The Bean Consulting Group
The Challenges of Hyperion Support - Part 3

Megan Feldman Bettencourt
Compassion For Self and Others

Karen Cannell
Apex 5 Interactive Reports

Sumeet Kabra
E-Business Suite 12.2

Robert T. Mason
NoSQL Databases and Data Modeling

Board Focus - Susan Llanes
Rolta AdvizeX brings you a complete solution for all your application, data, and infrastructure needs. Our customers use technology to drive innovation and achieve their most important business goals because we first help them focus on a strategy that leads to success, then bring the data, applications and infrastructure expertise needed to create and sustain that success.

To explore how we can help your organization, contact Rolta AdvizeX at 800-220-8454 or visit Rolta AdvizeX online at www.advizex.com

#CloneAttack
What’s in it for you?

Going to RMOUG Training Days 2015? Try Delphix #CloneAttack free for 90 days. Install your own ready-to-use virtual machines and get hands-on experience with this important new capability.

Delphix experts will be ready to assist, demonstrate, or answer your questions in the lab at the rear of the exhibitor hall. Find out more about #CloneAttack on Twitter or contact us at cloneattack@delphix.com

#CloneAttack, Wed-Thu, February 18-19, 2015 at RMOUG Training Days 2015 in Denver, CO.
ON THE COVER: Cover photo by Vincent Giasolli. This photo was taken at Coyote Ridge Park in Castle Pines on June 1. You can often catch Vincent exploring Colorado in his 2002 Ford Mustang convertible.
Hello RMOUG,

It is the start of a new season for the Rocky Mountain Oracle Users Group.

A quick look back, 2014-2015 was a good year for RMOUG, several good events, such as Quarterly Education Workshops and another successful Training Days Conference last February.

RMOUG has been blessed for the last 18 years by the contributions of Heidi Kuhn, we look back at all she has done and all Heidi has accomplished, taking care of us RMOUG members and the User group activities and business interest. The term, the glue that holds a group together, is a term meaning Heidi has been so important and so appreciated and a fantastic person. Good news for Heidi is she is moving on to other great things to accomplish, we wish her well and say a thousand thanks for being a part of the RMOUG family for so long. Best wishes.

RMOUG is excited to announce that Peggy King has agreed to help us out with all of the mountain of tasks that Heidi has been doing in the past. Peggy has a tremendous depth of knowledge of our user group having served in roles such as President and Training Days Director over the years. Please welcome Peggy to this new role.

The New Board of Directors For 2015-2016.

Here is some information on those folks that have stepped up to make a difference in our IT Oracle and related products and charitable activities.

Sruthi Kumar Annamnidu – Voting Member

Campaign Statement

As a member of RMOUG for the past few years, I have served on the board of directors as a “Member at Large”, contributed to newsletter, actively volunteered at many events and board meetings. I have introduced RMOUG to educational institutions like University of Denver, and also helped RMOUG make more connections with Denver SQL Server User Group. Recently, I have focused on Special Interest Group (SIG) and started ‘RMOUG Big Data SIG Meetup’ group

Rene Antunez Member-at-Large, Biography:

My current position is as a Team Lead at The Pythian Group, and my company is able and committed to offer me great support to be involved in regional user groups and SIGs, and as an Oracle ACE, I'm able to help people see the value of RMOUG as valuable regional user group during presentations and webinars.

Chris Chase - Voting Member, Biography:

Chris Chase is a managing partner with The Bean Group.

Hyperion Planning and Essbase professional with over 20 years of experience designing, developing, and implementing corporate IT solutions using OLAP and OLTP databases and object oriented programming languages

Strong presentation skills ranging from technical to analytical presentations given to all levels including C-Level executives

Self-motivated problem solver with strong analytical skills and an ability to develop and maintain a high level vision, while also working hands-on to complete detailed, complex tasks

Proven team leader with strong managerial skills and an ability to quickly master new technologies and understand how they fit into a corporate technology strategy

Bobby Curtis – Voting Member

Campaign Statement, Qualifications and Biography:

My position as an Infrastructure Principle with Accenture Enkitec Group, offers me incredible support from my company to be involved in my local and remote user groups. I hope to be a positive resource within the RMOUG board of directors and of assistance to other board members through collaboration and skills that will continue to make RMOUG the successful Oracle User Group it is today.

Karen Funston – Voting Member, Biography:

As the Director, Business Intelligence for the Colorado Community College System, I have built a department from a three-person group to a seven-person team in 4 years. We have gone from having a skeptical user community to being declared by executive staff as “critical for business continuity” in that same period. I have served as a program committee member on an annual, national conference for higher ed (Ellucian Live – formerly Summit) for the last 8 years. I have been fortunate to be able to help build the Business Intelligence community to the point that we now are recognized as a viable group. I was privileged to receive the 2015 Outstanding Contributor Award for Excellence in Community Building and Support of Peers and Colleagues from Ellucian – a leading software provider to higher education.

Vincent Giasolli – Voting Member, Biography:

Vincent Giasolli has been using Oracle and the Oracle E-Business Suite since 1992. He is currently working in downtown Denver in the Oil and Gas industry. He has been employed across industry segments including Manufacturing, Government, Financial, Medical, and Real Estate. as a DBA, Developer, Data
Tim Gorman – Voting Member, Biography:
Tim Gorman began his IT career in 1984 as a systems programmer on UNIX and VMS operating systems. He joined Oracle in 1990 and became a technical manager in professional services. In 1998 he founded a consultancy focusing on Oracle performance, then joined Delphix as a consultant in 2014. Mr. Gorman is co-author of six books on data warehousing and database administration for Oracle from John Wiley and Apress, and has served as technical reviewer for eight other books for Oracle from O’Reilly & Associates and Apress. He has published articles for RMOUG “SQL>Update”, for IOUG “Select” magazine, and for the NoCOUG Journal. He is invited to present at 10-12 Oracle conferences annually in the Americas, Europe, and Asia/Pacific. Mr. Gorman is recognized by Oracle as an ACE Director and by his peers as a member of the Oak Table network.

Komal Goyal – Member-at-Large
I am very excited to apply for a position as a Board of Director for RMOUG especially to help the Quarterly QEWS program. Although this will be my first time on the board, I would like the opportunity to serve on this position and help the Oracle User Group community in the Rocky Mountain region. As a programmer, technical lead and project manager my experience with Oracle products as a consultant and user will help me bring a different perspective to the board.

As a technology consultant with more than twenty years progressive experience in establishing and executing value driven global IT service strategy and delivery will help me as board member to bring innovative ideas to the programs run by RMOUG. Personally I am passionate about creating talent pipeline for the growing need in the technology world. I believe being part of such an instrumental User Group will help me implement the business ideas with in this program to attract the schools in the area to start attracting the regional talent to the program. I also have consulting experience with Oracle Enterprise Systems and technologies within commercial and public sectors, that will add another angle from which I should be able to contribute to the board.

While volunteering during Training days 2015 and being on the Women in Technology panel discussion I was able to see the impact RMOUG makes in this region. I believe in RMOUG’s mission and hope to get an opportunity to contribute.

Dan Hotka – Voting Member, Biography:
Dan Hotka is a training specialist and an Oracle ACE Director who has over 32 years experience in the computer industry, over 27 years of experience with Oracle products. His experience with the Oracle database dates back to the v4.0 days.

Mark James – Voting Member, Biography:
Mark James has over 16 years of IT Experience, mainly as an Oracle Database Administrator, while also being highly skilled on numerous other major database vendor platforms as well. He has been an OCP since Oracle 8i through all versions up to 11g. He holds a BS in Accounting (1993), an MBA in Finance and Accounting (1997) and an MScIS Degree in Database Technologies (2000). He taught Master’s Level Database Technologies Courses for almost 10 years at Regis University and has been teaching Master’s Level Database Technologies Courses for the University of Denver (DU) for over 9 years. He has worked as a DBA for several major companies in diverse business lines such as Internet Services, Telecom, Finance, Healthcare, Defense, Automotive, Energy, Oil and Gas.

John Jeunnette – Voting Member, Biography
I have worked as an application architect and developer in a variety of industries ranging from government agencies to transit companies to pharmaceutical companies to non-profit organizations for longer than I care to admit. Currently I am Oracle Applications Administrator for Dartmouth College supporting an Oracle E-Business Suite installation.

User group activities include volunteering at RMOUG Training Days, ODTUG conferences, and International Oracle User Group Community (IOUC) activities ranging from registration volunteer to conference chair to officer. Along the way I created the Conference Abstract Administration Tool used by over fifty conferences worldwide to process presentation submissions and registration, and the Group Administrator membership and event application.

Andreas Katsaris – Voting Member
Biographical Information:
My extensive background in IT (19 years) includes technical positions as a C++ developer, Oracle DBA, technical manager, data warehouse architect and OBIEE/ODI developer/architect in a variety of industries and environments. For the last 8 years I have been part of Arisant, LLC and as a co-founder and senior partner I have had the opportunity to grow my experience in the areas of Product Management, Account Management, marketing and business development. Over the years I have had the opportunity to work with many customers in the US as well as several countries in Europe and gain invaluable experiences.

I have been attending and presenting at RMOUG for several years (in a very non-regular and not predictable manner, I am afraid to say!) and feel that my time has finally come, to offer my time and expertise to RMOUG.

I believe my unique combination of technology and management skills will be invaluable to RMOUG.

Susan Llanes – Voting Member
Biography and Qualifications:
Susan has approximately 15 years in global business development and marketing strategy in technology. She currently leads this effort at 6e Technologies, an Oracle Gold Partner. She specializes in maintaining and developing client relationships for Oracle solutions in EBS, SOA, BI, Cloud and Application Testing Suite.

Art Marshall – Voting Member, Biography:
I have worked in the IT industry since 1980, having used various database systems from then on, including my first exposure to Oracle in 1991. My Oracle responsibilities started with platform migration, then query and Forms development, and then those of a database administrator since 1997, in both production and development environments, including those of non-clustered, Data Guard, and RAC. Versions that I have been involved with include 6, through 12. With this long-term involvement in IT, I have a strong commitment to continually expand my knowledge of important Oracle database technologies, and to facilitate the usage and sharing of that knowledge with others.
APEX 5 Interactive Reports Deep Dive: New Features and Upgrade Cheat Sheet - Part 1

by Karen Cannell, TH Technology

Abstract
APEX Interactive Reports have undergone major changes in APEX 5.0, bringing big new features and significant changes behind the scenes, changes that impact all interactive reports. Understanding these changes is essential for any developer who has enhanced or customized interactive reports and is planning on upgrading to APEX 5. This paper, Part 1, introduces the APEX 5 Interactive Report (IR) new features. Part 2, takes a deep dive into the structural and code changes behind the scenes that make these improvements possible.

Target Audience
All APEX developers should be aware of the APEX 5 new features in order to pass these features on to their users and to leverage new development efficiencies. In addition, all developers who have customized interactive report appearance, altered Actions, written IR dynamic actions or otherwise enhanced an IR in any earlier APEX version (most of us?) need to know the behind-the-scenes details of APEX 5.0 IRs. The changes are significant, and unless the APEX standard APIs have been used the customization may not upgrade smoothly.

Executive Summary
The purpose of this paper is to help readers learn about APEX 5 IRs, in two parts:

- Part 1 - APEX 5 Interactive Report new and enhanced Action Menu and support features.
- Part 2 - APEX 5 Interactive Report CSS and JavaScript changes, and how these changes impact existing APEX Interactive Reports.

The Appendix of this paper includes an APEX 5 IR Upgrade Cheat Sheet to assist in mapping pre-APEX 5.0 Interactive Report CSS and JavaScript elements to their APEX 5.0 equivalents.

Background
Introduced in APEX 3.2, IRs have become the default report option for APEX applications. IRs enable developers to deliver a wealth of end user functionality with a minimum of effort. To build an IR the developer enters a few keystrokes, a base SQL query, some optional declarative settings and voila - the end users gets a fully-operational interactive set of data from which they can apply filters, rearrange columns, highlight, sort, chart, group by and otherwise slice and dice their data.

Note: Most developers and end users are familiar with the search, filter, sort and other interactive features. The paper assumes readers are familiar with pre-APEX 5 IR’s. For those who are not, a full description can be found in this Oracle Technet (OTN) introduction to APEX IRs: http://www.oracle.com/technetwork/testcontent/irrs-083031.html

Every APEX release since 3.2 has added more features, most noticeable in the Action Menu. In addition, many developers have added customizations via CSS to improve appearance, dynamic actions to mimic MS Excel behavior, and otherwise extended standard IR functionality. Developers have made IR customizations through dynamic actions, plugins, JavaScript frameworks and other bits of code to effect the desired changes. In general, we APEX developers can be quite creative in delivering enhanced functionality to our end users!

APEX IR
As usual for a new APEX release, APEX 5 introduces new IR features: new and enhanced Action Menu features, some cosmetic uplifts and some report management improvements. However, with APEX 5, the major IR changes are behind the scenes – APEX IRs have been rebuilt from the inside out. Both the “inside” changes – the JavaScript engine – and the “outside” changes- CSS classes and Ids - are significant. The important note for developers is that because of this re-architecting, even when developers used the APEX-standard dynamic action or plugin frameworks, if the customization code references the pre-APEX id’s and class elements, the customization code will need to be refactored to upgrade.

Why so many changes, and why such a drastic change? Several reasons. The revised IR code allows for:

- Multiple IRs on one page (perhaps the biggest new feature, to be described in detail in later sections of this paper)
- Modal dialogs
- The Universal Theme and Theme Roller customizations
- APEX overall usability and accessibility

These features, particularly multiple IRs on a single page, were just not possible with the pre-APEX IR architecture. The new code makes sense, when one considers that multiple IRs on a single page was not possible with the legacy IR structure – something had to change. With all of the other IDE and end-user interface changes in APEX 5, the IR changes make sense.

The downside is, developers who have tooled outside of the APEX sandbox must now invest some time in upgrading to the new IR structure and JavaScript. All IR customizations made outside of
the standard APIs may not work in APEX 5. Developers who have made such changes will need to refactor their customizations.

The following sections discuss APEX 5 IRs in detail. The Know Your Users section applies to all developers. The Page Designer section gives an overview of the Page Designer, the new IDE for building APEX apps. The New Features section reviews key new features that developers and end users can leverage upon upgrade. In Part 2, the Changes section addresses the CSS and JavaScript changes such developers need to know to plan their upgrade to APEX 5.

Know Your Users

IRs are very powerful in that they deliver a lot of end user functionality with minimal developer effort. However, it is the developer’s responsibility to maximize IR effectiveness by using the declarative settings to tailor the IR to end user needs. This means the developer needs to be aware of such things as overall security needs, how user use the data set, which Action Menu features should be prohibited or restricted, how much training end users will require, which download options are required, and how users are likely to use Saved reports. It is the developer’s job to prepare and deliver the appropriate IR query and action set to support them.

In short, Know Your Users. Watch what they do, because what they really do is not necessarily what they say the do or need. Deliver the functionality they need, restrict the features they should not have, and ensure the data set they receive is useful to them.

Page Designer

Perhaps the most impressive new feature of APEX 5 is the Page Designer, an alternative to the Component View and Tree View modes of APEX development.

The Page Designer is a totally redesigned development environment for building APEX. It allows the developer to stay in “edit mode” in one window while viewing the results of changes in a separate window. This eliminates opening and closing edit dialogs – the Page Designer is always open. This new APEX IDE drastically changes the flow of how one develops in APEX, and in most cases streamlines the process. Before addressing APEX 5 IR new features, it is important to be a bit familiar the Page Designer. Figure 1 illustrates the APEX 5 Page Designer.

The key features are:
  • Left Pane – The Tree pane. This pane contains four views,
  Rendering, Dynamic Actions, Processing and Shared Components. Each view contains a tree structure of the components of that category on the page, and allows for selection of components and Attributes of those components. Selections here open the corresponding property edit dialogs in the right-pane Property Editor region.
  • Center Pane – The Grid Layout pane. This pane is a visual representation of the relative location of components on the page. One can rearrange components on the page by drag and drop of selected elements in this region. One can add components by dragging from the component Gallery at the bottom of the page and dropping onto the layout grid.
  • Right Pane – The Property Editor. This is where one edits properties of the components selected in the left, Rendering pane. The same attributes for each component type exists as in earlier APEX versions, this is simply a new interface for editing them.

For a better description of the Page Designer, see the APEX 5 Application Builder User's Guide, section 2.2.2.1. This is accessible from the Help menu of the APEX 5 Early Adopter.

It is easiest to understand and learn the Page Designer by diving in and working with it. While the Page Designer is a totally new development interface, this author found the transition quite smooth. The most difficult aspect was finding attribute locations in the Property Editor. To assist in this I developed the habit of using the Close All button to reduce the Property Editor to simple list, from which it is easier to find the section of interest.

The other change is operational - one makes changes in the Page Designer, then clicks Save, then Run. Run opens a version of the page in a separate window (or in a separate tab, depending on one's browser settings. Thus a developer can have the Page Designer, the edit interface, open in one window and the current version of the app open in a different window.
The recommendation is to spend a few minutes in the Page Designer to become familiar with its operation. Then spend a few moments specifically on an IR. Locate the Columns and Attributes in the Tree panel, then locate the corresponding Property Editors in the right panel. Don’t forget to Save (which the Page Designer will remind you of) and Run, both buttons located at the top right of the page. Also, become familiar with jumping from the Page Designer to your application is a separate page or window.

Note: Developers used to achieve as similar simultaneous builder-app window-mode by running the APEX Builder in one browser (i.e. Chrome) page, and running the application in a different browser (i.e. Firefox) page. There are other ways to achieve the same with virtual server names. We no longer have to do this. In fact, when we do, we then have two Page Designer windows and two app windows – this author found it gets confusing quickly. We still need to test in multiple browsers, but it is no longer essential to use multiple browsers or other tricks to have simultaneous develop and run windows. A big plus!

New Features
APEX 5 introduces a new IR look-and-feel, improved usability and accessibility features, new and enhanced Action Menu features. The following sections detail the most notable new features that developers and end users should be aware of.

Modernized Interface
The IR user interface has been updated to be “cleaner” and have “more semantic HTML markup”. The cosmetic changes are in line with the new Universal Theme. APEX 5 IRs are responsive, accessible, and easily customized via Template Options and Theme Roller. All popup windows are now modal. All icons are now scalable vector icons. The colors of the icons and popups will follow that of the theme in use.

Column heading dialogs, shown in Figure 4, are now activated by a click. The edit dialogs of the Control Panel are now improved for easier - fewer clicks – edits. All of these cosmetic and usability improvements are apparent and appreciated after a few minutes of working with an APEX 5 IR.

Floating Headers
At last, APEX IRs have declarative “floating headers”, a feature much requested by Microsoft Excel (MS Excel) users. This new feature allows a developer to declare that column headings stay fixed at the top of the Page the top of the Region, or scroll with the data (None) as the user scrolls down the page. Figure 5 illustrates fixed headers – fixed to the page – in action (as best one can in a static page). The headers for both IRs stick to the top of the page as the user scrolls through the IR rows. Figure 6 shows the Fixed To attribute options in the Property Editor, used to set the Fixed To column heading behavior.

Alas, the corresponding Fixed Column feature, another favorite of MS Excel users, is not part of APEX 5.

Enhanced GROUP BY
The GROUP BY action now allows up to 8 Group By columns. This is 3 more Group By columns that APEX 4.2. The Group By modal dialog is shown in Figure 7. Figure 8 shows access to edit an existing Group By pane and the Group By button to toggle between the Group By pane and other IR views (in this example the main tabular view and a Pivot view).

As with other optional report views, formatting options on the Group By view are restricted, in this case to Group By and Group By Sort. Figure 9 shows the limited set of format action for a Pivot report. As always, the Filter (shown), Rows per Page, Flashback, Save Report, Reset, Help and Download options, if declared and authorized, are still available in the Group By report view.
As with other optional report views, formatting options on the Pivot view are restricted, in this case to Pivot and Pivot Sort. Figure 12 shows the limited set of format actions for a Pivot report. As always, the Filter, Rows per Page, Flashback, Save Report, Reset, Help and Download options, if declared and authorized, are still available in the Pivot view.

Multiple IRs

Developers may now build “any number” of IRs on one page. This single feature is one of the reasons for the changes in how IR elements are named - it was essential to do so, so that each IR element is clearly identified. To create multiple IRs on one page, simply create another one using any of the Create Region options. The Create IR wizard used to block this operation – that block has been removed. Create away!

With the improved editing capabilities of the Page Designer, it is easier than ever to create multiple IRs on a single page, as it is very simple to copy (Duplicate) and move regions around on the page.

To copy an IR on the same page, from the Rendering view in the Tree pane, right-click the IR region to copy and then select the Duplicate option. If the new, duplicate region was not created where you want it, select the IR in the Grid Layout pane and start to move it. The objects in the Grid Layout pane will become highlighted in yellow – this indicates you are in the drag-and-drop mode that allows you to drag the selected region to a new location. Drag the selected region around to the desired location and let go.

Hint: You may have to go slow, and be patient with getting the drag-and-drop layout to open up the location you want.

Figure 13 shows selection of the Duplicate option from the Rendering pane.

Linking

Allowing multiple IRs on one page mandates changes in how filter parameters are passed to an IR in links between pages. The traditional IR filter options of IREQ_<COLUMN_NAME>, etc. shown in Table 1 work for a single IR on a page and will continue to work for the case of a single IR on a page. Applications
with links to pages that contain one IR will continue to work upon upgrading to APEX 5. When there are multiple IRs on a page, however, this link syntax changes.

**Figure 13 – Duplicate Region Option from the Rendering Pane**

![Image of duplicate region option from the rendering pane]

**APEX Multiple IR Link Qualifiers**

<table>
<thead>
<tr>
<th>Multiple IR Link Qualifier</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>IREQ[STATIC_ID]_&lt;column_alias&gt;</td>
<td>Equals</td>
</tr>
<tr>
<td>IREQLT[STATIC_ID]_&lt;column_alias&gt;</td>
<td>&lt;</td>
</tr>
<tr>
<td>IRLTE[STATIC_ID]_&lt;column_alias&gt;</td>
<td>&lt;=</td>
</tr>
<tr>
<td>IRGTE[STATIC_ID]_&lt;column_alias&gt;</td>
<td>&gt;=</td>
</tr>
<tr>
<td>IRLIKE[STATIC_ID]_&lt;column_alias&gt;</td>
<td>LIKE</td>
</tr>
<tr>
<td>IRN[STATIC_ID]_&lt;column_alias&gt;</td>
<td>NULL</td>
</tr>
<tr>
<td>IRNN[STATIC_ID]_&lt;column_alias&gt;</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>IRC[STATIC_ID]_&lt;column_alias&gt;</td>
<td>Contains</td>
</tr>
<tr>
<td>IRNC[STATIC_ID]_&lt;column_alias&gt;</td>
<td>NOT Contains</td>
</tr>
</tbody>
</table>

Note, no BETWEEN, IN and NOT IN options

**Table 1 – Single IR Link Filter Syntax**

<table>
<thead>
<tr>
<th>Single IR Link Qualifier</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>IREQ_&lt;column_alias&gt;</td>
<td>Equals</td>
</tr>
<tr>
<td>IR_&lt;column_alias&gt;</td>
<td>Equals</td>
</tr>
<tr>
<td>IRLT_&lt;column_alias&gt;</td>
<td>&lt;</td>
</tr>
<tr>
<td>IRLTE_&lt;column_alias&gt;</td>
<td>&lt;=</td>
</tr>
<tr>
<td>IRGT_&lt;column_alias&gt;</td>
<td>&gt;=</td>
</tr>
<tr>
<td>IRLIKE_&lt;column_alias&gt;</td>
<td>LIKE</td>
</tr>
<tr>
<td>IRN_&lt;column_alias&gt;</td>
<td>NULL</td>
</tr>
<tr>
<td>IRNN_&lt;column_alias&gt;</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>IRC_&lt;column_alias&gt;</td>
<td>Contains</td>
</tr>
<tr>
<td>IRNC_&lt;column_alias&gt;</td>
<td>NOT Contains</td>
</tr>
</tbody>
</table>

Note, no BETWEEN, IN and NOT IN options

With multiple IRs on a page, the link syntax must distinguish which IR to operate on. To do so, the APEX 5 link syntax for cases of multiple-IRs incorporates the Static Id of the IR, in square brackets, just after the link qualifier:

IREQ[STATIC_ID] <COLUMN_NAME>

The full list is the same as for single IRs, as shown in Table 2.

To facilitate passing parameters, it is recommended that the developer declare a Static Id for the IR. If a Static Id is not declared, one will be generated for you in the format R123456789012345 where 1234567890125 is the assigned static identifier. For this author, it is much easier to remember and use a declared Static Id (one that I define) than to use Inspect Element to dig out the APEX-supplied region identifier. It is simpler and more meaningful to pass references of the format IREQ[EMPS]_DEPTNO, for example, than write references such as IREQ[R1132354054905]_DEPTNO.

**Why Multiple IRs?**

One may ask, what is the value of multiple IRs on the same page? The use cases are as many and varied as there are APEX installations. The fact is, users are accustomed to search and filter operations on the report, they want and even expect those features on all of their reports. It is likely when there are multiple IRs on one page the span of actions allowed in each IR will be limited to those that make sense for the particular data set.

It is the developer’s responsibility to design the page and the IRs on it such that the user has exactly the functionality they need, and is not overwhelmed with options.

**Possibilities**

Multiple IRs on one page opens up several options not previously possible. Consider building a dashboard of reports, all IRs so that each data set may be configured with chart, group by and perhaps pivot views. Imagine driving the contents of all the IRs with a click on the value of one IR, say, the one in the top left position. Figure 14 illustrates such a scenario.

**Figure 14 – Multiple IR “Dashboard”**

![Image of multiple IR dashboard]

This scenario can be constructed by building multiple IRs on one page, then enabling a link on the Dept column of the top left IR
to the DEPTNO columns of the remaining IRs on the page. To do this, edit the Link attribute of the DEPT column of the top left IR, as in Figure 15. The Link attributes region may not readily appear in the column property editor. If it does not, ensure that the column type (at the top of the Attributes panel) is selected as Link. Then the Link attributes region should appear as shown in Figure 15.

![Image](Figure 15 – IR Column Link Property Editor)

The Link attributes region may not readily appear in the column property editor. If it does not, ensure that the column type (at the top of the Attributes panel) is selected as Link. Then the Link attributes region should appear as shown in Figure 16.

![Image](Figure 16 – Component View Icon from Page Designer)

An alternative way to set the Link attributes is to revert to the Component Editor, add the column link attributes there, then return to the Page Designer. The column Link attribute section will be there. The Component View can be reached by click on the Component View icon, shown in Figure 16, in the Page Designer top menu.

To return to the Page Designer from Component View use the corresponding Page Designer icon in the top menu bar, as shown in Figure 17.

![Image](Figure 17 – Return to Page Designer from Component View)

This same approach of switching to the Component View to perform certain settings, then switching back to Page Designer, is always available and can be helpful during a developer’s transition to to being fully comfortable in Page Designer.

### Clear and Reset

As with all links to IRs, it is often desirable or necessary to reset the IR to its original or default settings before passing in the next link. There are two options to do this, passed in the Cache position of the APEX URL:

- **RIR** – Resets the IR to the default settings. This is the most frequently used option.
- **CIR** – Clears all IR settings, regardless of the defaults.

A Reset after a Clear will return the IR to its default settings.

A plain RIR or CIR will reset or clear a single IR on a page. This same syntax will also reset or clear ALL IRs on a page. For example:

```
f?p=100:10:3629280452177::NO::RIR
```

will redirect to page 10 of application 100, no debug, resetting the single IR - or all IRs on the page – to the default settings.

To reset or clear only one IR when there are multiple IRs on a page, the syntax is a bit more complicated. The trick is to specify which IR you focusing on, using the format IR[STATIC ID] in the Request portion of the APEX URL, followed by the RIR or CIR directive in the Cache portion of the URL. For example:

```
f?p=100:10:3629280452177::IR[DEPT],IR[EMP]::RIR
```

Where one IR’s static id is DEPT, and another IRs static id is EMP, this will redirect to page 10 in application 100, reset the IRs with the DEPT and EMP static ids, but will not touch other IRs on the page.

Note that the expected syntax options of [STATIC_ID]RIR and RIR[STATIC_ID] in the Cache portion of the APEX URL do not work. The direction on which IR to operate on must be specified in the Request portion of the APEX URL.

There is always the option to use the APEX_IR API, the CLEAR_REPORT and RESET_REPORT procedures (either signature) in a dynamic action to achieve the desired individual report reset. There is also the option of triggering the APEX-standard `apexrefresh event`, as described in this article: [http://docs.oracle.com/cd/E59726_01/doc.50/e39147/extend_app002.htm#HTMDB30267](http://docs.oracle.com/cd/E59726_01/doc.50/e39147/extend_app002.htm#HTMDB30267)

Using a command like:

```
apex.eventtrigger( "#myRegionStaticId", "apexrefresh fresh" );
```

All of the Reset and Clear options described in this section are supported.

### Part 1 Summary

APEX 5 IRs contain some noteworthy new features for end users and developers. For end users, the most significant of those are the improved look-and-feel, modal dialogs, the enhanced GROUP BY action, the new PIVOT action, and the option for multiple IRs on a single page. For developers, the big new operational change is the Page Designer. This installment covers the highlights of those new and enhanced features.

In Part 2 of this series, we cover the most significant change for developers, the totally revamped APEX 5 IR underlying architecture. Developers who have done customizations on IRs in previous APEX versions may need to refactor their code to follow the new APEX 5 IR structure. With APEX 5, the overall power of APEX IRs has increased, but when it comes to customizations, some upgrade work may be necessary. Stay tuned for Part 2!

### References

- AMIS Pivot Implementation for pre APEX 5 [https://technology.amis.nl/2006/05/24/dynamic-sql-pivoting-stealing-antons-thunder/](https://technology.amis.nl/2006/05/24/dynamic-sql-pivoting-stealing-antons-thunder/)
- Manually Refreshing APEX Components [http://docs.oracle.com/cd/E59726_01/doc.50/e39147/extend_app002.htm#HTMDB30267](http://docs.oracle.com/cd/E59726_01/doc.50/e39147/extend_app002.htm#HTMDB30267)

Please see Karen’s Bio on Page 26
Accounting calendars can be the same as Gregorian calendars, however, they can also differ from Gregorian calendars. For example many companies utilize what is called a 4-4-5 calendar as a method of managing accounting periods; the 4-4-5 calendar divides a year into 4 quarters where each quarter has 13 weeks where each week starts on a Sunday and ends on a Saturday, and where the first four (4) weeks of the quarter constitute the first month of that quarter, the next four (4) weeks constitute the next month of the quarter, and the last five (5) weeks constitute the last month of the quarter. This can also take the form of 5-4-4 or 4-5-4 depending on the organization’s seasonality, industry, and financial reporting requirements; however, the 4-4-5 is the most common arrangement.

The 4-4-5 accounting calendar is a common accounting structure for industries such as retail and manufacturing, and the benefits include the following:

- A consistent 13 week, 91 day quarter to measure and compare quarter over quarter financial and operating results.
- Within manufacturing companies, aligns operational forecasts and production schedules to the accounting period thereby facilitating the planning process. This will be discussed further in the Driver Based Budgeting section below.
- Planning for Payroll becomes easier; assuming that the pay week begins on a Sunday and ends on a Saturday.
- The ability to forecast and plan in standard weekly buckets of data (if forecasting and planning is occurring at a weekly level).
- The end date of the period is always the same day of the week, which is useful for shift or manufacturing planning, and every period is the same length.

The disadvantages of the 4-4-5 accounting calendar are that month-by-month comparisons or trend over periods do not always make sense because one month is 25% larger than the other two in a particular quarter, and the 4-4-5 calendar has 364 days (7 days * 52 weeks), so that approximately every 5.6 years there will be a 53-week year, which can make year-on-year comparison difficult. However, you can still compare a period to the same period in the prior year, or use week by week data comparisons.

So, with that as a backdrop, let’s now discuss each of the finance processes and how Hyperion applications are involved in those processes.
Monthly Close

The monthly close is the processing of all transactions and journal entries, and the creation of financial statements, at the end of each fiscal month. For accounting purposes, it is imperative that the financial statements reflect only the transactions and journal entries having relevance to the current month’s revenues and expenses, and end-of-the-month assets and liability balances. If an organization doesn’t have sufficient controls to ensure this occurs, then the dreaded re-statement is necessary. So, to ensure that the monthly financial statements are accurate and timely, organizations will typically use standard and recurring journal entries, and close process checklists for the tasks that must be completed.

Technology’s Part

Many Enterprise Resource Planning (ERP) solutions can be used in the monthly close process, such as JD Edwards, PeopleSoft, and Oracle EBS to name just a few. As we discussed in Part 2 of our series, Smart View and User Support, the General Ledger (GL) systems allow transactions to be tracked by account and other “segments” with related properties.

Hyperion Financial Management

In conjunction with these ERP tools, Hyperion Financial Management (HFM) is a tool that assists in the financial close process by streamlining the consolidation, conversion, elimination and allocation processes that occur during the monthly close – many of which occur outside of the ERP solution. As such, it is imperative that the ERP general ledger chart of accounts be “configured” to support the requirements of the consolidation tool. As a simple example, inter-company eliminations should be targeted for specific “plug” accounts, and the consolidation tool should be “aware” of this so that eliminations occur at the correct level.

The monthly close process can take anywhere from 1-2 days for organizations that are very efficient, up to 2 weeks for an average organization, and over 3 weeks for other organizations that, let’s just say, are not so efficient. This can be seen in the diagram below.

By allowing rapid consolidations that can be run multiple times in a day, HFM users in the finance group can get a clear picture of what the organization's position will be on a repeated basis which allows them to begin preparing statutory filings such as the quarterly “Q”, and the annual “K”, as well as internal shareholder and management reporting.

The timing of the close process then becomes extremely important when layered with the budgeting and forecasting calendars – particularly when a monthly rolling forecast is used, as we will see. Thus, the faster the monthly close is completed, the sooner the organization has relevant information that that will allow it to compare against budgets, provide guidance, and revise its forecasts as necessary.

The best way to improve the cycle time on the close process is to utilize technology

Close Manager

As Oracle puts it, “Hyperion Financial Close Management is built for centralized, web-based management of period-end close activities across the extended financial close cycle. The first application of its kind, it helps manage all financial close cycle tasks, including ledger and sub-ledger close, data loading and mapping, financial consolidation, account reconciliation, tax/treasury and internal and external reporting processes – any task associated with the extended financial close.”

In short, Close Manager provides a mechanism to ensure that all steps involved in the close process are completed, in the correct order, and with appropriate sign off and approval. In addition, having a tool that provides visibility and traceability to the close process, ensures a much greater chance of SOX compliance.

Budgeting and Forecasting

Budgeting

Businesses plan financially using budgets, typically over a one year time period, where an attempt is made to forecast its revenues and allocate its expenses over the year. The goal is to insure that the business doesn’t spend more than it makes – that is, that it’s profitable. During the “budgeting season” management typically relies on “budget to actual” variance reports which display budgeted amounts contrasted with the actual amounts spent in order to assist in making decisions about the upcoming year.

Some of the more common methods for budgeting are incremental budgeting, zero based budgeting, and driver based budgeting. We will briefly discuss each below, along with when and how they are used, and how the Oracle Hyperion products are used in support of these processes.

Incremental Budgeting

Incremental budgeting is budgeting based on slight changes from the preceding period’s budgeted or actual results. This is a common approach in the public sector and businesses where management does not intend to spend a great deal of time formulating budgets, or where it does not perceive any great need to conduct a thorough re-evaluation of the business from the ground up. This mindset typically occurs when there is not a great deal of competition, and where revenues tend to be perpetuated from year to year.

There are several advantages to incremental budgeting, which are as follows:

• The primary advantage is the simplicity, since it is based on recent financial results or a recent budget.
• The second advantage is stability, both from a “funding” perspective, and also from an “operational” perspective. If a program requires funding for multiple years in order to achieve a certain outcome, incremental budgeting is structured to ensure that funds will keep flowing to the program; and this approach will also ensure that departments are operated in a consistent and stable manner for long periods of time.
There are however several downsides to incremental budgeting that make it a less than ideal choice. These downsides include:

- As the name implies, is it incremental in nature, which means it assumes only minor changes from the preceding budget, when in fact there may be major structural changes in the business or the business climate that call for much more significant budget changes. And, when the budget is carried forward with minor changes, there tends to be little incentive to conduct a comprehensive review of the budget, so that inefficiencies and budgetary slack are automatically rolled into new budgets.
- Incremental budgets fosters an attitude of “use it or lose it” in regard to budgeted expenditures, since a drop in expenditures in one period will be reflected in future periods; and managers tend to build too little revenue growth and excessive expenses into incremental budgets, so that they will always have favorable variances.
- Since an incremental budget allocates most funds to the same uses year over year, it is difficult to obtain a large funding allocation to direct at a new activity. Thus, incremental budgeting tends to foster the status quo, and doesn’t encourage risk taking, which can mean death in some industries.

These downsides might lead one to the conclusion that incremental budgeting can lead to such a conservative mindset that the budgeting process itself may actually be a driver in killing the company over time. Alternatively, businesses can engage in a thorough strategic re-assessment of the business from the bottom up when constructing its budget. This notion of what is commonly referred to as zero based budgeting, forces a detailed investigation of expenditures, and can result in significant changes to the allocation of funds from period to period if the business conditions justify such re-allocations.

**Zero Based Budgeting**

As we’ve mentioned, zero-based budgeting is an approach to planning and budgeting decision-making that reverses the traditional process of incremental budgeting - where departmental managers justify only variances over prior “baseline approved budgets”. By contrast, in zero-based budgeting, every line item of the budget must be approved, rather than only changes to line items. Zero-based budgeting requires that budget line item requests be re-evaluated thoroughly every budget cycle, starting from an assumption of no allocation of funds, and an evaluation of the line item’s expense contribution to the business’s overall strategic objectives.

The next step in budgeting is to “marry” these budget requests to targeted operational changes that are intended to improve the competitive position of a business, and we have driver based budgeting, the last and final type of budgeting we will discuss here.

**Driver Based Planning**

Driver-based planning is an approach that attempts to identify an organization’s key business drivers, and uses those drivers as key inputs to business models that calculate how those inputs impact an organization’s success. The goal of driver-based planning is to focus business’s resources into those things that are most likely to drive success.

Identifying business drivers is tricky when done subjectively, and people within the same organization can have different perceptions about what the key drivers for the business’s success are. This is why driver based forecasting can provide an objective mechanism to measure which drivers are most strongly correlated to the business’s success. The driver based models can be created with spreadsheets like Excel, or with more advanced data modeling software applications such as Oracle Hyperion Planning (just to name one).

The concept of driver-based planning was gained notoriety in the movie MoneyBall; the film, based on the book by Michael Lewis, was a true story about how the Oakland A’s assistant general manager used statistical analysis to identify the organization’s key driver for success - on-base percentage. Once this became a key driver in player selection, the A’s were able to successfully compete against other teams that spent much more on player payroll.

The most important aspect of driver based planning is that it aligns operational goals and forecasts with the financial forecasts, and provides potential leading indicators for the health of the organization as data is collected and compared against the “plan”.

**Forecasting**

Forecasting is a lot like budgeting, however, forecasting uses information regarding current financial conditions and expected future financial and operational conditions, to project a company’s financial position, cash flows, sales and other figures into the future. Forecasts tend to change over time as a company’s financial position and other influencing factors change.

Forecasting and budgeting are both ways of preparing an organization for the future; however, budgets are typically prepared yearly, while forecasts are prepared more frequently – usually quarterly, or even monthly. Forecasts tend to change based on financial conditions, whereas budgets are more concrete - once the budget is set, the organization will try to stick to it. After the budget period is over, a review is conducted to determine areas where the budget was correct and where it was incorrect so that the next budget can be revised accordingly. In short, budgets identify what the organization wants to accomplish; forecasts identify what can be accomplished given the current financial position of the company and the likely financial, operational and market conditions of the near future.

Of course many of the budgeting processes previously described can also be used in forecasting, such as driver based planning, etc.

**Rolling Forecasts**

Rolling forecasts are a method of planning that looks beyond the current financial year, and contain a minimum of 12 forecast periods (typically months) but can include 18, 24 or 36 or more. Each time the model is “rolled” forward and updated with a period of actual results, the forecast is extended to maintain the required number of forecast periods. This is depicted in the forecast on the following page.

The above example is a 12 month rolling forecast where the green cells indicate actual data being loaded into the monthly forecast, and an additional month being added as the forecast “rolls” from one month to the next. And, at the end of the year, the rolling forecast becomes the budget for the next year.

**Technology’s Part - Redux**

_A Hyperion Planning_

Again, according to Oracle, “Hyperion Planning is a centralized, Excel and Web-based planning, budgeting and forecasting solution that integrates financial and operational planning pro-
cesses and improves business predictability. Oracle Hyperion Planning provides an in-depth look at business operations and its related impact on financials, by tightly integrating financial and operational planning models. With Oracle Hyperion Planning you can meet your immediate financial planning needs while enabling a platform for future cross-functional expansion and automated process integration.”

So what does that mean?? Well, in short it means that Hyperion Planning has the functionality to do incremental, zero based, and driver based budgeting and forecasting; as well as the ability to do rolling forecasts – all the things that users in the financial planning and analysis (FP&A) group of the finance office want from a planning system.

And with Essbase as the underlying engine that powers Hyperion Planning, organizations have the ability to model complex business calculations. And this is where it all comes together hopefully... it’s important to understand the business model and the drivers to the business model, and how the organization supports and tracks those drivers to those business models that becomes so important. Typically these will be operational measures tracked in the organization’s ERP system, or CRM system, HR system, or some combination of these or other systems. Pulling all this data together requires coordinated integration, which we will discuss in more detail in Part 4 of the series, Data Integration, Ownership and Security. And, with this data in a system rather than in Excel spreadsheets, the data has controls and processes that can be reviewed and audited.

Just like the financial close process, the budgeting and forecasting process can have multiple steps that must be completed in a particular order, and may require approvals throughout the process that involve different people within the organization. These processes can be simple in many cases, such as simple manager approval... and they can also be complex in nature that may require the involvement of multiple people from different departments within the organization to complete, may involve data from multiple sources, and which may be automated at times.

Enter Workflow and Task Lists, the two features that exist in Hyperion Planning which provide a mechanism for managing those aspects of the process. Process Management (aka Workflow) allows administrators of the budgeting and forecasting process to define the approval processes required – both simple and complex; and, Task Lists allow the administrators of the process to build a “business process”, or sequence of steps for budgeting and forecasting users to follow.

**Conclusion**

In conclusion, the importance of understanding the “why and how” of the use of the Hyperion suite of tools is critically important in supporting Hyperion applications and environments, particularly when these financial cycles are considered against the additional IT requirements of performing upgrades and maintenance to software and infrastructure that support the applications.

In addition, the use of the Process Management and Workflow tools available in the Hyperion suite of applications can be extremely valuable in managing the financial processes of an organization.

Chris Chase is a managing partner with The Bean Group. He is a Hyperion Planning and Essbase professional with over 20 years of experience designing, developing, and implementing corporate IT solutions using OLAP and OLTP databases and object oriented programming languages. Chris is a long time member of RMUG and has served as the leader of the Hyperion/EPM special interest group (SIG) for the past 3 years. He is a frequent speaker at RMUG meetings, the Hyperion/EPM SIG, and at the annual Training Days conference, as well as other local conferences and national conferences such as Oracle Open World, Collaborate, and KScope. Chris is also a member of the board of directors for the Colorado chapter of the Oracle Apps Users Group (Colorado.oaug.org) and is also an active member of ODTUG.
NoSQL Databases and Data Modeling Techniques for a Document-Oriented NoSQL Database

by Robert T. Mason

Introduction

The increase of data volume (Big Data) during the last decade is attributed to a variety of data sources, such as social media, GPS data, sensor data, surveillance data, text documents, e-mails, etc. For example, the Internet of Things adds urgency for companies to be able to handle vast amounts of data (Devlin, 2014). Data that was once considered too expensive to store, can now be captured, stored and processed. A decade ago, large data stores that were measured in Terabytes are now being measured in Petabytes (1,000 Terabytes). According to the media “hoopla”, we are living in a brave new (improved) world that has been created by the ingenuity of Web 2.0 companies, such as Yahoo, Google, Amazon and Facebook (Mohan, 2013).

The term NoSQL has come to mean databases that are alternatives to the conventional RDBMS (e.g. Oracle, MS SQL Server and IBM DB2). However, scholars that have watched the evolution of the database technology over the last 30 years are cautious and skeptical (Mohan, 2013). Many of the lessons learned during the evolution of the RDBMS are being ignored or discounted by the venture capitalist companies at the center of the NoSQL database movement. Many of the technical problems that have been resolved and automated by RDBMS database vendors are now the responsibility of NoSQL database administrators and application developers. Mohan (2013) cautioned that this type of adhoc development approach for NoSQL databases can lead to long-term disastrous results for end users.

NoSQL Databases

Although traditional Relational Database Management Systems (RDBMS) have existed for decades and are constantly being improved by the database vendors, RDBMS’s struggle to handle the large volumes of data (Mohan, 2013). However, a new category of database technology called NoSQL databases is able to support larger volumes of data by providing faster data access and cost savings. Basically, the cost savings and improved performance of NoSQL databases results from a physical architecture that includes the use of inexpensive commodity servers that leverage distributed processing. For example, according to Cloudera (2014), traditional RDBMS SAN storage costs average $30,000+ per terabyte, whereas storage for NoSQL databases average $1000 per terabyte. This dramatic reduction in storage cost has made it possible to store data that was previously considered too expensive.

In addition to distributed processing and inexpensive hardware, NoSQL databases differ significantly on their approach to maintaining data integrity and consistency (Roe, 2012). A more relaxed approach to data consistency helps NoSQL databases improve the performance of data storage. Because RDBMS highly value data integrity, they use the ACID theorem for data consistency which was presented in the early 1980’s by Jim Grey. ACID is an acronym for Atomicity, Consistency, Isolation and Durability and supports the concept of a transaction.

Atomicity – ensures that all tasks with a transaction are performed completely (all or nothing).
Consistency – ensures that a transaction must leave the database in a consistent state at the end of the transaction.
Isolation – ensures that transactions are isolated and do not interfere with other transactions.
Durability – ensures that once the transaction is complete, the data will persist permanently, even in the event of a system failure.

In contrast, NoSQL databases use the CAP theorem (Consistency, Availability and Partition Tolerance) for data consistency which was presented in 2000 by Eric Brewer at an ACM symposium (Roe, 2012). The CAP Theorem states that of the three possible combinations of CAP, only two are available at a given point in time. In another words, NoSQL databases can have partition tolerance (in a distributed environment) and consistency or partition tolerance and availability, but not all three factors at the same time according to Abramova, Bernardino, & Furtado (2014). CAP theorem has evolved into what is now called BASE (Basically Available, Soft state and Eventual consistency).

Basically Available – means that data will be available, however the response from the database can be a failure to retrieve the data or the data retrieved may be in an inconsistent state.
Soft state – means that the data can change over time as the database seeks consistency.
Eventual consistency – means that sooner or later, the database will eventually become consistent.

Mohan (2013) argues that although NoSQL database do not guarantee the concept of ACID transactions, they must support some form of a smaller transaction to promote data consistency within the database. Early RDBMS’s supported the concept of uncommitted reads and different levels of locking. Therefore, the concept of BASE is not a new idea in the area of database technologies and distributed processing. ACID was applied to the RDBMS as customers demanded better data integrity and reliability from the RDBMS vendors. Leavitt (2010) cautions users that the lack of ACID requires additional programming to guarantee data consistency and therefore makes NoSQL databases less reliable. Since NoSQL databases don’t support SQL, complex query programming can be time-consuming and challenging.
NoSQL databases also differ from the traditional RDBMS in the areas of structure and scaling. An RDBMS requires that the database structure (schemas) must be defined in advance of loading and then accessing the database. This predefined requirement is often called Schema Write by industry experts and can be a burden when data is in an unstructured format (Moniruzzaman & Hossain, 2013). In contrast, NoSQL column-family databases, such as HBase and Cassandra, provide a flexible schema structure that facilitates changes to the schema definition when new types of data are encountered. This type of a flexible schema structure is called Schema Read. New data sources can be easily incorporated into a read schema within minutes, thus altering the structure and data content dynamically. However, Sadalage and Fowler (2013) caution that although a NoSQL database schema can be dynamically altered, there is still the consideration that existing database applications that use the database will have to be altered to use the new data structures. Thus, there is an expense of maintaining existing code to use new data structures, which should be considered when making structural changes to existing NoSQL databases.

RDBMS scaling is done vertically and is usually accomplished by adding additional memory and/or CPU to one or more servers. However, NoSQL databases can scale both horizontally and vertically (Abramova, Bernardino, & Furtado, 2014). Most commonly, additional nodes (commodity servers) are added to a NoSQL cluster (group of servers) to scale horizontally (called sharding). Because distributed processing is used for a NoSQL database, after new nodes are added to the cluster, the existing data is automatically distributed evenly across the cluster by the NoSQL database management system. To avoid data loss, complicated data replication methods are applied across commodity servers in preparation for an eventual server failure. The failed server can be easily replaced within minutes and the replicated data is used to populate the new server with the original data.

**Types of NoSQL Databases**

There are four types of NoSQL databases: Document-oriented, Key-Value Pairs, Wide Column (or Column Family) and Graph (Abramova, Bernardino, & Furtado, 2014). IT organizations will use one or more of the NoSQL database types based upon the characteristics of the data that must be processed.

**Document-oriented** – as the name implies, stores related information in the form of a document. In a third normal form data model, data is normalized (separated) into different entities with relationships to reduce redundancy and to avoid update anomalies. Within a document-oriented NoSQL database, the data is denormalized, semi-structured and stored hierarchically (Moniruzzaman & Hossain, 2013). For example, each book in a library of books can be stored in a collection of documents called BOOK. Not only will the book title be stored in a BOOK document, but details about the book such as a list of one or more authors, publication date, edition, publisher, publisher location and ISBN numbers will be also embedded in the same document.

In addition to embedding information within a particular document, it is possible to provide a reference to another collection of documents (Moniruzzaman & Hossain, 2013). A reference (link) is a similar to the concept of a foreign key that is used by RDBMS. A Document-oriented NoSQL Database has a similar structure to a XML document which is hierarchical. MongoDB is an example of a NoSQL document-oriented database. An example of a particular BOOK document is

<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Book Title</td>
<td>Business Intelligence and Analytics: Systems for Decision Support</td>
</tr>
<tr>
<td>Author (set)</td>
<td>Ramesh Sharda</td>
</tr>
<tr>
<td></td>
<td>Dursun Delen</td>
</tr>
<tr>
<td></td>
<td>Efraim Turban</td>
</tr>
<tr>
<td>Publication Date</td>
<td>2015</td>
</tr>
<tr>
<td>Edition</td>
<td>10th</td>
</tr>
<tr>
<td>Publisher</td>
<td>Pearson</td>
</tr>
</tbody>
</table>

**Key-Value Pairs** – stores information in form of matched pairs with only two columns permitted - the key (hashed key) and the value (Moniruzzaman & Hossain, 2013). The values can be simple text or complex data types such as sets of data. Data must be retrieved via an exact match on the key. The advantage of this type of NoSQL database is that new types of data about a book can easily be added to the database as new key value pairs. Examples of NoSQL databases that use Key-Value Pairs are Project Voldemort, Cache and Dynamo. In the prior book example, the book information from Figure 1 would be stored as shown in the table below:

<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Wide Column (Column Family)** – has a format of data storage that is very similar to RDBMS (Abramova, Bernardino, & Furtado, 2014). Although the RDBMS generally tends to have simple data types and a predefined schema (structure), Column-oriented NoSQL databases provide much more flexibility. They can support complex data types, unstructured text and graphics (e.g. jpeg, gif, bmp, etc.). In the example shown below; author, publication date, edition and publisher can all be included in a complex data type called book details. Cassandra is an example of a Column Family NoSQL database.

<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Book Details (includes authors, year, edition, publisher, etc.)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ramesh Sharda</td>
</tr>
<tr>
<td></td>
<td>Dursun Delen</td>
</tr>
<tr>
<td></td>
<td>Efraim Turban</td>
</tr>
<tr>
<td>Publication Date</td>
<td>2015</td>
</tr>
<tr>
<td>Edition</td>
<td>10th</td>
</tr>
<tr>
<td>Publisher</td>
<td>Pearson</td>
</tr>
</tbody>
</table>

**Graph** – supports data that has an undefined number of network connections (Abramova, Bernardino, & Furtado, 2014). This type of data supports map data, bus transportation links and

In addition to embedding information within a particular document, it is possible to provide a reference to another collection of documents (Moniruzzaman & Hossain, 2013). A reference (link) is a similar to the concept of a foreign key that is used by RDBMS. A Document-oriented NoSQL Database has a similar structure to a XML document which is hierarchical. MongoDB is an example of a NoSQL document-oriented database. An example of a particular BOOK document is

<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Ramesh Sharda</td>
</tr>
<tr>
<td></td>
<td>Dursun Delen</td>
</tr>
<tr>
<td></td>
<td>Efraim Turban</td>
</tr>
<tr>
<td>Publication Date</td>
<td>2015</td>
</tr>
<tr>
<td>Edition</td>
<td>10th</td>
</tr>
<tr>
<td>Publisher</td>
<td>Pearson</td>
</tr>
</tbody>
</table>

In addition to embedding information within a particular document, it is possible to provide a reference to another collection of documents (Moniruzzaman & Hossain, 2013). A reference (link) is a similar to the concept of a foreign key that is used by RDBMS. A Document-oriented NoSQL Database has a similar structure to a XML document which is hierarchical. MongoDB is an example of a NoSQL document-oriented database. An example of a particular BOOK document is
relationships found in social media. For example, traversing a graph to find the shortest distance between cities is a daunting task using a conventional RDBMS. However, a Graph NoSQL database can facilitate this type of processing. Allegro, Neo4j and Virtuoso are examples of Graph Databases. An example of a graph database for the distance between major cities in Colorado is shown here:

![Graph Database Example]

**Data Modeling Design Techniques for a MongoDB NoSQL Database**

MongoDB is an open source Document-oriented NoSQL database that was initially developed in 2007 by a company called 10gen (Medina, 2014). Data modeling on the conceptual level (CDM) and the logical data model (LDM) is very similar to what is done for a RDBMS (Hoberman, 2014). Either normalized data models (3rd normal form) or dimensional models (Star Schemas) are acceptable modeling approaches. The major changes to the data model occur when the data model is transformed from the LDM to a physical data model (PDM). For example, shown in Figure 3, is a very simple example of a 3rd normal form data model (a.k.a. Entity Relationship Diagram - ERD) for a fictitious Rental Car company that includes entities and the associated relationships.

![Data Model Example]

In this example, an occurrence of a rental is created when a vehicle is rented by a customer from a particular branch. Customers must have a physical address and a Branch must have a physical address.

Hoberman (2014) shows how to conduct a grouping process to convert a LDM into a PDM which can be later used to create the Document-oriented NoSQL Database. There are two options within MongoDB for modeling relationships, embedding data or referencing documents. Embedding is the process of merging together two or more entities into one entity using a hierarchy. Referencing is similar to creating a foreign key in a RDBMS that serves as a pointer from one entity (collection) to another entity (collection).

There are five heuristics that Hoberman (2014) suggests for deciding whether to embed or reference:

1. **Data that is frequently queried from multiple entities at the same time can be embedded into one document.**
2. **Entities that are considered dependent entities can be embedded into one entity.**
3. **If there is a one to one relationship between two entities, we embed one of the entities into the other entity.**
4. **Entities that experience similar volatility (inserts, updates and deletes) at the same rate can be embedded together.**
5. **Entities that are not key entities, but have relationships with key entities, can be referenced and not embedded.**

Using the LDM provided for the Rental Car Company and the heuristics listed above, entities are grouped together.

The logic behind these entity groupings for the PDM is as follows:

- **Branch Address** (black circle) and **Customer Address** (red circle) are both dependent entities of the Address entity. Therefore, two embedded entities for Customer Address and Branch Address will be created.
- **Rental, Customer and Branch** (blue circles) will be frequently queried together and will have the same volatility. Although Rental is not a dependent entity since it has a Rental_SID, the volatility makes it a good candidate to be embedded in a new entity called CustomerRental. This new entity will include the embedded attributes of Branch SID and Name and will have a reference to Vehicle which is an independent entity. The PDM after the grouping process has been completed is shown below. Notice that Customer Rental has references to Branch Address and

![Grouped Data Model]

The PDM can then be used as a template for a collection of documents. Below are samples of MongoDB documents for the Vehicle entity and the CustomerRental entity. Partitioning and additional secondary indexes can be added to improve performance.
during physical implementation (MongoDB, 2014).

Vehicle: 
{ Vehicle_SID: “2134567”,
VIN: “12342C33456XYY2”,
Available_Indicator: “Y”,
Odometer: “3507”,
Make: “Toyota”,
Model: “Camry”,
Year: “2014”,
Condition_Desc: “Scratches to paint on the left back door and scratches on the rear bumper”
}

In the example shown below, John R. Smith has rented two cars over the last year. The first car is the Toyota Camry that was picked up at the Aurora location and returned to the Broadway location. Notice the reference to the Toyota Camry via the Vehicle_SID. The second car rental included a different car that was rented at the end of August and returned in September. Notice how the two rentals are embedded in the document underneath CustomerRentals.

Customer Rental:
{CustomerSID: “987765”,
LastName: “Smith”,
FirstName: “John”,
MiddleName: “Ricardo”,
Rentals: [ 
{ Rental_SID: “2300453”,
Pickup_Date: ISODate(“2014-07-13”),
Branch_SID: “3374”,
Branch_Name: “Aurora”
Dropoff_Date: ISODate(“2014-07-19”),
Branch_SID: “3370”,
Branch_Name: “Broadway”,
Vehicle_SID: “2134567”},

{ Rental_SID: “2307111”,
Pickup_Date: ISODate(“2014-08-25”),
Branch_SID: “3374”,
Branch_Name: “Aurora”
Dropoff_Date: ISODate(“2014-09-02”),
Branch_SID: “3374”,
Branch_Name: “Aurora”,
Vehicle_SID: “2134977”},
] }

Abramova, Bernardino, & Furtado (2014) conducted performance testing using five different NoSQL databases that included the testing of MongoDB. They found that MongoDB had the largest increase in run time (slow performance) when the number of updates increased. MongoDB uses a data locking mechanism that has the direct effect of slowing update performance. However, in regards to read performance, MongoDB performed very well and is considered a database that is optimized for reads. The authors did not provide details about how the test data was modeled (e.g. embedded or referenced).

**Conclusion**

NoSQL databases are an important component of Big Data for storing and retrieving large amounts of data. RDBMS's use the ACID theorem for data consistency, whereas NoSQL Databases use BASE. An RDBMS scales vertically and NoSQL Databases can scale both horizontally (sharding) and vertically. Although NoSQL databases provide performance gains, some researchers are cautious and skeptical about data consistency. Four types of NoSQL databases are Document-oriented, Key-Value Pairs, Column-oriented and Graph. Data modeling for Document-oriented databases is similar to data modeling for traditional RDBMS during the conceptual and logical modeling phases. However, as was demonstrated in this paper, physical data modeling for a document-oriented database is different. Separate entities can be merged (denormalized) into one document by using embedding and the concept of a foreign key is supported by a reference.
Those of us in midlife or just joining the workforce happen to be living amid one of the most difficult professional environments in U.S. history. Between a tight job market, a shrinking middle class, and stagnant wages against inflation, getting ahead is a constant challenge. To make it even tougher, graduate degrees don't even necessarily guarantee an advantage like they did a generation ago.

The workforce seems awash with a groundswell of pressure – pressure to be the smartest, the best, and the most successful. Pressure to stand out, but in a way that still conforms to professional norms. Pressure to have enough experience to be proven and competent, but not too much, because of course you can't be old and played out. Maddening, right?

On top of all of that, there's social media. On long, tough days, it's all too easy to see other people's promotions and accomplishments and new vacation homes bandied about on LinkedIn and Facebook and wonder, “Am I that cool? Why haven't I gotten that far or risen that high?”

When you're constantly working to be better and do better, comparing yourself to others and wondering if you measure up, it's stressful. And stress is physical. When stress hormones flood our brains, we go into fight-or-flight mode, impeding our problem-solving ability and putting ourselves at risk for anxiety and depression.

One way people respond to stress is to become small-hearted. Maybe you apply high standards to yourself like a whip, lacerating yourself when you make a mistake or lose a prospect or fail to get promoted. Maybe you tell yourself you're stupid, or uncool, or not enough. Maybe you're mean to others, too. Maybe you're jealous of those who seem to be getting farther than you, who seem better somehow. Maybe you feel small and stupid, and so you try to puff yourself up by putting others down. As a card-carrying, type-A perfectionist, I have done all of that and more. It happens to the best of us, and in the end, it only hurts our chances for success and fulfillment.

When you lack compassion for yourself and others, it makes you less generous and less of a team player. It makes you more likely to be depressed and to stop trying new things for fear of failure. This means that lack of compassion is the enemy of inspiration and creativity, not to mention joy. It's small. It's boring. It's confining. Thomas Edison never would have invented electricity if he stopped creating solutions after one failure, or even ten or 20. Where there is no compassion, no freedom to fail, there is no risk, no experimentation, no creativity, and no entrepreneurship. And what about fun? How fun is it to do the same things over and over again, trying to do them more perfectly, while ignoring the calls and whispers of your heart?

Being big-hearted, being compassionate with yourself and others, may not be the most natural response to living in a stressful, pressure-filled world, but it is the most effective and fulfilling one. Research shows that people with self-compassion have more satisfying relationships, and are more favorably rated by their relationship partners. Self-compassion enhances work performance, too. Studies show that people who think kind, positive thoughts about themselves prior to shooting a basket or giving a presentation do a much better job than those who engage in negative self-talk before performing. And being kind and generous to others drives our own success. In the book Give & Take, Wharton professor Adam Grant shows that the highest achieving people – and the happiest – also happen to be the most generous.

I learned all of this while writing a book about forgiveness over the past two years. Exploring forgiveness through stories and science, I profiled extraordinary people in ten states and in the heart of Africa, I tested forgiveness techniques and read studies, and I confronted some dark chapters of my own past, altering my life in unexpected ways. Along the way, I did my best to stop beating myself and others over the head with my high standards. I'm still a perfectionist, but I don't torture myself as much over mistakes, and that has given me a new level of peace. It also gives me room to try new things - in writing and in life. My book about forgiveness launches this summer, and I never would have written it if I were consumed with being perfect or stuck due to fear of failure.

We can't change the fact that we're living in a time of great competition and pressure. But we can change how we respond to it. We don't have to let it determine the quality of our lives or the amount of joy in our hearts. We have a say about that. We can choose kindness and compassion.

Three tips on how to do that:
1. As the author Anne Lamott says, “Don’t compare your insides to other people’s outsides.” Remember when you look at social media: Those pictures show what people want you to see – the triumphs, the joys, the accomplishments - with none of the struggle and doubt and pain that’s probably happening at the same time. Our society doesn’t value sharing the hard stuff on the outside, on social media or anywhere else. But that doesn’t mean that every single person you see isn’t dealing with something hard on the inside.

2. Remember that just like you, most people are doing their best with what they have, from where they are. That goes for the guy who cut you off on the freeway and the colleague who bragged about her money or her promotion or making the sale or winning the award. It even goes for the father who let you down when you were a kid, the mother who hurt your feelings, or the boss who refused your raise or embarrassed you in front of your co-workers. They’re all just people, fumbling and imperfect, trying to do their best.

3. Take risks and step out of the box, no matter how old or young you are. Explore the big ideas, try that thing you’ve always wanted to try, attempt that goal that you don’t know you’re capable of attaining. And when you stumble or fall, as we all do, be kind to yourself.

If ever you’re in doubt, be compassionate and generous, to yourself and everyone else. You’ll be proud of that in ten years. And you’ll be happier and more successful, too. I promise.

Oracle E-Business Suite latest release 12.2 brings online patching as a standard to apply application patches. Essentially, during patch application in 12.2, users can stay online most of the time and will have minimal and predictable downtime during the course of online patching cycle (discussed later).

This is a concise technical article which covers EBS 12.2 Online Patching (in short passages and bullet points). In addition, this also covers some techniques which come in use while applying EBS 12.2 patches.

Online Patching Fundamentals

The phenomenon of online patching is possible by using two technology features in conjunction. These are

Edition Based Redefinition (EBR) feature of the database

EBR got introduced from 11gr2 onwards and is a feature of the database. As the term indicates, the database allows its editions to be created and managed. An Edition can be thought of as a copy of code objects (or editionable objects) in the existing database system.

There are two types of objects in a database (from 11gr2 onwards).

Editionable Objects -> These include all code objects such as packages (spec/Body), Views, Synonyms, Procedures, Functions, and Editioning views.

Non Editionable objects -> These are storage objects such as Tables, Indices, Sequences and materialized views.

Dual File system of the applications

EBS 12.2 delivers dual applications file systems (code trees) to facilitate online patching. These are named as ‘fs1’ and ‘fs2’ on the applications code directory and are designated as ‘RUN’ and ‘PATCH’ file systems at the initial installation of EBS 12.2. However, later designation of ‘fs1’ and ‘fs2’ as ‘RUN’ and ‘PATCH’ file system may change after every adop patching cycle.

Although dual file system facilitates Online Patching, there is also a third file system in EBS 12.2. This file system is called the non-editioned file system (fs_ne file system). This file system is used to house application patches, patches’ logs, java code sign certificate files, etc.

Online Patching Procedure

In EBS 12.2, application patching is done using ‘ADOP’, (Applications Dba Online Patching) tool.

ADOP is new and the Oracle recommended tool to apply application patches. The patching tool ‘adpatch’ should not be used explicitly to apply apps patches.

To achieve the online patching feature and a predictable downtime (cutover) window, patching is done using adop tool in the phases mentioned below.

Following are the phases in which an apps patch is applied.

Prepare Phase
- This phase prepares the system for patching
- Perform ADOP validations on all the application nodes
- Perform cleanup phase in standard mode, if not done explicitly before prepare phase
- Creates patch edition in the database
- Synchronize previously applied patches to current patch file system
- ConfigChange Detector (CCD) API identifies and propagates config changes. (NOTE: CCD detects most of the changes in Run File System and synces it to Patch File System. However, It does not sync the techstack patches applied to technology homes in application code tree)
- Preserve customizations using adop_sync.drv
- Checks to see if enough space is available in database
- Initiates ‘Online Patching In Progress’ (ADZDPATCH) concurrent program
- • Required online phase in adop patching cycle

Syntax:
$adop phase=prepare

Apply Phase
- Patches are applied to the Patch Edition of appl filesystem and database
- RFS and Database Run Edition are unaffected by the changes
- Users can continue to connect to the application and can perform their work
- Application code level changes are applied to Patch APPL_TOP
- All DB changes are applicable only to the Patch Edition, created in prepare phase
- Default patchtop is $NE_BASE/EBSapps/patch
• This can be changed by 'patchtop' option in apply phase
• All adpatch options are still valid to use
• Patches can be merged on-the-fly using merge=yes option
• Required online phase in adop patching cycle

Syntax:
$adop phase=apply patches=12345678
$adop phase=apply patches=12345678_E:u12345678.drv,
12345678_PTB:u12345678.drv
$adop phase=apply patches=12345678,87654321
merge=yes

Finalize Phase
• Compile invalid objects
• Generate derived objects
• Pre-compute DDL to be run at Cutover Phase (Downtime Phase)
• Can be invoked explicitly or implicitly run during cutover Phase
• Can be specified to run in 'quick' mode or 'full' mode ('quick'
is default; 'full' mode collects data dictionary statistics)
• Recommended to be done explicitly in order to reduce cutover (downtime) phase time
• Patching cycle can pause at this stage for as long as required
• Waiting for an appropriate downtime window (If there is a
requirement to apply another patch after the finalize phase ,
this can be done by invoking apply again followed by running
finalized again)
• Optional, however recommended, online phase in adop patching cycle

Syntax:
$adop phase=finalize [ finalize_mode=full ]

Cutover phase
• Performs ADOP validations on all the application nodes
• Users are logged off the system (even all 3rd party connections will be terminated)
• Application Services are brought down
• The Patch file system is promoted to be the Run file system, through context update
• The Patch database edition is promoted to be the Run database edition
• Services are brought back online on the patched (now ’run’) file system
• Terminates ‘ADZDPATCH’ concurrent program which was started in prepare phase
• Database does NOT need to be bounced
• Suggested to take online complete system backup just before cutover
• Required downtime phase in adop patching cycle

Syntax:
$adop phase=cutover

Important Note on Cutover Phase
1. By default, cutover phase will bring down and bring up all apps services. If you want to control operating services, you can bring down all services manually before commencing cutover.
2. To prevent auto startup of services by cutover , you can use ‘mtrestart=no’ with cutover syntax and then start services whenever deemed appropriate.
3. By default, cutover will wait indefinitely for in-progress concurrent requests to finish. You can specify time in minutes to specify how long adop should wait for concurrent managers before aborting it or you can abort the concurrent manager services before commencing cutover. This can be done by using ‘cm_wait=<max_mins_to_wait>’ with cutover command.

Cleanup Phase
• Drops Obsolete Crossedition Triggers (quick)
• Drops Obsolete Seed Data (quick)
• Drops obsolete editioned code objects (standard)
• Drops all obsolete code and data from earlier database editions (full)
• Optional, however recommended, online adop patching cycle phase
• If not run explicitly , the next prepare phase will perform cleanup in standard mode

Syntax:
adop phase=cleanup [ cleanup_mode={ quick|standard|full } ]
Standard mode is default

Image source : E-business suite 12.2 Maintenance Guide

All the above mentioned adop phases constitute an online patching cycle to apply 1 or more apps patches. Besides these phases, ADOP also offers special phases to efficiently manage EBS 12.2 applications environment. These special phases are mentioned below.

Abort Phase
• Abort will terminate the online patching cycle any time prior to cutover
• Rollback any changes made thus far by online patching cycle
• Checks for the existence of a patch edition and drops one if it exists.
• Cancels the ADZDPATCH concurrent program, if it is running.
• Typically required in the case of cannot-be-resolved issues with prior patching cycle
Also, ADOP stores all of its execution information in database tables.

The primary tables which can be queried are ‘ad_adop_sessions’ and ‘ad_adop_session_patches’.

**HOTPATCH and DOWNTIME Mode**

Apart from the online patching cycle phases discussed above to apply apps 12.2 patches, there are two other modes in which apps patches can be applied.

These are hotpatch and downtime modes respectively.

**Hotpatch mode:** This is similar to hotpatch mode of pre 12.2 releases. All application services can stay up online during the application of a patch in hotpatch mode.

Syntax:
```
adop phase=apply patches=12345678 hotpatch=yes
```

**Downtime mode:** This mode has been introduced in 12.2. All services need to be brought down during application of the patch in this mode.

Syntax:
```
adop phase=apply patches=12345678 apply_mode=downtime
```

**Important Note:** Both the above modes should not be used in production and production support environments unless, where explicitly documented, or when directed by Oracle Support or Development. Also both these modes cannot be used while a patching cycle is active.

**Restart ADOP**

In case of failure during any phase, restarting of adop is as easy as issuing the same command again. ADOP will resume from the point of failure.

In case of a failed patch during the apply phase,

To resume a failed patch, pass the additional options along with phase=apply as below:
```
adop phase=apply patches=12345678 abandon=no restart=yes
```

To restart failed patch from the beginning, pass additional options along with phase=apply as below:
```
adop phase=apply patches=12345678 abandon=yes restart=no
```

In case of reapplying the same patch again, which was applied successfully already
```
adop phase=apply patches=12345678 options=forceapply
```

**Case of the Abandoned Node**

In a multimode applications system, during the online patching cycle, ADOP runs its processes on all the nodes via remote invocation of adop commands through password-less ssh setup across the nodes.

Please note that we need to issue adop commands only on the primary node in the system.
If any subtask during any phase, or a complete phase, fails on any of the secondary nodes and if services on all other nodes in the system except the failed node(s) are sufficient to run EBS 12.2, adop will report the failure on the affected node during the end of that phase.

At this moment, it becomes important to investigate and resolve the failure followed by running the phase again, ensuring the phase completes successfully on all the nodes.

If for some reason, failure cannot be resolved and it is decided to move ahead with the subsequent adop phases, adop will prompt the below message

Previous tasks have failed or are incomplete on node: <servername>
Do you want adop to continue with other completed nodes [y/n]

If you respond ‘y’ to the above prompt, and all the essential services are available on completed nodes, adop will ignore any failed nodes and continue processing with the remaining available nodes.

If an essential service is not available on any of the remaining available nodes, adop cannot continue further and will display the following error:

Unable to continue with other completed nodes:
<comma-separated list of completed nodes>

Once the cutover phase completes on the primary node, adop will mark all failed nodes as ‘ABANDONED’.

An ‘ABANDONED’ node essentially means that it is not in sync with rest of the cluster nodes.

This can be checked by running $AD_TOP/sql/ADZDNODESTAT.sql

If one or more nodes get abandoned, it can be resolved by deleting and adding the affected node(s) back in the topology.

Please refer MOS 1677498.1 [How to Restore Abandoned Node in Oracle E-Business Suite 12.2]

Applying HRMS Legislative Patch (hrglobal.drv)

Applying HRMS Legislative Patch (hrglobal.drv) is a special case when using ADOP.

• Apply patch during an online patching cycle on a live 12.2 environment
• DataInstall utility should be run after prepare phase
• Apply hrglobal.drv using options=nocopyportion, nogenerateportion, forceapply

Please refer MOS 1469458.1 [DataInstall and HRGLOBAL Application: 12.2 Specifics] for details.

Post Patch Tasks

When the patch readme explicitly states post patch instructions, such as compiling a db object, all such tasks should be done on the patch edition during an active patching cycle.

For any reason, if the patch is applied in downtime/hotpatch mode, then the post patch instructions should be executed on the ‘RUN’ edition.

Customization/Cemli Promotions

It is always recommended to promote CEMLIs (Configuration, Extension, Modification, Localization, and Integration) during the online patching cycle after the prepare phase, moving the application files to patch the file system and compiling any custom code in the patch edition of the database.

However, if required, it is still fine to promote CEMLIs in the ‘RUN’ edition before online patching cycle begins, moving the application files to the run file system and compiling any custom code in the run edition of the database. Using this strategy, the code should not acquire any locks on the run time objects or executing any DDL operations on the apps seeded objects.

Under no circumstance should CEMLIs be moved to the ‘RUN’ edition in the middle of an online patching cycle, as it introduces a possibility of losing all CEMLI changes after the cutover.

If for any reason the patch is applied in downtime/hotpatch mode along with CEMLI promotion, then the custom objects (files/db) should be moved to ‘RUN’ edition.

Applying Techstack Patches

It is recommended to apply techstack patches to the patch file system during an online patching cycle in order to use the change window for applying other applications patches. This should be followed by cutover and running fs_clone.

However, if required, it is still fine to apply techstack patches to the run file system if there is no immediate online patching cycle or the techstack patch is a priority bugfix for the customer’s environment which cannot wait for an online patching cycle. This should also be followed by running fs_clone.

Single File System

As mentioned above, application’s dual file system is required for online patching cycle.

However in certain situations, such as in some development and test environments, one can have a single file system. This is done by installing Release 12.2 in the normal way, and then removing the $PATCH_BASE directory (i.e $rm -rf $PATCH_BASE).

The benefit of a single file system is that it will save the space of a second file system.

The disadvantage would be that patches cannot be applied using an online patching cycle.

Patches can only be applied in either hotpatch mode or downtime mode. In addition the cleanup phase is also possible in single file system.

Please note that a single file system is only supported in test or development systems, and not for production environments. Using a single file system is possible from AD/TXK Delta 6 patch level onwards.

Conclusion

As detailed in this article, Online Patching is a standard way of applying applications patches in E-business suite 12.2. This not only keeps the business users online for most of the patching duration but also facilitates minimal and predictable downtime at a business-suitable cutover window.

References:
Oracle E-Business Suite Documentation Web Library
MOS 1677498.1 [How to Restore Abandoned Node in Oracle E-Business Suite 12.2]
MOS 1469456.1 [DataInstall and HRGLOBAL Application: 12.2 Specifics]

Disclaimer: Opinions expressed in this article are mine only and not my employer’s.
Karen Cannell is president of TH Technology, a small consulting firm providing Oracle technology services, lately focused on Application Express. A mechanical engineer by degree (one of them), she has analyzed, designed, developed, converted, upgraded, enhanced, and otherwise worked on legacy and commercial database applications for over 25 years, concentrating on Oracle technologies since 1994. She has worked with Application Express since its Web DB and HTMLDB beginnings and continues to leverage the Oracle suite of tools to build quality web applications for clients in government, medical, and engineering industries. Karen is editor of the ODTUG Technical Journal and co-author of Agile Oracle Application Express, Expert Oracle Application Express and Beginning Oracle Application Express 4.

Sumeet Kabra

Sumeet is an Oracle Certified Database and E-business suite professional. He has over 10 years of experience managing and implementing Oracle Databases and Oracle E-business suite systems. He is BTech in Computer Science and Engineering from NIT, Jaipur, India. Sumeet has been member of RMOUG for two years and was a speaker in RMOUG Training Days 2015.

ACCELERATE YOUR DATABASE AT THE RIGHT COST
KAMINARIO K2 ALL-FLASH ARRAY

- Average $2/GB usable with guaranteed effective capacity and all-inclusive software
- High performance with consistently low latency
- Flexible scale-up and scale-out design
- Macro efficiency with selective deduplication and compression
- Enterprise-grade resiliency
- Ease of use and management

www.kaminario.com
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.

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

Sruthi Mandhapati
I am currently pursuing my Master's in Database technologies at Regis University. This was a great start to my career and I am very passionate about learning new technologies. I have given various presentations during my undergrad and won many awards. I would love to see myself as a professional software engineer down the lane and this scholarship really boosted my confidence. RMOUG is an awesome group and I love being a part of it.

MEMENTIS
Sensitive Data Lifecycle Management™
www.mentissoftware.com | sales@mentissoftware.com | 1 800 267 0838
I remember vividly the summer of 1985 falling in love…that is in love with my first computer. My Mom was a school teacher, and would get to bring home every summer the Apple IIe which had the double floppy disk drives. My Mom used to let me play a computer game called “Math Man”, which I was obsessed with. When I went back to school in the fall, the teachers said I belonged in the advanced math class and even talked about moving me up a grade in math. Having mild learning disabilities, school never came natural to me. But this was the first time ever I was with the smart kids.

I also remember our first home computer, when the internet first came out, and how cool it was to “chat” with people around the world. When I went off to college, I was a social butterfly and my parents worried sick about me because they could never reach me in my dorm room. It was my emails from the college library that seemed to calm them down and let them know I was doing well. Funny enough, it forced my parents to learn the latest technology. I ended up majoring in marketing management and sales management, but always seem to find myself in career roles where it required me to learn a new technology.

During college, I had a Marketing internship with the VP of Marketing at Diebold. The best advice he ever gave me was to work 10% harder than my peers and I will stick out like a sore thumb. After graduating college, I went to work for a small data storage media company as their Sales and Marketing Assistant where I was easily bored. I vividly remember the owner of the company only allowing me to have access to the internet for email. He believed that employees became less productive with access to the internet. Needless to say, I felt like an alien there.

In 2001, I was offered an opportunity to move to Colorado with a friend. Having little job experience and no fear, I moved away from home. Struggling for work, I found myself working in customer service at a midsize corporation. One of my daily tasks was entering customer data into SAP. I always found myself wondering why they did not create a self-service portal for customers and automate this process.

My first introduction to Oracle was selling a complimentary Oracle EBS product for automating IT processes for Financials and Manufacturing. All I knew when I started was Oracle EBS was the competitor ERP to SAP. My head was spinning trying to learn an entire technology stack and all the major players. All of my clients used different ERPs. Through customer conversations, I started learning quickly why people need a robust ERP and why EBS users loved this tool.

I worked at 4 different companies in Colorado, wearing multiple hats around Sales, Marketing and Operations. I have always enjoyed working directly with people, especially when I am passionate about a product or service. Oracle products became that passion and in 2014, I found myself at my current role with 6e Technologies, who specialize in Oracle EBS consulting and was just getting into Oracle Cloud consulting. I quickly had to come up to speed and absorb copious amounts of Oracle product information. I had no idea how many Oracle Products were out there.

After going to Oracle Open World, there was no turning back. I fell in love with the amazing loyal Oracle community and became the “newbie”. I started attending the RMOUG events and saw the same amazing community. Shortly after being a regular at RMOUG events, I received an email asking to volunteer for the Training Day conference, which opened up a new door with so many perks. “Me? Volunteer?” My first thought was absolutely - but wait - do I have the time? Life as a working mom was nowhere close to what I had imagined growing up. I don’t know if I would...
describe my days as juggling, prioritizing, making sacrifices or a blend of all three. I am one of the fortunate few who have an amazingly supportive spouse where we equally divide responsibilities at home, shuttling kids to activities and value each other’s careers. I knew I could find the time. This was my opportunity to jump into the Oracle community in a volunteer area that came second-nature to me.

Throughout my career, I have attended – as well as hosted - my share of conferences. But have you ever put on a large conference or been part of a committee for a conference with a low budget? It takes months of meticulous and creative planning from booking the conference rooms, having cutting edge content, good presenters, good food, snacks, sponsors, attendees and the list goes on and on. I have had my share of decorations falling, laptops crashing, no-show presenters or caterers arriving late. When I first heard RMOUG was held at the Colorado Convention Center in Denver, CO, I about fell out of my chair. How could RMOUG possibly pull this off? And then I learned the secret sauce, RMOUG Board volunteers recruiting more volunteers! I thought, “Wow, this is a great way to meet the Oracle community and I can jump in wherever needed”.

My favorite part of being an RMOUG volunteer was helping presenters at the TD conference. It was really easy. I assisted the presenters with timing presentations, troubleshooting technical difficulties, and handing out surveys. By assisting these presenters, I gained access to the firehouse abundance of Oracle presentations and was introduced to the highly skilled resources in my community. Information you would probably never find on the Oracle Knowledge Zones. The RMOUG volunteer members not only made me feel welcome as a newbie volunteer, but also answered my novice questions on the latest Oracle products, features, gaps, etc. In addition, there are networking capabilities available that you can’t exactly get through LinkedIn and other sites. I have made so many great connections through volunteering with RMOUG. After my first TD conference volunteering, I had made contacts around the globe and in many different business areas. But best of all, I have established some wonderful friendships. That is invaluable both in business and in life! Meeting this great community has inspired me to be involved on the RMOUG board to help carry on this tradition. Moving forward, I hope to assist RMOUG with marketing and increasing the attendance at every event.

Susan Llanes (jah-nez) aka TechSuzy
Follow @TechSuzy

Hobbies: Entertaining friends, Koi Pond, Gardening, Skiing and Love of Mountains

Note from the editor: I asked Susan about a funny story of a past conference she hosted. Her response was: “The strangest conference I have hosted was training at the Stanley Hotel in Estes Park, Colorado which is considered one of the most haunted hotels in the United States. Many attendees who didn’t even know the history of the hotel complained of strange occurrences to children singing outside of their room, witnessing a chair get pulled out by itself from a joker attendee who landed his bum on the floor and a door to a conference room get pushed open by itself that made the presenter speechless.”
Please Welcome Your 2015-2016 RMOUG Board of Directors

President - Tim Gorman
Vice President – Bob Mason
Secretary – Chris Chase
Treasurer – John Jeunnette
Director of Member Relations – pending acceptance
Director of Newsletters – Vince Giasolli
Director of Programs/Education – Dan Hotka
Director of Training Days – Karen Funston / Kellyn Pot’vin Gorman – Co – Directors
Director of Marketing – Susan Llanes
Webmaster – Art Marshall
Director of Special Interest Groups – Sruthi Kumar Annmniudu
Director of Social Media – Bobby Curtis
Director of Scholarship – Mark James

Our sincere appreciation goes to John Peterson, our outgoing President, for all his time and efforts. Thank you, John!
Index To Advertisers

Advisex................................................................................................. 2
Delphix .................................................................................................... 2
Kaminario.............................................................................................. 26
Mentis ..................................................................................................... 27
QEW August 2015.................................................................................. 31
Solarwinds............................................................................................. 21
Training Days 2016............................................................................... 32

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

August 7, 2015

Summer QEW At Elitch Gardens!

• OEM Hands-On Lab
• Lunch With Speakers & Peers
• Access to the Water/Theme Park

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

ProgramsDir@rmoug.org

Watch RMOUG’s Web Page for August Training Topics www.rmoug.org
SAVE THE DATE - FEBRUARY 9-11, 2016

RMOUG Training Days, February 9-11, 2016, at the Colorado Convention Center in Denver, CO, is the largest regional Oracle-related user group conference in North America.

At RMOUG Training Days, you will hear the same high-quality presentations as those of Oracle OpenWorld, with the added bonus of being able to talk personally with the speakers. Training Days also offers the best presentations in other database technologies such as MySQL, SQL Server, Hadoop, and more!

Want to bring your technical knowledge to RMOUG Training Days? The call for abstracts will be open August 3, 2015. Check www.rmou.org for the latest updates.

We already have sponsors and exhibitors signed up for 2016. Contact our vendor coordinator Jill Colsch at jill@teamycc.com, or 910.452.0006 to reserve your spot today!