.

To provide this capability, Pythian streams the recordings to our Hadoop cluster. There we run OCR in parallel on each screenshot captured in the recording, and isolate the text on the screen and the DBA keystrokes. Other jobs mine the data for suspicious key words ("drop table"), patterns (credit card numbers), and root access. This system lets our customers verify that their databases are safe without excessive work on their or our part.

Connecting the Big Dots

Hopefully, by now you see how Hadoop can complement a traditional data warehouse and how you can use Hadoop to process more data types or just more data.

But clearly for all this wonderful stuff to work in practice, we need a way to get data from a relational database into Hadoop and, more important, from Hadoop into a relational database.

Let's look into techniques for doing just that.

Sqoop

Sqoop is an Apache project designed to transfer data between Hadoop and relational databases.

Its main modules are sqoop-import, which transfers data from a relational database into Hadoop, and sqoop-export, which transfers data from Hadoop to relational databases.

Both modules work in a similar manner: Sqoop starts multiple Map/Reduce jobs, which connect to the database and run queries in parallel. Sqoop-import runs select statements and writes the results on HDFS; sqoop-export reads data from HDFS and runs DML statements to insert the data into database tables.

The main argument to sqoop-import is --table, which allows the developer to specify which database table to import into Hadoop. Sqoop-import also allows specifying exactly which columns to import, along with a where clause to limit the rows imported.

For example:

```
sqoop import --connect jdbc:oracle:thin:@//dbserver:1521/masterdb
--username hr --table emp
--where "start_date > '01-01-2012' "
```

will import into Hadoop data about employees who started work at the beginning of last year.

By default, sqoop-import runs four map processes to read the data from the database, but the number of map processes can be modified in a parameter. To split the data between the map processes, sqoop-import looks for the primary key, checks the minimum and maximum values, and divides that range between the map processes.

Suppose our emp table has a primary key of (id), the range of ids in our table is 1 to 12, and we are running two map processes.

In this case one map process will run the query:

```
"select * from emp where start_date > '01-01-2012' and id >= 1 and id < 6"
```

and the other map process will run:

```
"select * from emp where start_date > '01-01-2012' and id >= 6 and id < 13"
```

You can specify a column other than primary key to split the work between map processes:

```
sqoop import jdbc:oracle:thin:@//dbserver:1521/masterdb
--username myuser
--table shops --split-by customer_id
--num-mappers 16
```

In this example, we decided to run 16 map processes and split the work by customer_id, even though the primary key is shop_id.

It is critical to remember that the split-by column should be either indexed or a partition key, otherwise our example will result in 16 parallel full table scans on the shops table, which is likely to cause severe performance issues.

Sqoop-import can be even more flexible and allow you to import the results of any query:

```
sqoop import  --query 'SELECT a.*, b.* FROM a JOIN b on (a.id == b.id)
WHERE a.type="PRIME" and  $CONDITIONS'
--split-by a.id --target-dir /user/foo/joinresults
```

In this case $CONDITIONS is a placeholder for the where-clause that sqoop-import adds for splitting work between the map processes. Since the query is freeform, it is our responsibility to figure out the split column and the right place to add the condition in the query.

Sqoop-export is not quite as flexible as sqoop-import, and it is important to be aware of its limitations.

Sqoop-export works by turning each input line into an insert statement and commits every 1000 statements. This means that failure of a sqoop-export job will leave the database in an unknown state with partial data inserted. This can cause retries of the export job to fail due to collisions or to leave duplicate data, depending on the constraints on the table.

By default, sqoop-export fails if any insert statement fails. This can become a problem if there are unique indexes and existing data in the target table. Sqoop-export can be configured to retry unique-constrain violations as updates, effectively merging the data into the target table.

In general sqoop-export is not as flexible and configurable as sqoop-import. Developers can specify source files on HDFS, target tables, number of map processes that perform the inserts, and the update behavior:

```
sqoop export
--connect jdbc:oracle:thin:@//dbserver:1521/masterdb
--table bar
--export-dir /results/bar_data
```

Due to those limitations, sqoop-export is rarely used for databases other than MySQL (sqoop-export has a special "direct mode" for MySQL that overcomes many of these issues).

Fuse-DFS

Oracle highly recommends using external tables to load data into a data warehouse (**http://www.oracle.com/technetwork/database/bi-datawarehousing/twp-dw-best-practices-for-implem-192694.pdf**), with good reason:

- External tables allow transparent parallelization of the data access.
- External tables allow you to avoid staging tables. Instead you use SQL to read data from the source files, transform it, and place it in target tables, all in one command.
- External tables allow parallel direct path writes, further improving performance.

So when I had to load large amounts of data from Hadoop into an Oracle Data Warehouse, naturally I looked at ways that I could use external tables. With parallel reads and writes, direct path writes, and one-step ETL, it is guaranteed to beat sqoop-export performance without having to worry about partial data load.

If Oracle could access files on HDFS, this would be no problem at all. As you'll see soon, it can do just that using Oracle's Hadoop Connectors. But those connectors have significant license fees, which some of my customers were unwilling to invest in. By default, Oracle external tables can only access files on the server's filesystem. Which means we need to mount HDFS as a POSIX-like filesystem. This is done with fuse-DFS, which is relatively easy to install and configure:

```
sudo yum install hadoop-0.20-fuse
hadoop-fuse-dfs dfs://<namenode_hostname>:<namenode_port>
<mount_point>
```

Now that every user can see the HDFS files, it is easy enough to create an external table to read them. The catch is that files on Hadoop are not always in plain text, and you may need to add a preprocessor to read them. You can read a detailed description of how to do that on the Cloudera blog: **http://blog.cloudera.com/blog/2012/09/exploring-compression-for-hadoop-one-dbas-story/**

The important benefit of using fuse-DFS and external tables is that it allows you to use standard Oracle tools and leverage all your hard-won experience as an Oracle tuning expert to squeeze every last bit of performance out of the data load process. Sqoop-export does not give you the flexibility to do this.

Oracle Loader for Hadoop

Oracle Loader for Hadoop is a high-performance utility used to load data from Hadoop into Oracle database.

Oracle Loader for Hadoop runs as a Map/Reduce job on the Hadoop cluster, shifting the processing work to Hadoop and reducing load on the Oracle database server.

The Map/Reduce job partitions the data, sorts it, and converts it into Oracle database file formats before loading the data into Oracle database using a direct write path. All this pre-processing is done on the Hadoop cluster, close to the data origins and where processing power is easier to scale. Oracle database only has to place prepared data blocks into data files.

Loading pre-sorted data into Oracle tables means that index creation will be faster and require less I/O and CPU. Compression, especially HCC, will also be faster, take less CPU, and result in higher compression ratios than when compressing unsorted data.

In version 2 of the connector, Oracle added support for Avro file type, for Hadoop compression, and for loading data from Hive tables.

If part of your ETL process includes frequent data loads from Hadoop to Oracle, the performance benefits of using Oracle Loader for Hadoop are difficult to ignore. The main drawback is that it is not open-source and requires a license to run.

Oracle SQL Connector for Hadoop

In a previous version, this connector was called Oracle Direct Connector for HDFS and provided a pre-processor for creating an external table from files residing in HDFS. This connector was benchmarked by Oracle and shown to be about five times faster than using fuse-DFS for the external tables.

In version 2, the connector was rewritten with a new interface, and it is now more powerful and easier to use. It runs as a Map/Reduce job and automatically creates the external table using either data in the Hive data dictionary or by assuming that all columns in a delimited text file are varchar2 type. Just like Oracle Loader for Hadoop, the SQL Connector also supports Avro file types and compressed data.

Once the external table exists, it can be used for ETL the same way any external table can, and the connector transparently handles parallelism.

Closing notes and tips for the aspiring Hadooper

It is important to remember that while Hadoop offers exciting new possibilities, Oracle database is a powerful and well-understood platform. I always advise customers to first make sure they are using Oracle correctly before venturing out to new platforms. Are you using external tables for your ETL? Efficient direct path writes? Is your data partitioned correctly? Are you using partition-wise joins and star transformations? Moving an ETL process to Hadoop is far more challenging than making sure the existing process is optimal, so start with Data Warehouse 101 tuning.

At the time of writing, Hadoop is still best optimized for batch processing and jobs. Real-time ability to query Hadoop is still in beta, and even simple Map/Reduce jobs take a few seconds to process. If real-time analytics is part of the requirements, I'd wait before adopting Hadoop.

As always before embarking on a new data warehouse project, make sure you have clear requirements, goals, and deliverables. Make sure Hadoop's capabilities and limitations make sense in the context of those requirements and goals. It is easy to get excited about adopting new technology while losing the overall picture.

(continued on page 26)

NoCOUG Winter Conference

Session Descriptions

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

–Keynote–

Oracle NoSQL Database and Oracle Database: A Perfect Fit

Dave Rubin, Oracle Corporation 9:30–10:30

Oracle NoSQL Database and Oracle Database are complementary technologies that work together in order to solve enterprise-class, high-velocity big data problems. Although NoSQL databases are ideal for certain kinds of workloads, leveraging operational enterprise data in conjunction with a NoSQL database can deliver compelling solutions to the organization. This presentation focuses on the integration between Oracle NoSQL Database, a highly scalable and available transactional key-value store, and related Oracle technologies such as Oracle Database, with some example use cases that illustrate the value of the combined solution.

Dave Rubin has been involved with big data from the perspective of a user as well as a developer of big data technologies. In his current role at Oracle, he leads the development of Oracle's NoSQL database. Previously, Dave led the infrastructure engineering team at Cox Digital Solutions, developing big data solutions in the area of online display advertising. He holds four U.S. patents in the areas of query optimization and advanced transaction models.

–Auditorium–

Big Data: The Big Story

Jean-Pierre Dijcks, Oracle Corporation 11:00–12:00

Weblogs, social media, smart meters, sensors, and other devices generate high volumes of fast-moving data: big data. This session explains how to harness big data, your existing data, and predictive analytics to make better decisions in an environment of rapid shifts in behavior and instant feedback. It outlines a technology landscape for maximizing the impact of your big data implementation on your business. You will learn about the technologies that constitute a big data architecture, how to leverage and implement advanced analytics for real-time decisions, and the tools needed to know the unknown.

Building the Integrated Data Warehouse with Oracle Database and Hadoop

Gwen Shapira, Pythian 1:00–2:00

From tracking customers in online stores to tweets and blog posts, unstructured data is rapidly growing, and businesses are looking for ways to analyze it. In this presentation, I will explain why storing and processing unstructured data is a challenge best answered by specialized systems such as Hadoop. I will dive into how Hadoop works and why it is such a scalable solution for managing unstructured data, and I will show how to integrate Hadoop with existing DWH systems on Oracle to allow using the data in existing BI tools and reports.

Data Management in an Oracle NoSQL Database Application

Anuj Sahni, Oracle Corporation 2:30–3:30

Curious about how to write an Oracle NoSQL Database application? Wonder what data management functions look like in a NoSQL environment? Attend this lab session, and find out. The Oracle OpenWorld 2012 big data demo contains several components that rely on Oracle NoSQL Database applications. Learn how those apps were designed, how the functionality was implemented, and how the NoSQL data was accessed and managed. Go through practical, hands-on exercises with real live data to read and write data to Oracle NoSQL Database. Examine and discuss portions of the Oracle NoSQL Database application code. Understand the Oracle NoSQL DB Java API from a practical, Java application developer–centric point of view. Ask questions, and get tips and tricks from the man who wrote the application code.

–Room 102–

A Technical Look at Oracle Fusion Applications

Editor's Pick

Ernesto Lee, Aspect 11:00–12:00

In this presentation, you will first learn exactly what Oracle Fusion Applications is from a technical perspective, and you will also learn key strategies and lessons learned from our experiences. From a technical perspective, some things worked well and others did not. From a practical perspective, you will gain insight into our approach to standing up the Fusion Apps environment and overcoming technical hurdles. Hear the real-life war stories surrounding what it takes to stand up this truly incredible product.

Databases Virtualization: Instant Zero Space Cloning

Kyle Hailey, Delphix 1:00–2:00

Overview of current technologies for virtualizing databases. Database virtualization technology includes copy-on-write filesystems, journal file systems, point-in-time snapshots, point-in-time writeable clones, and the NFS technology stack. This technology stack will be explained and the presentation will go into differences in specific technologies as implemented by Oracle, Delphix, EMC, and NetApp.

Reduce Database Latency

Matt Goldensohn, WHIPTAIL Storage 2:30–3:30

At Whiptail, we have developed an all-Flash-based storage array to help database applications experience significantly lower response times. We do this by removing much of the application latency that comes from the data storage layer of the infrastructure that the DBs run on. These are just some of

Many Thanks to Our Sponsors

NoCOUG would like to acknowledge and thank our generous sponsors for their contributions. Without this sponsorship, it would not be possible to present regular events while offering low-cost memberships. If your company is able to offer sponsorship at any level, please contact NoCOUG's president, Naren Nagtode. ▲

Long-term event sponsorship:

CHEVRON

ORACLE CORP.

Thank you! Gold Vendors:

- Confio Software
- Database Specialists
- Delphix
- Embarcadero Technologies
- GridIron Systems
- Quilogy Services
- WHIPTAIL Storage

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

TREASURER'S REPORT

Dharmendra Rai, *Treasurer*

Beginning Balance October 1, 2012		**$ 60,011.74**
Revenue		
Membership Dues	4,728.00	
Meeting Fees	450.00	
Vendor Receipts	6,985.20	
Training Day Fees	3,900.00	
Interest	1.29	
Total Revenue		**$ 16,065.29**
Expenses		
Regional Meeting	9,792.48	
Journal	3,573.90	
Membership	146.27	
Administration	23.84	
Board Meeting	376.79	
Training Day expenses	3,841.14	
Membership and conference s/w	1,127.00	
Insurance	546.12	
Vendor expenses	175.67	
Total Expenses		**$ 19,603.21**
Ending Balance December 31, 2012		**$ 56,473.82**

the benefits that our other referenceable customers have seen in their DB environments:

- Reduced cost and reduced complexity as compared to larger-scale solutions. Most of our solutions install in hours, not days.
- No (or minimal) change to current storage infrastructure.
- Uses traditional protocols (Fibre Channel/iSCSI/NFS), so there is no need to re-write your current application (instead, put in faster storage).
- Lower response times (microseconds versus traditional milliseconds).
- Multiple and concurrent workloads up to 650,000 IOPS.
- Reclaim traditional storage capacity that has been over-provisioned in order to achieve performance.
- Much lower power/cooling/ floor space requirements.

Exadata Success Story at PayPal

Amit Das, PayPal 4:00–5:00

This presentation will share our story of going live with Exadata in 60 days. It will cover the difficulties in implementing the world's largest OLTP configuration. It will explain how Exadata is helping to run one of our OLTP systems (most people think that Exadata is great for data warehousing only). This presentation will share how we are leveraging smart cache and smart redo to run our OLTP system with 10X performance improvement. Also, it will share the architectural detail of our Exadata configuration, which is known as one of the world's largest OLTP environments on Exadata.

–Room 103–

Understanding SQLTXPLAIN (SQLT) Main Report by Navigating Through Some Samples

Carlos Sierra, Oracle Corporation 11:00–12:00

SQL tuning is a daunting task. Too many things affect the cost-based optimizer (CBO) when deciding on an execution plan. CBO statistics, parameters, bind variables and their peeked values, histograms, and a few more are common contributors. The list of areas to analyze keeps growing. Over the past few years, Oracle has been using SQLTXPLAIN (SQLT) as a systematic way to collect all the information pertinent to a poorly performing SQL statement and its environment. With a consistent view of this environment, an expert on SQL tuning can perform a more diligent task, focusing more on the analysis and less on the information gathering. This tool could also be used by an experienced DBA to make life easier, at least when it comes to SQL tuning. This session uses some SQLT sample files to familiarize participants with navigation through the set of files and within the main SQLT report. Special attention is given to SQLT XTRACT and SQLT XECUTE files. All sections of the main SQLT report are explained.

The Sins of SQL Programming That Send the DB to Post-Upgrade Performance Purgatory

Abel Macias, Oracle Corporation 1:00–2:00

A "sin" in this case is a bad practice that causes great hardship to the business during database upgrades. This paper shows examples of real-world cases submitted to support "sinful" queries giving wrong results or unexplainable data error messages, usually after an upgrade. These practices can have a negative effect on the performance of the queries after the migration, too, so learn to recognize them and avoid purgatory. The usual argument from customers is that "it used to work."

Advanced SQL Injection Techniques

Slavik Markovich, CTO, McAfee. 2:30–3:30

SQL injection remains one of the most widely used techniques leveraged by hackers to get at sensitive information or to gain further system access to wreak havoc with more attacks. Though the majority of SQL injections are fairly simple and familiar, the more advanced ones can evade detection by some conventional security measures. In this presentation, I will do an in-depth analysis of some sophisticated SQL injection hacks and attacks, and offer up some best practices on how to harden your applications and databases against them in order to keep business-critical information safe. Live demos of various SQL injection types will be performed. Code for the sample application and attacks will be available for download.

Looney Tuner? No, there IS a method to my madness!

Janis Griffin, Confio Software 4:00–5:00

Query tuning is often more art than science and it can quickly eat up a lot of DBA and/or Developer time. This presentation will outline a method for determining the best approach for tuning queries by utilizing response time analysis and SQL Diagramming techniques. Regardless of the complexity of the statement, this quick, systematic approach will lead you down the correct tuning path with no guessing. If you are a beginner or expert, this approach will save you countless hours tuning a query. ▲

(continued from page 23)

Have realistic expectations. Hadoop is a relatively new technology. It is not as mature as Oracle and can be much more challenging to deploy, tune, and troubleshoot. Projects will take longer than you are used to, and unexpected snags and bugs will show up.

When things go wrong, don't forget the basics. Hadoop is different from relational databases in many aspects, but many skills and tasks apply: If things are slow, find the bottleneck. Use operating system tools to find out what is slow—storage, network, CPUs? Expect performance problems as data and workloads grow. Plan to grow the cluster accordingly.

Make sure you are using the right tool for the job—structured data, real-time reports, BI integration, frequent updates, and OLTP-like workloads belong in Oracle data warehouse. Unstructured and semi-structured data, large bodies of text, and data whose structure can change frequently without notice belong in Hadoop.

Because a Hadoop cluster can be created from any combination of servers, there is no excuse not to have a "toy cluster" to try new ideas. Perhaps your first Hadoop cluster is a server from QA that no one uses and two laptops—it's enough to get started and explore whether Hadoop can add any value to your data warehouse, and no one will stop the project due to high costs.

It's a new world out there. Have fun exploring. ▲

Database Specialists: DBA Pro Service

DBA PRO BENEFITS

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

CUSTOMIZABLE SERVICE PLANS FOR ORACLE SYSTEMS

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

DBA Pro's mix and match service components

Access to experienced senior Oracle expertise when you need it

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

24 x 7 availability with guaranteed response time

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

Daily review and recommendations for database care

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

Monthly review and report

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

Proactive maintenance

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

Onsite and offsite flexibility

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

DatabaseSpecialists

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

NoCOUG
P.O. Box 3282
Danville, CA 94526

NoCOUG Winter Conference Schedule

Thursday, February 21, 2013—Oracle Conference Center, Redwood Shores, CA

Please visit **http://www.nocoug.org** for updates and directions, and to submit your RSVP.
Cost: $50 admission fee for non-members. Members free. Includes lunch voucher.

8:00 a.m.–9:00	Registration and Continental Breakfast—Refreshments served
9:00–9:30	**Welcome:** Naren Nagtode, NoCOUG president
9:30–10:30	**Keynote:** *Oracle NoSQL Database and Oracle Database: A Perfect Fit*—Dave Rubin, Oracle Corporation
10:30–11:00	**Break**
11:00–12:00	**Parallel Sessions #1**
	Auditorium: *Big Data: The Big Story*—Jean-Pierre Dijcks, Oracle Corporation
	Room 102: *A Technical Look at Oracle Fusion Applications*—Ernesto Lee, Aspect **Editor's Pick**
	Room 103: *Understanding SQLTXPLAIN (SQLT) Main Report by Navigating Through Some Samples* —Carlos Sierra, Oracle Corporation
12:00–1:00 p.m.	**Lunch**
1:00–2:00	**Parallel Sessions #2**
	Auditorium: *Building the Integrated Data Warehouse with Oracle Database and Hadoop* —Gwen Shapira, Pythian
	Room 102: *Database Virtualization: Instant Zero Space Cloning*—Kyle Hailey, Delphix
	Room 103: *The Sins of SQL Programming That Send the DB to Post-Upgrade Performance Purgatory* —Abel Macias, Oracle Corporation
2:00–2:30	**Raffle**
2:30–3:30	**Parallel Sessions #3**
	Auditorium: *Data Management in an Oracle NoSQL Database Application* —Anuj Sahni, Oracle Corporation
	Room 102: *Reduce Database Latency*—Matt Goldensohn, WHIPTAIL Storage
	Room 103: *Advanced SQL Injection Techniques*—Slavik Markovich, CTO, McAfee
3:30–4:00	**Break and Refreshments**
4:00–5:00	**Parallel Sessions #4**
	Auditorium: *TBD*
	Room 102: *Exadata Success Story at PayPal*—Amit Das, PayPal
	Room 103: *Looney Tuner? No, there IS a method to my madness!*—Janis Griffin, Confio Software
5:00–	**NoCOUG Networking and No-Host Happy Hour**

RSVP *required* at http://www.nocoug.org

The *NoCOUG Journal* design and production: Giraffex, Inc., S.F.
Front cover photo: Composite of Airplane With Computer Monitor And Clock by Comstock.