Monish Sharma
Predictive Change Management

Mike Messina
AWR Central Repository

Tim Gorman
Fastest Isn’t Always Shortest

Summer QEW
Wonderful Time At Elitch Gardens

Member Focus - Dan Hotka
Colorado Fun
The School of Computer & Information Sciences in Regis University College for Professional Studies uses state-of-the-art learning environments, experienced faculty and strong industry relationships to teach students the latest technologies and software.

As an accredited university ranked by U.S.News & World Report, busy adults choose Regis University for the academic rigor, and our College for Professional Studies for the flexibility.

> Accelerated 5- and 8-week classes
> Online and campus-based learning
> Multiple start dates throughout the year

Todd uses his degree to lead by example. He shows young kids how to build influence, even with LEGO® robotics. How will you be influential?

The School of Computer & Information Sciences in Regis University College for Professional Studies uses state-of-the-art learning environments, experienced faculty and strong industry relationships to teach students the latest technologies and software.

As an accredited university ranked by U.S.News & World Report, busy adults choose Regis University for the academic rigor, and our College for Professional Studies for the flexibility.

> Accelerated 5- and 8-week classes
> Online and campus-based learning
> Multiple start dates throughout the year

Todd uses his degree to lead by example. He shows young kids how to build influence, even with LEGO® robotics. How will you be influential?

The School of Computer & Information Sciences in Regis University College for Professional Studies uses state-of-the-art learning environments, experienced faculty and strong industry relationships to teach students the latest technologies and software.

As an accredited university ranked by U.S.News & World Report, busy adults choose Regis University for the academic rigor, and our College for Professional Studies for the flexibility.

> Accelerated 5- and 8-week classes
> Online and campus-based learning
> Multiple start dates throughout the year

Todd uses his degree to lead by example. He shows young kids how to build influence, even with LEGO® robotics. How will you be influential?

The School of Computer & Information Sciences in Regis University College for Professional Studies uses state-of-the-art learning environments, experienced faculty and strong industry relationships to teach students the latest technologies and software.

As an accredited university ranked by U.S.News & World Report, busy adults choose Regis University for the academic rigor, and our College for Professional Studies for the flexibility.

> Accelerated 5- and 8-week classes
> Online and campus-based learning
> Multiple start dates throughout the year

Todd uses his degree to lead by example. He shows young kids how to build influence, even with LEGO® robotics. How will you be influential?

The School of Computer & Information Sciences in Regis University College for Professional Studies uses state-of-the-art learning environments, experienced faculty and strong industry relationships to teach students the latest technologies and software.

As an accredited university ranked by U.S.News & World Report, busy adults choose Regis University for the academic rigor, and our College for Professional Studies for the flexibility.

> Accelerated 5- and 8-week classes
> Online and campus-based learning
> Multiple start dates throughout the year

Todd uses his degree to lead by example. He shows young kids how to build influence, even with LEGO® robotics. How will you be influential?

The School of Computer & Information Sciences in Regis University College for Professional Studies uses state-of-the-art learning environments, experienced faculty and strong industry relationships to teach students the latest technologies and software.

As an accredited university ranked by U.S.News & World Report, busy adults choose Regis University for the academic rigor, and our College for Professional Studies for the flexibility.

> Accelerated 5- and 8-week classes
> Online and campus-based learning
> Multiple start dates throughout the year
On the Cover:

Glenn Goodrum, an RMOUG member since 2007, is the principal data architect at SquareTwo Financial, responsible for designing their new data warehouse and related business intelligence systems. He took this photograph of Trout Lake on Colorado Highway 145 south of Telluride in the San Juan Mountains on a crisp autumn afternoon in 2010.
For those of you and your families who joined us at our wonderful quarterly educational workshop last month at Elitch Gardens Six Flags, please join me in thanking Programs Director and Vice President Carolyn Fryc and our sponsors Arisant and Oracle. We enjoyed two hard-hitting and information-packed presentations in the morning from Dan Hotka on APEX and from Graham Woods and John Beresniewicz on Active Session History (ASH). Then, we had lunch courtesy of Arisant with wine and beer, a raffle, and then we had passes to spend the rest of the afternoon and evening in the park with our families. It was a great way to thank our families for their patience and support, and we are definitely planning to do the same thing at the August QEW next year. Please come join us! And thanks again to Arisant and Oracle, whose sponsorship of the meeting made everything possible.

With the rehosting of our website from static HTML to Wordpres, RMOUG now has a way to keep information fresh using blog aggregators, links to social media sites, and updates about upcoming events. And that’s not all...

Even though RMOUG has its roots firmly in Colorado, we have been drawing speakers and writers from all over the world for many years, filling the content of our award-winning newsletter, the quarterly educational workshops, the Regis DBLabs, and our awesome Training Days conference.

In a several informal discussions on the global ORACLE-L email list (http://www.freelists.org/list/oracle-l), the annual RMOUG Training Days conference has for many years been listed among the top half-dozen Oracle conferences in the world, alongside: Oracle Open World, Collaborate, UKOUG, KScope, and Hotsos.

Recently, the RMOUG board of directors has been testing the usability of various web-conferencing packages. Web-conferencing will reduce our administrative costs and increase efficiency by enabling internal RMOUG meetings to be held regardless of where our busy volunteers work, reside, or have fled. Less time on traveling, more time doing.

But more importantly, the addition of web-conferencing to RMOUG opens another channel for knowledge sharing and transfer within our membership, and indeed the rest of the world. In addition to the other events that RMOUG produces, we’re going to be creating an “RMOUG Channel” online for a series of frequent, brief 15-, 30-, or 45-minute presentations on relevant and useful topics in Oracle technology. These presentations will be recorded and archived on the website, along with accompanying materials such as PowerPoint presentations, so that members and the general public can search them, view them, and learn from them as desired. If you can’t attend a presentation on the “RMOUG Channel” live, then you can download the recording of the presentation from the RMOUG website and view it at your convenience.

This also opens up the possibility of live feeds via web-conference from the Training Days conference in the future. While we’ve not yet decided whether or not to try this during the upcoming Training Days 2013 conference, we’re certainly looking hard at doing so at future conferences.

Additionally, back issues of the RMOUG newsletter are in the process of being posted in PDF format to the RMOUG website by our newsletter editor, Pat Van Buskirk, working with our webmaster, Art Marshall. So, now you can search for the good stuff in a newsletter article you saw a while ago, or read it from work in case you left your copy of the current newsletter at home.

All of these new developments mean more information for you to help you do your job better.

And, at the same time, they represent more opportunities for you to advance your career in information technologies by presenting on a topic in which you are expert, providing more methods to get your message out to RMOUG members and the rest of the world. If you’ve got a tip, trick, or technique that you’re willing to share, then I’ve always said that you couldn’t do a presentation for a nicer bunch of folks than the RMOUG membership. That holds true whether you’re presenting in-person at a quarterly educational workshop (QEW), at the Training Days conference, in a newsletter article, or in a web-conference on the upcoming “RMOUG Channel”.

Please feel free to email me at “president@rmoug.org” if you have any ideas or suggestions, and if you’d like to find out how to volunteer with RMOUG, please visit our volunteer webpage at “http://www.rmoug.org/volunteer/” or email me.
Stan Yellott Scholarship Fund

Eligibility Requirements

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

Award notification will be given within 45 days following application deadlines. Upon acceptance of the scholarship, additional information may be required.

For Details, Visit the RMOUG Website
www.rmoug.org

RMOUG Scholarship Mission

To provide educational opportunities to members of the organization about the information technology industry in general, and in particular the technology of Oracle Corporation to include databases, storage, networking and application development, specifically the products and services of the Oracle Corporation.

To collect, post and distribute information about Oracle technologies and other related technologies to members.

To provide members with the ability to help their peers maximize their knowledge and skills working with products in information technology and in particular Oracle products.

To provide a consolidated channel of communication, conveying needs, concerns, and suggestions, for members of the organization, to Oracle Corporation and other vendor corporations involved with Oracle related technology.

To encourage members to present their information technology experiences using Oracle and other products and services.

To provide a consolidated channel of communication between members of the RMOUG and other communities in related information technology industries.

To promote educational opportunities for students of information technology through directed funding and services for educational purposes.

RMOUG is committed to supporting others in the pursuit of technical knowledge.

The Scholarship Fund started in 2001 to encourage future IT professional in their efforts to broaden their knowledge. In 2007, RMOUG voted to rename the scholarship fund to honor the memory of Stan Yellott. Stan was a long time member of RMOUG where he supported the user community by serving on the RMOUG board. Stan focused on expanding Oracle educational opportunities. Stan’s vision was to include high school and college students as the next generation of IT professionals.

Quarterly Education Workshops
- Special Annual Training Days Rates
- Database Labs
- List Server Subscriptions
- SQL>Update Magazine

www.rmoug.org/member.htm
Predictive Change Management
A Proactive and an Adaptive Approach
for Database Performance Management

by Monish Sharma

Background

Consider the below scenarios:

As part of the quarterly patching policy, a system administrator patches the operating system on the database server on a planned maintenance window on a Sunday morning. On Monday morning, users experience severe performance degradation with some of their analytical queries. The DBA sees huge waits on logical reads on the database. Based on the recommendations of their tuning tools and automatic advisors, the DBA tweaks the database configuration parameters and even reboots the database. After a few painful days of trying several options, the problem was narrowed down to a buggy OS patch that was applied to the server. A workaround provided by the OS vendor fixed the issue.

The storage team upgraded the microcode of the SAN storage used by a database server on a planned maintenance window on a Saturday afternoon. On Sunday evening, some scheduled application reports ran three times slower than usual. Based on the data gathered by the performance monitoring tools and through tracing, the DBA found that the physical reads on the database were very slow. DBAs started adding indexes on some of the tables used by the report queries. This somewhat helped, but created new performance issues with some other queries. By Tuesday, the problem was traced back to the SAN microcode upgrade. The microcode was downgraded to fix the issue.

Had the DBAs known about the potential impact and extent of these environmental changes before they were implemented, they would have made better decisions either by pushing for a full load test on the QA server with the new OS patch or would have prepared themselves for switching to a standby database that used a different SAN storage. Incorrect diagnosis and troubleshooting is expensive and error-prone. Also, these events end up repeating themselves across time and systems.

In today’s rapidly evolving digital age, organizations undergo a tremendous amount of changes. According to a Forrester research report, organizations make up to 500 changes per month to their IT infrastructure (Forrester, 2007). These changes are fueled by factors like mergers, acquisitions, explosive data growth, consolidation, virtualization, competitive landscape and long-term investments (McKendrick, 2011). As a result, the Information Technology (IT) environment within the organizations are becoming highly integrated, dynamic and more and more complex (Böhm et al., 2010; Corp, 2005).

A typical database environment, as shown in Figure 1 below, has several layers (Schallehn, 2010; Shasha, 1992). Given the heavy use of interdependent technologies and also the high density of the IT environment, an issue at any of these layers of the database environment stack has a high potential of causing impact to other layers within the stack. Since organizations have large numbers of database systems within their complex, virtualized and highly-integrated environments, the impact and the extent of environmental changes to the performance across databases becomes significant and widespread.

Figure 1: Typical Organizational Database Environment Stack

As seen from the above two cases, operating database systems at high performance levels under complex, dynamic and a dense environment requires the database administrators (DBAs) to frequently conduct performance tuning or optimizations (Rabinovitch, 2009; Schallehn, 2010; Telford et al., 2003). This manual performance tuning task is repetitive, expensive, time-consuming and error-prone (Gil et al., 2002; Oliveira et al., 2006; Wiese & Rabinovitch, 2009). Also, since it’s a context-dependent task, the following organization-specific factors typically influence the tuning task and its outcome:

Aggressive tuning deadlines: Speedy and timely requirement of information from the database systems dictates the aggressive deadlines for the tuning tasks. As a result, this task becomes cognitively taxing for the DBAs.

Performance tuning costs: These are the opportunity costs that an organization incurs as a result of the performance problem. This is also responsible for aggressive deadlines.
Environmental change impact uncertainty: This is the result of complexity and density of the database environment stack as shown in figure 1. This uncertainty makes the troubleshooting of the performance problem a challenging task.

Manually tuning a databases under such factors could result in an incorrect diagnosis and error-prone tuning. Hence, acquiring the knowledge of the impact and extent of environmental changes proactively, i.e., before the changes are even implemented, could minimize the impact of some of these adverse effects.

Although, there have been several developments in the area of autonomous as well as semi-autonomous performance tuning research, these approaches, as evident from the aforementioned scenarios, are limited in their use and effectiveness. This is because they adopt a narrow focus towards the organizational database environment stack by focusing primarily on the database layer within the environment stack. These approaches largely ignore the impact and the extent of organization-specific environmental changes within the stack. This article addresses these issues by proposing an predictive framework – DECIPHER (Database Environmental Change Impact Prediction in Human-driven tuning Efforts in Real-time) that not only acquires the organization-specific knowledge of the impact and the extent of environmental changes but does so in a proactive fashion. This framework, using the Oracle Data Miner (ODM), predicts the impact of environmental changes and its dependencies that have not yet been implemented by using predictive algorithms on the change management stores commonly found within organizations like incident reporting systems or emails or wikis.

Related Work

Database performance tuning is one of the most significant, time-consuming and repetitive tasks performed by the database administrators (DBAs) in order to meet the organization-specific performance goals (Belknap, Dagoveil, Dias, & Yagoub, 2009; Boughton, Martin, Powley, & Horman, 2006; Charvet, 2003; DBTA, 2009; Embarcadero-Technologies, 2010; Oliveira et al., 2006; Wiese, Rabinovitch, Reichert, & Arenswald, 2008). The DBAs that are able to perform such tuning successfully, efficiently and consistently are expensive and increasingly harder to find (Chaudhuri & Weikum, 2006; Krayzman, 2005; Schallehn, 2010; Sullivan, Selitzer, & Pfeffer, 2004; Wiese et al., 2008). This performance tuning work also usually takes the DBAs time away from focusing on strategic value added tasks required for an organization’s growth (Krayzman, 2005; TDA, 2009). Furthermore, this task can also be error prone which can introduce system unpredictability or even unavailability (Oliveira et al., 2006). Since organizations typically have a large number of database systems, the tuning task consumes most of the DBAs time preventing them from focusing on strategic and long-term value adding organizational initiatives (DBTA, 2009; Embarcadero-Technologies, 2010; Oliveira et al., 2006).

In order to address the aforementioned issues of manual performance tuning, the focus adopted by existing solutions can be broadly classified into autonomous and semi-autonomous tuning approaches. These approaches are proposed as potential solutions towards reducing or eliminating the need for human-driven performance tuning from a maintenance, administration and resource consumption perspective (Chaudhuri & Narasayya, 2007; Kephart & Chess, 2003; Shasha, 1992; Wiese et al., 2008):

### Autonomous Tuning Approaches

At a very high level, these approaches can be classified based on their integration with the database and the temporal nature (how and when) of their tuning decision (Chaudhuri & Weikum, 2006). This article assumes a database as a relational database system that is designed to function under all types of workloads. There are several specialized database technologies and architectures that are designed for specific performance requirements that are not considered in this article. For more information on such technologies/architectures, see Stonebraker et al. (2007) and Stonebraker (2010). Autonomous can be broadly categorized into tradeoff elimination-based (Vengurlekar et al., 2008), feedback-based (Herodotos Herodotou, 2010), exploration-based (Sullivan et al., 2004), model-based (Chaudhuri and Weikum, 2006) and hardware-based (Chaudhuri & Weikum, 2006; Herodotos Herodotou, 2010; Krayzman, 2005; Schallehn, 2010; Shasha, 1992; Sullivan et al., 2004). A high level summary of these approaches are shown in Table 1.

<table>
<thead>
<tr>
<th>Tuning</th>
<th>Pros and Cons</th>
<th>References</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feedback-Based</td>
<td>Exits feedback-based designs. Provides close integration with the database.</td>
<td>Chaudhuri and Weikum, 2006.</td>
</tr>
<tr>
<td>Exploration-Based</td>
<td>Provides a theoretically sound solution.</td>
<td>Wiese and Rabinovitch, 2008.</td>
</tr>
<tr>
<td>Model-Based</td>
<td>Works on a model and optimizes the model.</td>
<td>Sullivan et al., 2004; Tandella et al., 2008.</td>
</tr>
<tr>
<td>Hardware-Based</td>
<td>Provides a theoretically sound solution.</td>
<td>Chaudhuri and Weikum, 2006.</td>
</tr>
</tbody>
</table>

Tradeoff elimination-based approaches are based on principle that if a policy or high level parameter or knob provides near optimal results (sweet-spot) under unseen or changing workloads then its low level knobs or parameters can be eliminated (Chaudhuri & Weikum, 2006; Schallehn, 2010). Further, this approach requires detailed understanding of the low level parameters and their sensitivities (Chaudhuri & Weikum, 2006) and typically these approaches are closely integrated to the database internals (Vengurlekar, Vallath, & Long, 2008).

Feedback-based methodologies largely employ exploitation or control-loop or pay-as-you-go approaches towards performance tuning. Such methodologies follow a step-wise performance tuning approach wherein one parameter or knob is changed at a time based on some pre-defined threshold value or recorded or saved tuning plans in order to adapt to various database workloads (Rabinovitch & Wiese, 2007; Sullivan et al., 2004). Feedback-based approaches are also closely integrated to the database system (Chaudhuri & Weikum, 2006; Markl, Lohman, & Raman, 2003).

Exploration-based methodologies utilize explorative or comparison-based approaches wherein comparisons can be made...
proactively or even reactively with past measurements of parameters or knobs in order to reach an optimal value using an empirical or experimental exploration process (Herodotos Herodotou, 2010; Sullivan et al., 2004). These approaches are static in nature; i.e., tuning is not performed continuously but initiated by the database system (Schallehn, 2010).

Model-based methodologies usually employ approaches that use probabilistic or statistical models that can predict the database system's performance under various workload situations (Sullivan et al., 2004). These approaches are also static in nature; i.e., not performed continuously but initiated by the database system (Schallehn, 2010). Also, in this approach the decision-making and execution of the decision can be de-coupled with the database system and can also be supported by external tools (Chaudhuri & Weikum, 2006; Schallehn, 2010).

Hardware-based tunings employ solutions that involve hardware upgrades or hardware accelerators to improve the performance of a database system under different workloads (Chaudhuri & Weikum, 2006; Krayzman, 2005; Mueller & Teubner, 2009).

**Semi-Autonomous Tuning Approaches**

There are very few research efforts compared to autonomous tuning approaches that focus on semi-autonomous approaches that combine autonomous tuning approaches with the human knowledge. Sullivan, et al., (2004) research proposes a probabilistic reasoning approach as part of a model-based tuning approach to automate software tuning in general. The author in this research proposes that the domain experts with detailed knowledge of internal workings of a system construct initial models for inter-dependent low level system functions in order to attain the desired performance goal. These models can be trained under various workloads to automatically handle tuning of various knobs to achieve the desired tuning goals. Such an approach can be a very challenging task to do for today’s database systems, given their internal complexity. Also, this approach solely focuses on tweaking or tuning of so called knobs or parameters and may not work for performance issues that either do not have tunable knobs or may require tuning that is applicable to other components within a database environment (e.g. application, database, operating system, network, and storage and system hardware). Rabinovitch (2009) research formalizes the DBAs database-specific tuning knowledge into textual information called “tuning plans” and saves them in a best practice repository. Policies are then used to activate or deactivate these plans to address the performance problem as part of a feedback-based tuning methodology. Other approaches in this category focus on the human database tuner user either reviewing the solutions provided by the autonomous approaches or providing higher level workload-specific goals or policies (Herodotos Herodotou, 2010; Ziauddin, Das, Su, Zhu, and Yagoub, 2008).

**Limitations with the Existing Approaches**

Since performance tuning is a context-dependent knowledge-intensive task, the component based reference tuning knowledge model proposed by Wiese and Rabinovitch (2009) can be used to generalize the existing autonomous and semi-autonomous tuning approaches and help us better understand the limitations with these approaches. This model lays out the knowledge components required for successful database performance tuning in a controlled environment. In this layered model, shown in Figure 2 below, each knowledge component builds on top of the other. The foundation of this model is the database workload knowledge and it considers workload changes as the only source of changes in a database environment. Tuning policy knowledge refers to the specific knowledge of resources that need to be monitored or changed along with their specific thresholds and user-defined performance goals. Problem resolution knowledge refers to the database specific procedural knowledge on actions needed to resolve the performance problem. Problem diagnosis refers to best practices used by DBAs to diagnose or troubleshoot performance problems. Database internals knowledge refers to the expert knowledge of the internal workings of database system components and understanding of their inter-dependencies and inter-reactions.

As evident from Figure 2 above, the foundational knowledge component for the autonomous and semi-autonomous tuning approaches is the workload knowledge. These approaches do not account for changes that organizational database environments typically go through that have an effect on the database performance. Hence, it’s crucial that we consider the knowledge of impact and extent of environmental changes within the stack besides just the workload changes. The tuning knowledge model can be extended to incorporate the environment change impact knowledge (impact and extent of environmental changes) as shown in grid pattern in figure 3 below.
(Kephart & Chess, 2003) provides an adaptive approach for the design of DECIPHER. According to this architecture, the knowledge is the central component and should provide a common and shared understanding of the environment and problem space (Bell, 2004; Miller, 2005). The two main components of this architecture are the managed element (Database Environment Stack) and the tuner (Human or Autonomic agent). The sensor (Human or Autonomic agent) collects and retrieves information about the current state of the database environment stack then compares it with expectations that are held in the knowledge base. The required action is executed by the effector (Human or Autonomic agent). Figure 4 below shows the instantiation of the autonomic architecture proposed by (Kephart & Chess, 2003) to include the complete database environment stack from Figure 1 and the extended knowledge model from Figure 4 that was discussed in the previous section.

The tuning process has four distinct areas of functionality (Bell, 2004; Kephart & Chess, 2003; Miller, 2005; Wiese & Rabinovitch, 2009):

1. Monitor: The step involves identifying the existing performance problem. This step utilizes the environmental change impact, database workload and tuning policy knowledge components.

2. Analyze: This step is the key knowledge intensive step of M-A-P-E. It involves troubleshooting or diagnosing the performance problem; e.g., enabling query tracing for poor performing application queries. This step utilizes the environmental change impact, database workload, problem diagnosis, tuning policy and database internals knowledge components. This step is also responsible for predicting the impact of changes (workload as well as environmental) and the dependencies of the impact on other environmental factors from a database performance perspective (Bell, 2004; Wiese & Rabinovitch, 2009).

3. Plan: This step involves coming up with a plan for actions that need to be taken to fix the performance problem; e.g., adding indexes. This step utilizes the environmental change impact, database workload, tuning policy, database internals and problem resolution knowledge components.

4. Execute: This step involves the execution of the plan to carry out the actions. This step utilizes the problem resolution knowledge component.

Most organizations use some or other form of a change management store (e.g., email, database or wikis) to manage the changes to their Information Technology (IT) environment, including the database environments (Forrester, 2007; Hass, 2003). Changes to any production environment component (e.g., operating system, hardware, and database) are facilitated by this change management system (Conradi & Westfechtel, 1998). Furthermore, the increase of regulatory compliance needs (e.g., Sarbanes-Oxley) have also pushed for a wider adoption of change management processes and tools for achieving better traceability (Chen, Kurtz, & Lee, 2009). Typically, the change management stores have a vast amount of information about the changes to the organizational IT environment in unstructured form like notes or comments; e.g., “added more memory to a database server to increase system performance” or “changed kernel parameters for the operating system to fix swapping issue.” Before a change is executed in production environments, they go through some form of formal or informal approval process. The approver for these change requests is typically the stakeholders responsible for the target system. Change requests typically include information about the change purpose, date of their execution, the change work plan, perceived impact and a plan for reversing the changes if they result in any issues.

**DECIPHER Design**

At the core of DECIPHER are two major modules a) Impact Prediction Module (IPM): responsible for impact prediction of new environmental changes in real-time and b) Impact Extent Identification Module (IEIM): responsible for the extent identification of change factors that are dependent on new incoming changes in real-time. A high level DECIPHER architecture is shown in Figure 5 below.

**DECIPHER Implementation**

Impact Prediction Module (IPM) Implementation: The IPM implementation involves the following steps:

a) Data Transformation or Term Extraction: The unstructured (text) data stored as comments within un-implemented tickets in historical organizational change management stores is transformed into a vector space model (term vectors or environmental change factor vectors) using term frequency–inverse document frequency (TF-IDF) measure. These are stored in a nested database column within the database. This transformed text is now ready to be used as any other attribute in the building, testing, and scoring...
Impact Extent Identification Module (IEIM) Implementation

IEIM implementation has the following steps:

a) Feature Extraction: this step uses the non-negative matrix factorization (NMF) algorithm to extract features from the term vector that was created as part of the data transformation stage of IPM. NMF is found to be very effective in text mining domains (Guduru, 2006; Lee & Seung, 1999). The NMF algorithm decomposes a text data matrix Amn, where columns are tickets and rows are terms, into the product of two lower rank matrices Wmk and Hkn represented by the below equation (Guduru, 2006). To prevent cancellation, NMF expects Amn and Hkn to have non-negative entries. The NMF algorithm employs an iterative procedure to modify the initial values of Wmk and Hkn so that the product approaches Amn (Guduru, 2006; Lee & Seung, 1999). The procedure terminates based on error convergence value or when the specified number of iterations is reached. Each user-defined feature after NMF decomposition is a linear combination of the original attribute set and has non-negative coefficients. For more information on NMF, refer to (Guduru, 2006; Lee & Seung, 1999). These features are the change factors that are saved in an Impact Extent Database (IED). Impacts with “Low” values are ignored for IEIM analysis because they represent localized change impact.

b) Similarity matching: This step probabilistically matches the unimplemented changes with the change factors in IED using the Jaro-Winkler distance similarity algorithm (Winkler & Nov, 2006). The features set whose change factors have a maximum number of matches with the new features are returned along with the change features ranked by their coefficients. The change factors within this feature set, excluding the ones that matched, identify the factors that are dependent on the new changes.

DECIPHER Module Build Validation and Evaluation

This section explains how the accuracy of DECIPHER modules will be evaluated and validated. Let’s first start with the Impact Prediction Module validation.

Impact Prediction Module (IPM) Validation: The validation of IPM involves answering the following two questions:

1. How well does IPM predict the impact of environmental changes?
2. How does SVM’s prediction accuracy compare with another classifier for impact prediction?

In order to validate these two points for IPM, we used the incident reporting or trouble ticketing system as the organizational change management store. Cases or tickets filed from 2008 - 2011 were used to train and test the IPM model. Sixty percent of the data was used for training the model and 40% for testing the model. The tickets were considered from the RT change management database across different organizational teams (Hardware, Database, User, Network, Operating system etc.). This enabled us to use the changes from all layers, shown in Figure 1, for the analysis. The SVM’s prediction accuracy were compared with Naïve Bayes classifier. Change tickets had the following information:

1. Ticket #
2. Ticket Subject (Text)
3. Change Purpose (Text)
4. Owner of the ticket
5. Approver of the ticket
6. Work-plan for the change (Text)
7. Impact (High or Medium or Low)
8. Back-out plan (Text)
9. General Comments/Notes (Text). For IPM, Impact field is the dependent or target attribute. All other variables are independent or predictor variables

Impact Extent Identification Module (IEIM) Validation:
The validation for IEIM involves, answering the following two questions:

1. Do patterns (change factors) reoccur in future years, and to what extent?
2. What is the time interval across which pattern (change factor) reoccurrence is the maximum and how does this impact prediction accuracy?

For IEIM model validation, two data sets were created across time intervals. One data set was treated as an old data set and the second as a newer data that is not yet implemented. After feature extraction using NMF, pattern reoccurrence was measured as the average of maximum similarity of the new feature set Ci (e.g. 2010 Q1 dataset) with older feature set Cj (e.g. 2009 Q1 dataset). This is represented as:

$$\\text{max}_{\text{similarity}}(C_i, C_j)$$

For time interval validation, pattern reoccurrence was calculated with following configurations a. 2009 (Q1) with 2010 (Q1), b. 2009 (Q1 and Q2) with 2010 (Q1 and Q2) c. 2009 (full year) with 2010 (full year), d. 2008 and 2009 (full years) with 2010 (full year). This interval was used for deciding the ideal data set for IED database creation and also IPM model building.

DECIPHER Module Build Results:
The Oracle data miner (ODM) tool was used since the change management data for the incident reporting system was already in an Oracle database and also because ODM offers powerful in-database data mining algorithms like feature extraction, clustering, and classification on unstructured data (Guduru, 2006; Hamm & Burleson, 2006; Oracle, 2011).

First we will examine how well the IPM predicts the known values:
a) **IPM Predictive Confidence**: Predictive confidence provides an estimate of the overall goodness of the model. This indicates how much better the predictions made by the tested model were than predictions made by a naive model (Oracle, 2011). The naive model always predicts the mean for numerical targets and the mode for categorical targets. The following formula defines Predictive Confidence (Oracle, 2011):

\[
\text{Predictive Confidence} = \text{MAX} \left( \frac{1 - (\text{error of model})}{\text{error of naive model}} \right), 0
\]

If predictive confidence is 0, the model’s predictions are no better than predictions made using the naive model. If predictive confidence is 1, the predictions are perfect. IPM’s support vector machine (SVM) model is ~ 50% better than naïve model and 13% better than naïve bayes (NB) model.

b) **IPM Performance Matrix (Confusion Matrix)**: This measured the probability of the model to predict incorrect and correct values and also indicated the types of errors that the model is likely to make. This is shown in Figure 6.

IPM’s SVM model has identified over 80% accurate predictions. This matrix displays the number of correct and incorrect predictions made by the model compared with the actual classifications in the test data. The performance matrix is calculated by applying the model to a hold-out sample (the test set, created during the split step in a Classification activity) taken from the build data. Now let’s examine the results of IEIM:

a) **IEIM Pattern Reoccurrence %**: This helps us understand if the patterns (change factors) reoccur in future years.

<table>
<thead>
<tr>
<th>#</th>
<th>Old Data Set</th>
<th>New Data Set</th>
<th>IEIM Pattern (Change Factors) Reoccurrence %</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2009 Quarter 1</td>
<td>2010 Quarter 1</td>
<td>59.96</td>
</tr>
<tr>
<td>2</td>
<td>2009 Quarter 1 &amp; 2</td>
<td>2010 Quarter 1 &amp; 2</td>
<td>75.608</td>
</tr>
<tr>
<td>3</td>
<td>2009 Full Year</td>
<td>2010 Full Year</td>
<td>91.301</td>
</tr>
<tr>
<td>5</td>
<td>2008 &amp; 2009 Full Years</td>
<td>2010 Full Year</td>
<td>90.225</td>
</tr>
</tbody>
</table>

The results in Table 2 show that there is a significant change factor reoccurrence between the old and new change data sets. Also, these numbers highlight the repetitive aspect of the environmental changes.

b) **IEIM Prediction accuracy and optimal time interval**: This helps us understand the optimum interval across which change factors reoccur and its effect on the impact prediction accuracy. Based on the results in Table 2, the optimum time interval is a full complete year. It is also evident from these results that going beyond the year for old data sets actually drops the reoccurrence percentage. This makes a lot of sense because for most organizations the change cycle is annual. This time interval can also be used to IPM instead of the last 4 years of data that we had used earlier. This will significantly speed up the model build process.

**Scoring with DECIPHER**

This section explains the real-time scoring of DECIPHER. Let’s take an example of a change ticket that is not yet implemented to understand the scoring of DECIPHER. The un-implemented ticket has the following info:

1) Ticket #: 14356
2) Ticket Subject (Text) : Update firewall rule policy
3) Change Purpose (Text) : We need to update the firewall rule policy to add new forecasting BI server to sales portal.
4) Owner of the ticket: Dylan
5) Approver of the ticket: Susan
6) Work-plan for the change (Text): The new firewall rule for salesportal_dmz is as below: 123.0.xx..
7) Impact (High or Medium or Low): Low
8) Back-out plan (Text): Revert the rule
9) General Comments/Notes (Text)

DECIPHER IPM Scoring Result for ticket 14356:

<table>
<thead>
<tr>
<th>Predicted Impact</th>
<th>Prediction Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>High</td>
<td>90%</td>
</tr>
</tbody>
</table>

DECIPHER IEIM Scoring Results for ticket 14356:

<table>
<thead>
<tr>
<th>#</th>
<th>Dependent Change Factor</th>
<th>Coefficient</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Sales_recos_db</td>
<td>0.13</td>
</tr>
<tr>
<td>2</td>
<td>connection</td>
<td>0.12</td>
</tr>
<tr>
<td>3</td>
<td>hang</td>
<td>0.12</td>
</tr>
<tr>
<td>4</td>
<td>port</td>
<td>0.11</td>
</tr>
<tr>
<td>5</td>
<td>1924</td>
<td>0.10</td>
</tr>
</tbody>
</table>

...continued on page 23
Central Repository for Oracle
Automatic Workload Repository (AWR)

by Mike Messina
Senior Managing Consultant, Rolta Solution, TUSC Infrastructure Services

The Oracle Automatic Workload Repository started with UTLBSTAT/UTLESTAT, evolved into Statspack, and with Oracle 10g became the Oracle Automatic Workload Repository (AWR). The AWR data are collected via snapshots manually or at defined intervals (default 60 min) and kept for a defined period of time (default 7 days 10g and 8 days 11g). The interval and length of time snapshots are taken is configurable via the DBMS_WORKLOAD_REPOPOSITORY package.

AWR allows the creating of AWR reports using a starting snap and an ending snap for a database. AWR also allows the comparison of two different pairs of snaps, allowing the compare of workload from two different time periods within the same database. ADDM analysis, by default, is done each time an AWR snapshot is taken. However, to specifically analyze for times in the past, an ADDM report can be run using the addmrpt.sql script in ORACLE_HOME/rdbms/admin or an analysis can be done using the DBMS_ADVISOR package.

What if you want to keep your AWR data for a long period of time to use for performance analysis over time?

What if you wanted to compare a load test from a staging database to the current actual production database?

Creating a central AWR repository such as this will reduce the amount of data that would have to be kept in the live production database. The Central AWR will also allow you to have all the AWR data from all your databases in a single location. This will allow running of reports, analysis of data and long term storage without affecting live databases. Once the AWR data are loaded into the central AWR, reports can be executed using the central AWR.

Database for Central AWR

We need a database that will store that AWR data from all the databases in our environment. The database needs to be at the highest version of the database in your environment or greater. For example if you have a 10.2.0.4, 11.1.0.7 and 11.2.0.1 databases in your environment, the central AWR must be at 11.2.0.1 or higher. Many have used the OEM Grid Control Repository database as their Central AWR database to cut down on an additional database to support. We can load AWR data from earlier database versions but cannot load data from later versions into earlier versions.

In order to set up a centralized AWR repository, we would need the following, in addition to the database, which will act as the repository itself:

- A database schema that will own the central AWR objects with the permissions to own tables, stored procedures and directories as well as select access to the AWR data.
- A file system on each of the servers hosting the source database as well as the centralized AWR database. This file system does not need to be very large and based on number of targets, but environments with 100 databases have used < 2GB as it is only to facilitate the extract and loading of AWR data.
- Create an Oracle directory object pointing to this new file system in each of the source databases from which we will be extracting the AWR data.
- Create an Oracle directory object pointing to this new file system in the Central AWR database.

Extract AWR Data from Database(s)

To extract AWR data from you databases we have two options:

- Manually using the script $ORACLE_HOME/rdbms/admin/awrextr.sql provided by Oracle and then copying the extracted file manually to the central AWR database server at the location where it will be loaded from. This will utilize Data Pump therefore you must ensure that a directory location and database directory object exists for the Datapump export.
- Automate the process by creating a stored procedure with the SYS.DBMS_SWRF_INTERNAL.AWR_EXTRACT package procedure and DBMS_FILE_TRANSFER.PUT_FILE to move the extract to the server housing the central AWR repository. Use DBMS_SCHEDULER to automate the extracts at regular intervals. This too will utilize datapump and therefore require a directory location and database directory created for the datapump export.

Example Manual AWR Extract
SQL> @awrextr.sql
~~~~~~~~~~~~
AWR EXTRACT
~~~~~~~~~~~~
- This script will extract the AWR data for a range of snapshots into a dump file. The script will prompt users for the following information:
Choose a Directory Name from the above list (case-sensitive).

Enter value for directory_name: AWR
Using the dump directory: AWR

Specify the Name of the Extract Dump File
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
The prefix for the default dump file name is awrdat_300_312. To use this name, press <return> to continue, otherwise enter an alternative.

Enter value for file_name:
Using the dump file prefix: awrdat_300_312
| The AWR extract dump file will be located in the following directory/file:
| /u01/app/oracle/AWR
| awrdat_300_312.dmp
| ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
| *** AWR Extract Started ...
| | This operation will take a few moments. The progress of the AWR extract operation can be monitored in the following directory/file:
| | /u01/app/oracle/AWR
| | awrdat_300_312.log
| | Load AWR data into Central AWR Repository from Database(s)

There are two options for loading of the AWR data into the Central AWR database:

- Manually via the $ORACLE_HOME/rdbms/admin/awrload.sql script provided by Oracle. This will utilize Datapump for the import and therefore a directory location and database directory object must exist for the Datapump import and the file to be imported placed in the directory location.
- Automate the process by creating a stored procedure that will load the AWR files using the SYS.DBMS_SWRF_INTERNAL.AWR_LOAD and SYS.DBMS_SWRF_INTERNAL.MOVE_TO_AWR packaged procedures. These utilize Datapump for the import and therefore a directory location and database directory object must exist for the Datapump import and the file to be imported placed in the directory location.

Since we are automating the load from several databases you will need to ensure that the files dropped into the load location have unique names so that no files get overwritten. Since we could potentially be loading a large number of files, use SYS.DBSMS_BACKUP_RESTORE.SEARCHFILES to generate a list of files to loop through for processing.

This procedure can be scheduled via the DBMS_SCHEDULER, fully automating the load of AWR extracts from many databases on a regular interval.
Note: When loading AWR data into the central repository make sure the tablespace SYSAUX, the stage tablespace selected, and the TEMP tablespace have enough room to complete the operation. It is recommended to have Oracle Enterprise Manager Grid Control or a custom process monitor free space in the central AWR database.

Example Manual AWR Load
SQL> @awrload.sql

AWR LOAD

~ This script will load the AWR data from a dump file. The ~
~ script will prompt users for the following ~
~ information: ~
~ (1) name of directory object ~
~ (2) name of dump file ~
~ (3) staging schema name to load AWR data ~

Specify the Directory Name
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Directory Name     Directory Path
----------------   --------------------
AWR             /u01/app/oracle/AWR
DATA_PUMP_DIR   /u01/app/oracle/admin/CAWR/dpdump/
EMREP_DIR_NAME  /u01/app/oracle/product/11.2.0/
                 dbhome_1/ccr/state
ORACLE_OCM_CONFIG_DIR /u01/app/oracle/prd
                      uct/11.2.0/dbhome_1/ccr/state
XMLDIR         /ade/b/2125410156/oracle/rdbms/xml

Choose a Directory Name from the list above (case-sensitive).

Enter value for directory_name: AWR

Using the dump directory: AWR

Specify the Name of the Dump File to Load
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Please specify the prefix of the dump file (.dmp) to load:

Enter value for file_name: awrdat_300_312

Loading from the file name: awrdat_300_312.dmp
Staging Schema to Load AWR Snapshot Data
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
The next step is to create the staging schema where the AWR snapshot data will be loaded. After loading the data into the staging schema, the data will be transferred into the AWR tables in the SYS schema.

The default staging schema name is AWR_STAGE. To use this name, press <return> to continue, otherwise enter an alternative.

Enter value for schema_name:

Using the staging schema name: AWR_STAGE

Choose the Default tablespace for the AWR_STAGE user
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Choose the AWR_STAGE user’s default tablespace. This is the tablespace in which the AWR data will be staged.

TABLESPACE_NAME CONTENTS DEFAULT TABLESPACE
----------- --------- ------------------
MGMT_ECM_DEPOT_TS PERMANENT
MGMT_TABLESPACE   PERMANENT
SYSAUX           PERMANENT *
USERS            PERMANENT

Pressing <return> will result in the recommended default tablespace (identified by *) being used.

Enter value for default_tablespace:

Using tablespace SYSAUX as the default tablespace for AWR_STAGE

Choose the Temporary tablespace for the AWR_STAGE user
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Choose the AWR_STAGE user’s temporary tablespace.

TABLESPACE_NAME CONTENTS DEFAULT TEMP TABLESPACE
----------- --------- ---------------------
TEMP        TEMPORARY *

Pressing <return> will result in the database’s default temporary tablespace (identified by *) being used.

Enter value for temporary_tablespace:

Using tablespace TEMP as the temporary tablespace for AWR_STAGE

... Creating AWR_STAGE user

Loading the AWR data from the following directory/file:

```
/u01/app/oracle/AWR
awrdat_300_312.dmp
```

... AWR Load Started ...

This operation will take a few moments. The progress of the AWR load operation can be monitored in the following directory/file:
Maintaining Central AWR

To maintain the AWR snapshots in the local databases where the AWR data is being extracted from, we can utilize the standard AWR maintenance procedures already supplied for the Oracle database. Therefore, there are no additional processes or procedures to develop or maintain for this. For the Central AWR Repository database you can manually purge using the DBMS_WORKLOAD_REPOSITORY.DROP_SNAPSHOT_RANGE or create a stored procedure that uses the DBMS_WORKLOAD_REPOSITORY.DROP_SNAPSHOT_RANGE using a defined keep period. The procedure can then be scheduled at regular intervals using the DBMS_SCHEDULER.

Running AWR Report from Central AWR

Running an AWR Report from the central AWR Repository is much like running any AWR report except from the central AWR Repository we need to indicate the DBID that the report is to be executed for. We do this using the Oracle supplied script awrrpti.sql located in $ORACLE_HOME/rdbms/admin. This will allow us to specify the DBID and instance, which can vary for RAC environments, for the report.

```
SQL> @awrrpti

Specify the Report Type
~~~~~~~~~~~~~~~~~~~~~~~
Would you like an HTML report, or a plain text report?
Enter 'html' for an HTML report, or 'text' for plain text
Defaults to 'html'

Enter value for report_type: text

Type Specified:  text

Instances in this Workload Repository schema
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

<table>
<thead>
<tr>
<th>DB Id</th>
<th>Inst Num</th>
<th>DB Name</th>
<th>Instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>*: 1682379389</td>
<td>1</td>
<td>CAWR</td>
<td>CAWR</td>
</tr>
<tr>
<td>oel54node1.l</td>
<td>2001611575</td>
<td>1</td>
<td>PRDRMED</td>
</tr>
<tr>
<td>dm01db01.pro</td>
<td>2001611575</td>
<td>2</td>
<td>PRDRMED</td>
</tr>
<tr>
<td>744414708</td>
<td></td>
<td>1</td>
<td>ORCL11G</td>
</tr>
<tr>
<td>MRMESSIN</td>
<td>1545520884</td>
<td>1</td>
<td>PRDRMED</td>
</tr>
<tr>
<td>RMPRODRAC11</td>
<td>1203391489</td>
<td>1</td>
<td>ORCL</td>
</tr>
<tr>
<td>MRMESSIN</td>
<td>990749484</td>
<td>1</td>
<td>KAHN</td>
</tr>
<tr>
<td></td>
<td>1545520884</td>
<td>3</td>
<td>PRDRMED</td>
</tr>
</tbody>
</table>
```

Enter value for dbid: 2001611575
Using 2001611575 for database Id
Enter value for inst_num: 1
Using 1 for instance number

Specify the number of days of snapshots to choose from
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Entering the number of days (n) will result in the most recent (n) days of snapshots being listed. Pressing <return> without specifying a number lists all completed snapshots.

Enter value for num_days: 360

Listing the last 360 days of Completed Snapshots

```
<table>
<thead>
<tr>
<th>Snap</th>
<th>Instance</th>
<th>DB Name</th>
<th>Snap Id</th>
<th>Snap Started</th>
</tr>
</thead>
<tbody>
<tr>
<td>prdrmmed1</td>
<td>PRDRMED</td>
<td>1557</td>
<td>09 Nov 2010 19:39</td>
<td>1</td>
</tr>
<tr>
<td>prdrmmed1</td>
<td>PRDRMED</td>
<td>1558</td>
<td>09 Nov 2010 20:00</td>
<td>1</td>
</tr>
<tr>
<td>prdrmmed1</td>
<td>PRDRMED</td>
<td>1559</td>
<td>09 Nov 2010 20:07</td>
<td>1</td>
</tr>
<tr>
<td>prdrmmed1</td>
<td>PRDRMED</td>
<td>1574</td>
<td>10 Nov 2010 13:19</td>
<td>1</td>
</tr>
<tr>
<td>prdrmmed1</td>
<td>PRDRMED</td>
<td>1575</td>
<td>10 Nov 2010 14:00</td>
<td>1</td>
</tr>
<tr>
<td>prdrmmed1</td>
<td>PRDRMED</td>
<td>1576</td>
<td>10 Nov 2010 14:18</td>
<td>1</td>
</tr>
<tr>
<td>prdrmmed1</td>
<td>PRDRMED</td>
<td>1576</td>
<td>10 Nov 2010 14:18</td>
<td>1</td>
</tr>
<tr>
<td>prdrmmed1</td>
<td>PRDRMED</td>
<td>1576</td>
<td>10 Nov 2010 14:18</td>
<td>1</td>
</tr>
<tr>
<td>prdrmmed1</td>
<td>PRDRMED</td>
<td>1588</td>
<td>11 Nov 2010 11:45</td>
<td>1</td>
</tr>
<tr>
<td>prdrmmed1</td>
<td>PRDRMED</td>
<td>1589</td>
<td>11 Nov 2010 12:07</td>
<td>1</td>
</tr>
<tr>
<td>prdrmmed1</td>
<td>PRDRMED</td>
<td>1677</td>
<td>15 Nov 2010 10:00</td>
<td>1</td>
</tr>
<tr>
<td>prdrmmed1</td>
<td>PRDRMED</td>
<td>1678</td>
<td>15 Nov 2010 11:00</td>
<td>1</td>
</tr>
</tbody>
</table>
```

Specify the Begin and End Snapshot Ids
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Enter value for begin_snap: 1677
Begin Snapshot Id specified: 1677

Enter value for end_snap: 1678
End Snapshot Id specified: 1678

Specify the Report Name
~~~~~~~~~~~~~~~~~~~~~~

The default report file name is awrrpt_1_1677_1678.txt. To use this name, press <return> to continue, otherwise enter an alternative.

Enter value for report_name:

Using the report name awrrpt_1_1677_1678.txt

Running a Compare Report from Central AWR

Now that we have AWR data from multiple databases we can compare across snap time periods from one database, or even compare snap time periods across different databases, even if they are different versions. Oracle provides an easy way to execute the compares across the script awrddrpi.sql in the ORACLE_HOME/rdbms/admin directory.

```
SQL> @awrddrpi.sql
Specify the Report Type
```

Enter value for dbid: 2001611575
Using 2001611575 for database Id
Enter value for inst_num: 1
Using 1 for instance number

Specify the number of days of snapshots to choose from
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Entering the number of days (n) will result in the most recent (n) days of snapshots being listed. Pressing <return> without specifying a number lists all completed snapshots.

Enter value for num_days: 360

Listing the last 360 days of Completed Snapshots

Snap    | Instance | DB Name  | Snap Id    | Snap Started |
--------|----------|----------|------------|--------------|
prdrmmed1 | PRDRMED  | 1557     | 09 Nov 2010 19:39 | 1 |
prdrmmed1 | PRDRMED  | 1558     | 09 Nov 2010 20:00 | 1 |
prdrmmed1 | PRDRMED  | 1559     | 09 Nov 2010 20:07 | 1 |
prdrmmed1 | PRDRMED  | 1574     | 10 Nov 2010 13:19 | 1 |
prdrmmed1 | PRDRMED  | 1575     | 10 Nov 2010 14:00 | 1 |
prdrmmed1 | PRDRMED  | 1576     | 10 Nov 2010 14:18 | 1 |
prdrmmed1 | PRDRMED  | 1576     | 10 Nov 2010 14:18 | 1 |
prdrmmed1 | PRDRMED  | 1576     | 10 Nov 2010 14:18 | 1 |
prdrmmed1 | PRDRMED  | 1588     | 11 Nov 2010 11:45 | 1 |
prdrmmed1 | PRDRMED  | 1589     | 11 Nov 2010 12:07 | 1 |
prdrmmed1 | PRDRMED  | 1677     | 15 Nov 2010 10:00 | 1 |
prdrmmed1 | PRDRMED  | 1678     | 15 Nov 2010 11:00 | 1 |

Specify the Begin and End Snapshot Ids
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Enter value for begin_snap: 1677
Begin Snapshot Id specified: 1677

Enter value for end_snap: 1678
End Snapshot Id specified: 1678

Specify the Report Name
~~~~~~~~~~~~~~~~~~~~~~

The default report file name is awrrpt_1_1677_1678.txt. To use this name, press <return> to continue, otherwise enter an alternative.

Enter value for report_name:

Using the report name awrrpt_1_1677_1678.txt

Running a Compare Report from Central AWR

Now that we have AWR data from multiple databases we can compare across snap time periods from one database, or even compare snap time periods across different databases, even if they are different versions. Oracle provides an easy way to execute the compares across the script awrddrpi.sql in the ORACLE_HOME/rdbms/admin directory.

```
SQL> @awrddrpi.sql
Specify the Report Type
```
Would you like an HTML report, or a plain text report?
Enter 'html' for an HTML report, or 'text' for plain text
Defaults to 'html'

Enter value for report_type: text

Type Specified: text

Instances in this Workload Repository schema
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

<table>
<thead>
<tr>
<th>DB Id</th>
<th>Inst Num</th>
<th>DB Name</th>
<th>Instance</th>
<th>Host</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>CAWR</td>
<td>CAWR</td>
<td>oel54node1.localdomain</td>
</tr>
<tr>
<td>2001611575</td>
<td>1</td>
<td>PRDRMED</td>
<td>prdrmed1</td>
<td>dm01db01.prod.real-med.com</td>
</tr>
<tr>
<td>2001611575</td>
<td>2</td>
<td>PRDRMED</td>
<td>prdrmed2</td>
<td>dm01db02.prod.real-med.com</td>
</tr>
<tr>
<td>744414708</td>
<td>1</td>
<td>ORCL11G</td>
<td>orcl11g</td>
<td>MRMESSIN</td>
</tr>
<tr>
<td>1545520884</td>
<td>1</td>
<td>PRDRMED</td>
<td>prdrmed1</td>
<td>RMPRODRAC11</td>
</tr>
<tr>
<td>1203391489</td>
<td>1</td>
<td>ORCL</td>
<td>orcl</td>
<td>MRMESSIN</td>
</tr>
<tr>
<td>990749484</td>
<td>1</td>
<td>KAHN</td>
<td>kahn</td>
<td>MRMESSIN</td>
</tr>
<tr>
<td>1545520884</td>
<td>3</td>
<td>PRDRMED</td>
<td>prdrmed13</td>
<td>RMPRODRAC13</td>
</tr>
</tbody>
</table>

Database Id and Instance Number for the First Pair of Snapshots
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Enter value for dbid: 2001611575

Using 2001611575 for Database Id for the first pair of snapshots

Enter value for inst_num: 1

Using 1 for Instance Number for the first pair of snapshots

Specify the number of days of snapshots to choose from
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Entering the number of days (n) will result in the most recent (n) days of snapshots being listed. Pressing <return> without specifying a number lists all completed snapshots.

Enter value for num_days: 360

Listing the last 360 days of Completed Snapshots

<table>
<thead>
<tr>
<th>Snap</th>
<th>Instance</th>
<th>DB Name</th>
<th>Snap Id</th>
<th>Snap Started</th>
<th>Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>prdrmed1</td>
<td>PRDRMED</td>
<td>1557 09 Nov 2010 19:39</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>prdrmed1</td>
<td>PRDRMED</td>
<td>1558 09 Nov 2010 20:00</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>prdrmed1</td>
<td>PRDRMED</td>
<td>1559 09 Nov 2010 20:07</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>prdrmed1</td>
<td>PRDRMED</td>
<td>1574 10 Nov 2010 13:19</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>prdrmed1</td>
<td>PRDRMED</td>
<td>1575 10 Nov 2010 14:00</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>prdrmed1</td>
<td>PRDRMED</td>
<td>1576 10 Nov 2010 14:18</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>prdrmed1</td>
<td>PRDRMED</td>
<td>1588 11 Nov 2010 11:45</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>prdrmed1</td>
<td>PRDRMED</td>
<td>1589 11 Nov 2010 12:07</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>prdrmed1</td>
<td>PRDRMED</td>
<td>1677 15 Nov 2010 10:00</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>prdrmed1</td>
<td>PRDRMED</td>
<td>1678 15 Nov 2010 11:00</td>
<td>1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Specify the First Pair of Begin and End Snapshot Ids
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Enter value for begin_snap: 1677

First Begin Snapshot Id specified: 1677

Enter value for end_snap: 1678

First End Snapshot Id specified: 1678

Instances in this Workload Repository schema
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

<table>
<thead>
<tr>
<th>DB Id</th>
<th>Inst Num</th>
<th>DB Name</th>
<th>Instance</th>
<th>Host</th>
</tr>
</thead>
<tbody>
<tr>
<td>1545520884</td>
<td>1</td>
<td>PRDRMED</td>
<td>prdrmed1</td>
<td>RMPRODRAC11</td>
</tr>
<tr>
<td>1545520884</td>
<td>3</td>
<td>PRDRMED</td>
<td>prdrmed13</td>
<td>RMPRODRAC13</td>
</tr>
</tbody>
</table>

Database Id and Instance Number for the Second Pair of Snapshots
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Enter value for dbid2: 1545520884

Using 1545520884 for Database Id for the second pair of snapshots

Enter value for inst_num2: 1

Using 1 for Instance Number for the second pair of snapshots

Specify the number of days of snapshots to choose from
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Entering the number of days (n) will result in the most recent (n) days of snapshots being listed. Pressing <return> without specifying a number lists all completed snapshots.

Enter value for num_days2: 360

Listing the last 360 days of Completed Snapshots

<table>
<thead>
<tr>
<th>Snap</th>
<th>Instance</th>
<th>DB Name</th>
<th>Snap Id</th>
<th>Snap Started</th>
<th>Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>prdrmed11</td>
<td>PRDRMED</td>
<td>30108 08 Nov 2010 10:30</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>prdrmed11</td>
<td>PRDRMED</td>
<td>30109 08 Nov 2010 10:45</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>prdrmed11</td>
<td>PRDRMED</td>
<td>30110 08 Nov 2010 11:00</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>prdrmed11</td>
<td>PRDRMED</td>
<td>30111 08 Nov 2010 11:14</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>prdrmed11</td>
<td>PRDRMED</td>
<td>30112 08 Nov 2010 11:15</td>
<td>1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Enter value for end_snap2: 30308

Specify the Report Name

The default report file name is awrdiff_1_1677_1_30307.txt. To use this name, press <return> to continue, otherwise enter an alternative.

Enter value for report_name:

Using the report name awrdiff_1_1677_1_30307.txt

Screen Shots of AWR Compare Report from Central AWR Using an Application Express Front End

Michael Messina is a Senior Managing Consultant with Rolta Solutions, TUSC Infrastructure Services. An Oracle ACE, Oracle Certified Professional, Oracle RAC Administrator Certified Expert and Exadata Implementation Specialist with nearly 20 years experience with the Oracle Databases. He has contributed to several chapters in the newest Oracle Database 11g Release 2 Performance and Tuning Tips & Techniques from Rich Niemiec. Michael has presented at Open World, RMOUG Training Days, IOUG Collaborate as well as many local and regional users groups throughout the U.S. Michael’s experience includes Oracle and MySQL database administration and implementation, system and infrastructure implementation. Michael has led several performance improvement, maintenance, and implementation projects for large highly available systems.
Login security is the baseline defense to any kind of cyber attack. If the computer community would follow some simple guidelines, hacking into one’s email, Facebook, or database would be nearly impossible.

Simple things like:

- change your password with some degree of frequency,
- that the password be of at least 8 positions in length (I can’t find the detail but I read recently where it takes hours to hack a 6 position password, days for an 8 digit password, and nearly impossible for a 12 digit password),
- that this password contains a mix of upper case/lower case/numbers/special characters
- that any character string is not found in the dictionary

Additional items can include not allowing part of the user ID as the password, and that the password not be the same (or nearly the same) as past passwords used.

At the end of this article, I’ll illustrate the user Bob’s fancy password scheme that is probably more common than one would think.

APEX 4.1 can insure quite the password complexity. This article will focus just on APEX password security.

http://<hostid:8080/apex/apex_admin Manage Instance → Security

This screen shows the tab APEX Password Policy parameters enforced for all who use this instance of APEX.

The APEX password policy requirements are rather straightforward:

- A minimum password length can be specified
- A minimum number of differences from the current password can be specified
- The next series enforces a strong password policy of containing alpha, numeric, special characters, and upper/lower case characters.
- APEX also allows for certain character strings to NOT appear in the password
- The characters and punctuation characters can be defined

The next tab over allows for APEX to use this defined policy definition or the password policy as it is defined at the database level.

The upside is that these guidelines will make for a very strong password; the only downside I see is APEX 4.1 does not seem to track password history or allow for dictionary words to be excluded.

Why does the password policy appear to be not used in this screen shot? Because this is a screen shot from my APEX training
environment and I want APEX account passwords to be the same as the user ID!

The prior tab, Account Login Control, allows for passwords to expire, the number of attempts someone can try to gain access to an APEX account, and the life span in days of the password. Again, I have turned this feature off for my training lab environment.

By the way, Bob’s password is his name spelled backwards…all lower case! Easy to remember but not such a good idea.

Follow my blog for Q&A traffic from my user community: DanHotka.blogspot.com. Have a great day…

Dan Hotka is a Training Specialist and an Oracle ACE Director who has over 32 years in the computer industry, over 27 years of experience with Oracle products. His experience with the Oracle RDBMS dates back to the Oracle V4.0 days. Dan’s latest book is the Oracle SQL Tuning: A Close Look at Explain Plans by Amazon. He is also the author of TOAD Handbook by Pearson, SQL Developer Handbook by Oracle Press, Oracle9i Development By Example, and Oracle8i from Scratch by Que and has co-authored 7 other popular books including the Database Oracle10g Linux Administration by Oracle Press. He is frequently published in Oracle trade journals, and regularly speaks at Oracle conferences and user groups around the world.

---

Centroid is an Oracle Platinum Partner that focuses on implementing and supporting Oracle Solutions. From Infrastructure to Technology to Applications, we help customers realize the benefits and increase the value of their investment by running a fully-integrated Oracle suite of solutions. Our Oracle experts are focused on leveraging technology investments for increased value.

Centroid’s FutureProofing concept helps companies adapt the right Oracle technology at the right time as their business grows.

Contact us today and talk to one of our FutureProofing experts.
Summer QEW at Elitch Gardens in Denver. What could be more fun! Lunch sponsored by Arisant, followed by presentations and a great family day. If you missed it, please join us next year.
Many thanks to our sponsors - Arisant and Oracle Corporation
Shortest Isn’t Always Fastest

by Tim Gorman

It’s a problem many of us in Colorado have encountered; we want to drive from Winter Park to Denver. Easy: go over Berthoud Pass, down to I-70, and take I-70 east all the way in Denver. Should only take an hour and a half, tops.

Except when Berthoud Pass has been closed due to winter weather. What then? Do we start heading east on US Highway 40 until the snow becomes higher than the windshield? Don’t laugh, plenty of people would do essentially just this.

Or, would you turn the other way, head west on US Highway 40, all the way to Kremmling, then follow Highway 9 down to Silverthorne to I-70, through the Eisenhower Tunnel, and on into Denver. Yes, this route covers three times as many miles, but you get to Denver faster. It’s not a great choice, but first step is the most important. Turning in the opposite direction, and going that way as fast as you can. It’s counter-intuitive, but sometimes it works.

When loading large volumes of data into an Oracle database, the same thing can true.

Let’s suppose you have 100,000 rows of data to load into a slowly-changing dimension table in a data warehouse in an Oracle database? Of those 100,000 rows, 50,000 are new, so they need to be inserted. The other 50,000 rows already have older copies in the table, so those rows will need to be updated with the new data. Often, this is called an “up-sert” or “merge” load, a combination of INSERTs and UPDATEs. Most dimension tables have new rows inserted or modified rarely, if at all. However, there are often dimensions that change a little (or a lot) with every load cycle. Typically, these dimensions have something to do with people, such as a dimension for employees, or financial accounts, or members.

There are a number of ways to merge this data, the most straightforward of this to use the Oracle MERGE command, similar to the following...

MERGE into CURR_ACCT_DIM D
USING (select acct_key from EXT_ACCT_DIM) X
ON (d.acct_key = x.acct_key)
WHEN MATCHED then update set d.eff_dt = x.eff_dt,
  d.attr_01 = x.attr_01, d.attr_02 = x.attr_02, ..., d.attr_99 = x.attr_99
WHEN NOT MATCHED then insert (d.acct_key, d.eff_dt,
  d.attr_01, d.attr_02, ..., d.attr_99)
  values (x.acct_key, x.eff_dt, x.attr_01, x.attr_02, ..., x.attr_99);

This does the job, but like the story about driving to Denver over Berthoud Pass, sometimes the most direct and straightforward route is not the fastest way to the destination, especially when a large number of rows are being merged, and most particularly when a large number of rows are being updated.

In many ways, INSERTs are the fastest and most efficient type of data modification in any database, Oracle or otherwise.

• With an INSERT, there is no “search” phase to the operation, just placement of the row(s) into database blocks. From the perspective of transaction logging, not much redo (comparatively) is generated, just the “after image” of the newly-inserted row.
• With a DELETE, there is a little more to do. There is a “search” phase to find the row, then remove the row. From the perspective of transaction logging, not much redo is generated, just the “before image” of the newly-deleted row from the UNDO segment.
• With an UPDATE, there is a lot to do. There is a “search” phase to find the row, then there is the matter of changing data values in some or all of the columns. If the row grows in size and can no longer fit within the database block, then it must be migrated (similar to “chaining”). And, from the perspective of transaction logging, there is a lot of redo generated, both the “before-image” prior to the row being changed, stored within UNDO, and the “after-image” after the row has been updated.

Compared to DELETEs and UPDATEs, INSERT operations are a breeze. And they scale very well when run in parallel. And there is even a special version of INSERT operations, called “direct-path” used for large volumes of data being inserted.

And, as it turns out, INSERTs are the very fastest way to accomplish a boatload of DELETEs or UPDATEs, if you’re fortunate enough to be using table partitioning.

Here’s how...

First, the easy part - the very fastest DELETE is an INSERT, and here is how...

For each of the partitions in the table from which rows will be DELETEd...

1. Create a “scratch” table that has the same columns and physical storage attributes as the partition, using a CREATE TABLE ... AS SELECT (CTAS) statement which queries the partition from which rows will be removed. The WHERE clause of the SELECT statement in the CTAS is the reverse condition of a DELETE statement, finding the rows which will be retained
   • The INSERT portion and the SELECT portion of the CTAS statement can both be run in PARALLEL.
   • The INSERT portion can use the “direct-path” (a.k.a. APPEND) load mechanism in the CTAS statement.
   • The NOLOGGING clause can be used in the CREATE attributes of the CTAS statement.
   • The COMPRESS clause can be used in the CREATE attributes of the CTAS statement.
2. Perform ALTER TABLE ... EXCHANGE PARTITION between the original partition and the newly-created “scratch” table
   • As long as there are no global indexes, the EXCHANGE PARTITION operation is practically instantaneous.
3. Drop the “scratch” table
4. Repeat for all partitions involved

Fast, easy, and scalable. This is the way to delete hundreds of thousands or millions of rows from a table.

But UPDATEs aren’t so easy. So I’m going to set up a scenario to help explain how it can be done. We have a dimension table named CURR_ACCT_DIM which is loaded into nightly. During these nightly loads, 50,000 rows are typically inserted and 50,000 rows are typically updated. The CURR_ACCT_DIM table has one unique B*Tree index supporting the primary key, a column named ACCT_KEY, and 30 single-column bitmap indexes. When the MERGE statement shown above was tested against this table, it took about 8 hours to complete, which is about three times too long.

What can be done?

Assuming that there is an external table (based on a “flat” file) named EXT_ACCT_DIM with the 100,000 rows of data to be up-sorted or merged into the CURR_ACCT_DIM table. Here is how we do it...

1. Range-partition the CURR_ACCT_DIM table with one (and only one) partition (called PDUMMY)
   • The intent of doing this is only to use EXCHANGE-PARTITION during data loading, and not for partition-pruning during queries
2. Create a non-partitioned “scratch” table (named SCRATCH_CURR_ACCT_DIM) which is “shaped” exactly like CURR_ACCT_DIM...

```sql
create table SCRATCH_CURR_ACCT_DIM
parallel nologging compress as
    select acct_key, eff_dt, attr_01, attr_02, ..., attr_99 from
    (select acct_key, eff_dt, attr_01, attr_02, ..., attr_99, row_number() over
    (partition by acct_key order by eff_dt desc) rn
    from
    (select acct_key, eff_dt, attr_01, attr_02, ..., attr_99
    from CURR_ACCT_DIM union all
    select acct_key, eff_dt, attr_01, attr_02, ..., attr_99
    from EXT_ACCT_DIM ) ) where rn = 1;
```

• The inner-most sub-query is a UNION-ALL which queries all 100,000 of the rows from the EXT_ACCT_DIM and the existing contents of the CURR_ACCT_DIM. If CURR_ACCT_DIM previously had 10m rows already, then the output from this inner-most sub-query should be 10.1m rows (i.e. 10m plus 100k rows)

• The middle sub-query exists only to apply the ROW_NUMBER() window function to the results of the inner-most sub-query, so that we can easily identify the latest EFF_DT value for each ACCT_KEY value

• The outer-most sub-query filters out any rows except for the ones with the latest EFF_DT for each ACCT_KEY value (i.e. “RN = 1”) and inserts the remaining rows into the SCRATCH_CURR_ACCT_DIM table

3. Gather table- and column-level statistics on SCRATCH_CURR_ACCT_DIM

4. Create all LOCAL indexes that exist on CURR_ACCT_DIM onto SCRATCH_CURR_ACCT_DIM with the COMPUTE STATISTICS clause

5. Finally, use “alter table CURR_ACCT_DIM exchange partition PDUMMY with table SCRATCH_CURR_ACCT_DIM” to publish the new contents of dimension table for end-users

So in this way, the UPDATE/MERGE was performed with only INSERT/SELECT statements, and the resulting table can be compressed using Oracle9i “basic” compression or, if on HCC-enabled storage, compressed using one of the HCC methods instead, thanks to the user of direct-path APPEND inserts. If need be, the NOLLOGGING attribute can be set on the “scratch” table so that the direct-path APPEND loads don’t generate redo, with the normal caveats and cautions when doing so.

Sometimes the fastest way to Denver is through Kremmling. And sometimes the fastest DELETE or UPDATE is an INSERT. Trust me.

Have fun!

Tim

---

Monish Sharma, continued from page 11

As we can see, the perceived impact by the network engineer for the ticket 14356 was “Low,” implying a localized impact, but DECIPHER scored it with a predicted impact of “High” and with high probability. Also, the IEIM scoring shows the sales_recos_db and hang are dependent with the factors in the 14536 ticket. This shows that if the 14536 ticket were to be implemented then there is a high probability of a potential hang with connections for the sales_recos_db on port 1521. Having this knowledge ahead of time would allow the DBA to mitigate the potential risk of connection hangs by either enabling a different port or enabling additional connection tracing or even delay this change until it is thoroughly tested in the QA environment.

**Conclusion**

“It is not the strongest of the species that survives, nor the most intelligent that survives. It is the one that is the most adaptable to change” – Charles Darwin

This article addresses the important issue of the impact of environmental changes on the performance of databases. Acquiring the environmental change impact knowledge manually is a challenging task. The DECIPHER framework discussed in this article addresses this issue so that DBAs can make better decisions and adapt to the constantly changing IT environments. Today’s databases are designed to be very adaptive and fairly autonomic when it comes to changes in workloads but are not designed to anticipate and handle the organization-specific environmental changes. In order for the databases to be truly autonomic or self-tuning, it needs to acquire this environmental change impact knowledge and adapt to it.

Monish Sharma is currently working as the Sr. Director of Engineering at Epsilon Corporation managing database and business intelligence teams. He has close to 17 years of experience with Oracle technologies in various roles like data modeler, database developer, architect and database administrator. Currently, he is working on the design and implementation of database private clouds using Oracle VM (x86) and Oracle UEK.
Many of you know how I just love Colorado. I purchased a motor home five years ago just to visit Colorado in the Summers. Blue bug is the VW that always goes on these trips. Those who have sat in on my RMOUG Presentations know about my two current VW bugs.

This trip in July was to the Durango area. We took the train to Silverton and back. Honestly, you only need to take it one way as there is a bus that will take you the other way...but I’m into trains and didn’t mind maximizing my time on the thing.

It’s about a three and a half hour train ride to Silverton. Beautiful scenery chugging along at about 25 mph. The train uses 10,000 gallons of water to make the trip and it stopped twice to take on more water.
RMOUG Member Focus
We also visited Mesa Verde (green table in Spanish...).

This is the Mesa Verde. The entrance to the park is some 15 minutes from Durango. The park is huge with some famous Pueblo cliff dwellings ruins that were occupied for some 75 years. Other ruins on top of the Mesa were occupied much longer. The park contains over 4,500 archeological sites (mesa top sites), with 600 or so actual cliff dwellings. Archeologists called these people Anasazi. The park calls them Ancestral Puebloans.

This is the Spruce Tree House Cliff Dwelling. Cowboys discovered these ruins in the 1880’s. The park was created in 1906 to protect the heritage here. Looking at the doorways and the size of the room, my thought is the size of the people were probably shorter than me (5’6”).
This is the Cliff Palace, the largest of the cliff dwellings. The park offers guided tours thru this and many of the other sites.

I did not go see it but the Balcony House has been partially restored to show what it would have been like when people lived there. All other ruins have just been cleaned up.

The Far View Terrace has several small but connected mesa top settlement ruins. These communities were occupied much longer than the cliff dwellings. The posters had a lot of cool write-ups indicating how life was in these communities, what the social order was, how farming and other business was handled, etc. I hiked to several of these.
Reach A Targeted Oracle Audience
Advertise Now!

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

RMOUG SQL>Update Advertising Rates

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

Discounts available for RMOUG Members and Full Year Contracts

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

SQL>UPDATE is mailed to all RMOUG Members
and distributed during Quarterly Education
Workshops and Training Days each year
RMOUG Board of Directors

Meet Your Board

Tim Gorman
President & Vendors Director
E-mail: President@rmoug.org

Kelly Pot’vin
Training Days Director
E-mail: TrainingDaysDir@rmoug.org

David Fitzjarrell
Social Media Director
E-mail: TrainingDaysDir@rmoug.org

Kathy Robb
Board Member Emeritus
Arisant, LLC
E-mail: BoardME@rmoug.org

Ron Bich
Secretary & SIGS Director
E-mail: SIGSDir@rmoug.org

Thomas Green
Scholarship Director
E-mail: ScholarshipDir@rmoug.org

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

Art Marshall
Web Director
E-mail: ISDir@rmoug.org

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

Pat Van Buskirk
Treasurer & Newsletter Director
E-mail: NewsletterDir@rmoug.org
RMOUG Events Calendar

9/25/12  Board  RMOUG: Board of Directors Meeting, Location TBD
9/30-10/04/12  Open World  Oracle Open World 2012, Moscone Convention Center, San Francisco CA
10/15/12  Training Days  RMOUG: Deadline For Submitting Abstracts for Training Days 2013
10/15/12  Newsletter  RMOUG: Call for Articles & Photos, Winter Issue of SQL>Update
10/16/12  Board  RMOUG: Board of Directors Meeting, Location TBD
10/17/12  Conference  UTOUG: Utah Oracle Users Group Business Intelligence SIG, West Valley City, UT
11/15/12  Newsletter  RMOUG: Deadline for Articles & Photos, Winter Issue of SQL>Update
11/16/12  QEW  RMOUG: Autumn Quarterly Educational Workshop (QEW), Oracle in South DTC
12/15/12  Newsletter  RMOUG: Publication, Winter Issue of SQL>Update
1/9/13  Training Days  RMOUG: Final Day for Low Cost Advance/Early-Bird Registration, Training Days 2013
2/6/13  Training Days  RMOUG: Final Day for Standard Registration, Training Days 2013
2/11-13/13  Training Days  RMOUG: Training Days 2013, Colorado Convention Center, Denver CO
3/3/13  Symposium  HOTSOS Symposium 2013, Omni Mandalay, Irving, TX
4/7-11/13  OAUG  Collaborate (OAUG, Quest & IOUG) Conference, Colorado Convention Center, Denver, CO

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

<table>
<thead>
<tr>
<th>Breakfast</th>
<th>Discounted Ad Rate</th>
<th>Total Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/4 Page</td>
<td>$350.00</td>
<td>$175.00</td>
</tr>
<tr>
<td>1/2 Page</td>
<td>$350.00</td>
<td>$312.50</td>
</tr>
<tr>
<td>Full Page</td>
<td>$350.00</td>
<td>$500.00</td>
</tr>
<tr>
<td>Inside Cover</td>
<td>$350.00</td>
<td>$625.00</td>
</tr>
<tr>
<td>Back Cover</td>
<td>$350.00</td>
<td>$750.00</td>
</tr>
</tbody>
</table>

Contact Carolyn Fryc - Programs Director - 720-221-4432 - cfryc@orsportal.com

Bring Your Horses & Guests
Franktown, CO $640,000
6.88 Acres • In-Ground Pool • Huge Shop
• 4282 Square Feet
• Fully Finished Basement
• Wet Bar, Decks
• Apartment Above Shop
• Bonus Room Above Garage
• Two Fireplaces + WB Stove
• Pasture With Loafing Shed

Entertainment Splendor
In The Pines
Pat Van Buskirk • Coldwell Banker • Parker Office
(303) 243-0737 Cell (303) 841-5263 Office
www.patvanbuskirk.com pat@patvanbuskirk.com
Reach A Targeted Oracle Audience Advertise Now!

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

Contact NewsletterDir@rmoug.org

Quarterly Education Workshops

November 16, 2012 • Oracle DTC

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

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

Carolyn Fryc, Programs Director at ProgramsDir@rmoug.org

Watch RMOUG’s Web Page for November Training Topics
SUBMIT YOUR ABSTRACT BY
OCTOBER 15, 2012

RMOUG TRAINING DAYS
February 11-13, 2013  Denver, Colorado
Colorado Convention Center
www.rmouug.org

CALL FOR PAPERS OPEN NOW - TOPICS

- Application Development
- Business Intelligence
- Database Administration
- DBA Deep Dive
- Database Tools of the Trade
- Middleware