TechJournal

New York Oracle Users Group

Fourth Quarter 2012

Wednesday, December 12, 2012
The New Yorker Hotel
481 Eighth Ave. (at 34th Street)

Sponsored by:
Oracle Corporation, CCH, Confio, EMC, Gridiron Systems, Dell/Quest Software, and Apress

Free for Paid 2012 Members
Don't Miss It!

In This Issue –

Presentation Papers from the September 2012 General Meeting
Sharing Read-only Tablespaces, by Iordan K. Iotzov
Integrating XML Using Oracle SQL Developer 3.1 and Oracle 11gR2, by Coleman Leviter
The Present and Future of Data Warehousing, by Anthony D. Noriega

Nothing hunts down Oracle performance issues like Confio Ignite™.

Over 50% of DBAs who try Ignite resolve a performance problem on the first day.

Start your **free trial at Confio.com/p-hog**

CONFIO SOFTWARE

NYOUG Officers / Chairpersons

ELECTED OFFICERS - 2012

President
Michael Olin
president@nyoug.org

Vice President
Mike La Magna
vicepresident@nyoug.org

Executive Director
Caryl Lee Fisher
execdir@nyoug.org

Treasurer
Robert Edwards
treasurer@nyoug.org

Secretary
Thomas Petite
secretary@nyoug.org

CHAIRPERSONS

Chairperson / WebMaster
Thomas Petite
info@nyoug.org

Chairperson / Technical Journal Editor
Melanie Caffrey
editor@nyoug.org

Chairperson / Member Services
Robert Edwards
membership@nyoug.org

Chairperson / Speaker Coordinator
Caryl Lee Fisher
speakers@nyoug.org

Chairperson / Vendor Relations
Caryl Lee Fisher
vendorcoordinator@nyoug.org

Chairperson / DBA SIG
Simay Alpoge
dbasig@nyoug.org

Chairperson / Data Warehousing SIG
Vikas Sawhney
dwsig@nyoug.org

Chairperson / Web SIG
Coleman Leviter
websig@nyoug.org

Chairperson / Long Island SIG
Simay Alpoge
lisig@nyoug.org

Director / Strategic Planning
Carl Esposito
planning@nyoug.org

CHAIRPERSON / VENUE COORDINATOR

Michael Medved
venuecoordinator@nyoug.org

EDITORS – TECH JOURNAL

Associate Editor
Jonathan F. Miller
jonathanfmiller@earthlink.net

Contributing Editor
Arup Nanda - DBA Corner

Contributing Editor
Jeff Bernknopf - Developers Corner

ORACLE LIAISON

Kim Marie Mancusi
Kim.Marie.Mancusi@oracle.com

PRESIDENTS EMERITUS OF NYOUG

Founder / President Emeritus
Moshe Tamir

President Emeritus
Tony Ziemba

Chairman / President Emeritus
Carl Esposito
cesposi@bers.nyc.gov

President Emeritus
Dr. Paul Dorsey

Table of Contents

Message from the President's Desk 12
Reduce Your Disk Footprint by Sharing Read-Only Tablespaces 15
Integrating XML Using Oracle SQL Developer 3.1 and Oracle Database 11g Release 2 28
RAC One-Node: Reduce Costs and Replace Third-Party Cluster Services with Grid Infrastructure and RAC One-Node 46
Modernizing Workflow and Data Integration: The Present and Future of Data Warehousing 60

New York Oracle Users Group, Inc.
#0208
67 Wall Street, 22nd floor
New York, NY 10005-3198
(212) 978-8890

NYC Special Winter General Meeting - presented by NYOUG & Oracle Corporation
Wednesday December 12, 2012 at the New Yorker Hotel – 481 Eighth Ave. (at 34th Street) in Manhattan
Sponsored by Oracle Corporation, CCH, Confio, EMC, Gridiron Systems, Dell/Quest Software, APress

8:00-9:00	**REGISTRATION AND BREAKFAST**			
9:00-9:30	**Introduction and Welcome: Michael Olin – NYOUG President**			
9:30-10:15	**KEYNOTE: "What's Next for Oracle Database?"** **Willie Hardie, Vice President for Oracle Database Management**			
	DBA TRACK	**DBA/DEVELOPER TRACK**	**Developer TRACK**	**Implementation TRACK**
LOCATION	**Crystal Ballroom**	**Sutton Place**	**Gramercy Park**	**Herald Square**
10:30 - 11:30 **Session 1**	**DBA 1** **"Exadata Demystified for Oracle DBAs"** Arup Nanda Starwood Hotels	**DBA/DEV 1** **"Oracle WebLogic: Foundation of Oracle Fusion Middleware"** Lawrence Manickam Toyork Systems, Inc.	**DEV 1** **"Oracle Application Express 4.2 – New Features"** Mark Sewtz Oracle Corporation	**IMP 1** **"Alternative A…es to Deplo…n the Data Cent…"** Cancelled Erik de la Iglesia GridIron Systems
11:45-12:45 **Session 2**	**DBA 2** **"Edition Based Redefinition: Testing Live Application Upgrades (Without Actually Being Live)"** Melanie Caffrey Oracle Corporation	**DBA/DEV 2** **"Database Virtualization and Instant Cloning"** Kyle Hailey Delphix	**DEV 2** **"The Database Emperor Has No Clothes"** David Teplow Integra Technology Consulting	**IMP 2** **"DBAs: Take Control of The Storage Needs for your Oracle Environments"** Sam Lucido EMC
12:45-2:00	**LUNCH**			
2:00-3:00 **Session 3**	**DBA 3** **"Hey Oracle Optimizer! Don't Mess with MY Plans."** Janis Griffin Confio Software	**DBA/DEV 3** **"Mobile-Enable Enterprise Applications with Oracle ADF"** Juan Camilo Ruiz Oracle Corporation	**DEV 3** **"Responsive Web Design in Oracle Application Express with HTML5 and CSS3"** Shakeeb Rahman & Marc Sewtz Oracle Corporation	**IMP 3** **"Exadata X2/X3-8: A Critical Review"** Mike Ault Texas Memory Systems, an IBM Company
3:15-4:15 **Session 4**	**DBA 4** **"Partitioning: What, When & How"** Arup Nanda, Starwood Hotels	**DBA/DEV 4** **"Twelve Tips for Successful OBIEE / Applications Implementations"** Shyam Varan Nath IBM/BIWA SIG	**DEV 4** **"The Maturity of Tabular Forms"** Josh Millinger Niantic Systems, Inc.	**IMP 4** **"Reduce Downtime During Migrations & Upgrades"** Jeffrey Surretsky Dell/Quest Software
4:15-5:00	**VENDOR RAFFLES**			

NYC Special Winter General Meeting
December 12, 2012
ABSTRACTS

KEYNOTE 9:30-10:15 "What's Next for Oracle Database?"

Consolidating databases onto Private Clouds, utilizing Database Services in Public Clouds, deploying Engineered instead of manually built systems and integrating Big Data into Business Intelligence applications and dashboards are top priorities for many IT organizations. Oracle Database is fundamental to the success of IT projects like these. But what can Oracle customers anticipate next? This keynote discusses the latest innovations from Oracle Database Server Technologies that can help accelerate IT projects, protect enterprise data, lower IT costs, and add value to the business.

Willie Hardie is responsible for Oracle Database product marketing focusing on growing Oracle's business in the global Database, Data Warehousing and Embedded Database markets. His areas of expertise include Oracle Database and Oracle Exadata as well as Oracle Real Application Clusters, Oracle In-Memory Database Cache, Oracle Advanced Compression and other key Oracle Database Options. He has been in IT for over 25 years and has specialized in relational database technologies for over 20 years. Originally from Edinburgh in Scotland, he has worked with Oracle Databases since Release 5.

10:30-11:30 AM - Session 1 Presentations

DBA 1 10:30-11:30 Crystal Ballroom "Exadata Demystified for Oracle DBAs"

Exadata is fast – even superfast. That is not a secret anymore. But what really makes it fast? This presentation explores the various building blocks of Exadata: Storage Indexes, Cell Offloading, Smart Scan, smart Flash and more - all the components that make the database machine faster. Attendees will learn what these concepts are, how they are used, and how they can jumpstart their transition from an Oracle DBA to a DMA.

Arup Nanda has been working as an Oracle DBA for the past 17 years. Arup trains and mentors DBAs, has coauthored 4 books, written some 300 published articles (including the most popular 11g and 10g New Features Series on the Oracle Technology Network) and given numerous presentations. He is an OCP DBA, an Oracle ACE Director and a member of the Oak Table Network. In 2003, he won the coveted Oracle DBA of the Year award. Arup blogs at arup.blogspot.com.

DBA/DEV 1 10:30-11:30 Sutton Place "Oracle WebLogic: Foundation of Oracle Fusion Middleware"

This presentation will cover JavaEE, Web Server, Application Server Models, as well as the core components of the Oracle WebLogic Application Server and explain its position in the Oracle Fusion Middleware Stack. Attendees will also learn how to install and configure WebLogic Server and deploy an application.

Lawrence Manickam is the President of Toyork Systems Inc, An Oracle Gold Partner. He has 12+ years of Architecture, Administration experience in all versions of WebLogic Application Server.

DEV 1 10:30-11:30 Gramercy Park "Oracle Application Express 4.2 – New Features"

Application Express 4.2 is known as the "mobile release". With Oracle Application Express 4.2, APEX has evolved into a modern and productive mobile development platform. Via live demonstration, attendees will experience how easily you can optimize your own applications for mobile devices such as smart phones and tablets. With Oracle Application Express

4.2, APEX has been enhanced to be the underlying infrastructure of the Oracle Cloud Database service. This presentation will describe the new features in Application Express 4.2, driven by the development of the Oracle Cloud: RESTful Web Services, Responsive Web Design, Security and a new suite of Business Productivity Applications included with APEX 4.2.

Marc Sewtz is a Senior Software Development Manager at Oracle Corporation in New York. With over sixteen years of industry experience, Marc previously held roles in Consulting, Sales and Product Development and now manages a global team of Software Developers and Product Managers in the Oracle Application Express (APEX) development team, part of the Oracle Database Tools group. Marc and his team are responsible for features such as the development of Mobile Web Applications with APEX, Reporting and Charting, Tabular Forms, Oracle Forms to APEX conversion and integration with Oracle Business Intelligence Publisher. Marc has a Master's degree in computer science from the University of Applied Sciences in Wedel, Germany.

IMP 1 10:30-11:30 Herald Square "Alternative Approaches to Deploying Flash in the Data Center"

Database owners face constant pressure to increase application performance. Satisfying the I/O demand of application servers (with their ever-increasing core count and memory footprint) has proven challenging for legacy storage systems. New technology, such as flash storage (SSDs) and hybrid flash arrays, promise to overcome these challenges. This presentation addresses issues surrounding SSD technology and hybrid flash deployments with a focus on network-based options for high-throughput applications such as databases and analytic systems.

Erik de la Iglesia is currently the chief architect at GridIron Systems, responsible for machine learning, analytics, and flash technology. He was previously a founder and chief architect of content security pioneer Reconnex (acquired by McAfee) and architect of WebStacks (acquired by Extreme Networks). With 30 patents issued, his work has spanned full and semi-custom logic design, programmable logic systems and software applications. Erik holds a BSEE from University of Florida and an MSEE from Stanford where he was an NSF Graduate Research Fellow.

11:45 AM-12:45 PM - Session 2 Presentations

DBA 2 11:45-12:45 Crystal Ballroom "Edition Based Redefinition: Testing Live Application Upgrades (Without Actually Being Live)"

No matter how "agile" you believe your development environment to be, if you're using PL/SQL code and patching or upgrading frequently, you'll find that your patches and upgrades almost always incur necessary downtime. With Oracle 11gR2's edition-based redefinition, re-creating a stored procedure, changing a trigger, adding or revoking a grant, and changing a view (to name a few actions) have been made easier with the ability to access more than one occurrence of such objects to stage changes in a schema in complete isolation. This helps you to see how your changes will behave in a production environment without their being "live". This presentation provides an introductory discussion of the edition-based redefinition feature, the edition object introduced in Oracle 11gR2, editionable and non-editionable object types, and includes example code for code patches and schema changes, as well as caveats.

Melanie Caffrey is a senior development manager for Oracle Corporation, providing front-end and back-end Oracle solutions for the business needs of various clients. She is co-author of several technical publications including *Expert PL/SQL Practices for Oracle Developers and DBAs* and *Expert Oracle Practices: Oracle Database Administration from the Oak Table*, (Apress), as well as *Oracle Web Application Programming for PL/SQL Developers* and *The Oracle DBA Interactive Workbook*, (Prentice Hall) and she writes the SQL 101 series of articles for *Oracle Magazine*. She instructed students in Columbia University's Computer Technology and Applications program in New York City, teaching advanced Oracle database administration and PL/SQL development, and she is a frequent Oracle conference speaker.

DBA/DEV 2 11:45-12:45 Sutton Place "Database Virtualization and Instant Cloning"

This presentation provides an overview of current technologies for virtualizing databases. With database virtualization and instant cloning, database copies can be made in seconds with almost no extra disk space usage. Database virtualization technology includes on-copy on-write file systems, journal file systems, point-in-time snapshots, point-in-time writeable clones, and the NFS technology stack. This technology stack will be explained and the presentation will discuss the differences in specific technologies as implemented by Oracle , Delphix, VMware, EMC, and NetApp.

Kyle Hailey is the designer of Embarcadero Technology's DB Optimizer and a principal designer for the Oracle Enterprise Manager performance pages. He is a member of the Oracle Oak Table, co-author of *Oracle Insights: Tales of the Oak Table*, and was a technical editor of Oracle Wait Interface. Currently he is working with industry leading software, kernel, and filesystem designers at Delphix to take corporate data management to a new level of agility.

DEV 2 11:45-12:45 Gramercy Park "The Database Emperor has no Clothes"

Transactional systems were designed for just that – transactions, namely a point in time when a purchase occurred or an event happened. Big Data is largely a result of the electronic record we now have about the activity that precedes and follows that purchase or event. This data dwarfs transactional data in terms of volume and, more importantly, it does not usually lend itself to the structure of tables and fields. This presentation will review the complex and time-consuming machinations we go through to build Data Warehouses using relational databases, and discuss Hadoop's inherent advantages, especially where "Big Data" is involved.

David Teplow began using Oracle with version 2 in 1981, and has worked as a consultant for Database Technologies (1986-1999) and Integra Technology Consulting (2000-present). He served as president of the Northeast OUG from 1992-1999, and on the board of the IOUG from 2003-2008.

IMP 2 11:45-12:45 Herald Square "DBAs: Take Control of the Storage Needs for your Oracle Environments"

Many database teams are trending towards managing the entire Oracle stack from the binaries down through storage. Moving away from a shared services IT model and towards a departmental team approach enables faster provisioning, reduces time-to-market, and empowers DBAs. This presentation describes another option in owning storage that enables your Oracle team, remains an open platform, and has all of the built-in mobility, availability, continuity and recoverability that databases demand. The presentation will show how database teams can own their own storage service catalog with features like built-in web based user portal, easy administration requiring no storage skills, and select a performance level to meet their database IOPS requirements.

Sam Lucido has over 16 years experience with Oracle database (7.x – 11g) and the E-Business suite (10.6 – 12). Nine of those years were spent working at Oracle Consulting based in Chicago for the "Performance Architecture Group." Sam now leads EMC's Oracle Solutions Enablement group in which he is the owner of the Everything Oracle at EMC Community: www.emc.com/everythingoracle.

2:00-3:00 PM - Session 3 Presentations

DBA 3 2:00-3:00 Crystal Ballroom "Hey Oracle Optimizer! Don't Mess with MY Plans"

Unexpected execution plans due to system changes such as optimizer version and statistics, schema/metadata definitions, and system settings can have disastrous effects on database performance. Prior to 11g, you could not guarantee that a plan will change for the better. This has caused some DBAs to freeze execution plans and optimizer statistics. SQL Plan

Management (SPM) is a new 11g feature that provides controlled plan evolution by not permitting the optimizer to select a plan that is slower than the one currently executing, thus ensuring stability and consistent SQL performance. This presentation will explain how to set up SQL plan baselines and use them effectively. Attendees will also learn the difference between SPM compared to previous Oracle plan stability features such as Outlines and SQL Profiles.

Janis Griffin has over 25 years of Oracle DBA experience including design, development and implementation of many critical database applications. Before coming to Confio, Janis held DBA positions primarily in the Telecom/Network Industry, working with both real-time network routing databases and OLTP B2B applications. She was responsible for mentoring other DBAs on best practices in database performance tuning and managing all aspects of the Voice Network databases. Additionally, Janis has many years of experience implementing Oracle Applications (E-Business) as a consultant. As a highly successful liaison between management and technical staff, Janis has proven to be an effective collaborator implementing cutting-edge Oracle solutions.

DBA/DEV 3 2:00-3:00 Sutton Place "Mobile-Enable Enterprise Applications with Oracle ADF"

This session discusses and demonstrates how Oracle Application Development Framework (Oracle ADF) can easily extend Oracle Fusion Middleware and enterprise application functionality to mobile users. Whether users are accessing them from an iPhone, iPad, Android phone/tablet, or even feature phone, Oracle ADF provides a common framework to deliver browser-based as well as on-device mobile applications. Although most of the mobile platforms require extensive and expensive efforts to integrate with the enterprise platform, Oracle simply extends the enterprise platform to mobile devices through Oracle ADF, which lowers TCO and helps you deliver what really matters: compelling mobile applications for enterprise application users. This presentation explains the new ADF Mobile development fundamentals to extend Java EE enterprise applications to mobile devices and how to leverage ADF skills to create mobile applications.

Juan Camilo Ruiz is a Principal Product Manager for Oracle Development Tools. He is a frequent speaker at Oracle user groups meetings.

DEV 3 2:00-3:00 Gramercy Park "Responsive Web Design in Oracle Application Express with HTML5 and CSS3"

Users are no longer tied down to their PCs. They are interacting with Web applications from an increasing array of devices and browsers, each with its own individual screen size and resolution, providing whole new challenges to Web developers. Oracle Application Express addresses these development challenges by extending its user interface themes to support different UI types, including modern HTML5 and Cascading Style Sheets 3 (CSS3)–based templates that deliver a quality experience to users regardless of screen size. This session introduces the UI-related changes in Oracle Application Express Release 4.2 and shows how developers can incorporate modern CSS techniques and design principles such as fluid grids and media queries into their application designs.

Shakeeb Rahman is a Principal Software Developer at Oracle Corporation in Reston, Virginia. He has been a member of the Oracle Application Express (APEX) product development team for over three years and has played a key role in modernizing the look and feel of the product, including the Application Builder, Web Sheets, and Themes. Shakeeb has a keen interest in designing crisp and intuitive user interfaces that take advantage of emerging web standards (HTML5/CSS3) and applying them to several projects within Oracle, including APEX, Oracle Cloud, and Oracle Store.

See DEV 1 for Marc Sewtz bio

IMP 3 2:00-3:00 Herald Square "Exadata X2/X3-8: A Critical Review "

Almost every Exadata presentation is in praise of the system. While it may deserve praise, people should also be aware of the possible pitfalls. This presentation provides a look at Exadata's up and down sides. This presentation describes the

pros and cons feature-by-feature. From the Exadata storage cell-only software bits to the hardware and licensing issues, this presentation covers the possible pitfalls associated with the Exadata platform.

Mike Ault has worked with Oracle since 1990. Mike has written or co-written over 25 Oracle related books. Mike is a frequent presenter at international, national and regional Oracle user meetings.

3:15-4:15 PM - Session 4 Presentations

DBA 4 3:15-4:15 Crystal Ballroom "Partitioning: What, When & How"

So you have heard about partitioning but don't know where to start, or you are familiar with it but want to double check your design. This session explains when to do partitioning for tables and indexes, which partitioning options to choose depending upon the problem you are trying to solve - purge, performance, archival, backup or whatever, potential issues to watch out for and so on. This presentation will *not* describe the syntax and commands for partitioning, but rather how to decide which partitioning scheme to choose. Includes Oracle 11g.

See DBA 1 for **Arup Nanda** bio.

DBA/DEV 4 3:15-4:15 Sutton Place "Twelve Steps for Successful OBIEE/Applications Implementations"

This session will provide 12 tips from the field to help implement OBIEE and BI Applications, risk free. Attendees will learn about the best practices using real world customer case studies. In addition, the full tech-stack involved in Data Warehousing and how to configure and customize it will be described as well as how companies can take their BI experience to the next level by utilizing Advanced Analytics.

Shyam Varan Nath is an OBIEE Architect with IBM and has 22 years of industry experience. He is OCP as well as Certified in OBIEE / BI Applications. He is Director of Product Integration, an IOUG Board Member, and President of IOUG's BIWA and Exadata SIGs.

DEV4 3:15-4:15 Gramercy Park "The Maturity of APEX Tabular Forms"

Tabular Forms have been a great way to show and edit multiple rows of data on one screen since APEX was born. However, they haven't always been easy to work with depending upon business requirements. Many of these issues have been addressed in Apex 4.X to make developers lives easier. This session will demonstrate how to take advantage of these latest features to achieve the biggest gain for the least of amount of work. It includes a comparison of the level of effort previously required vs. how easily tasks can be accomplished now. Features such as the enhanced Tabular Form Validations and filtered Row Processing options will be discussed. In addition, examples of how to use JavaScript/jQuery will also be shown so developers can see how easy it is to extend their own custom needs into a tabular form.

Josh Millinger has been working with APEX since before it was even a product. During his 11-year tenure at Oracle Corporation he built countless APEX applications which helped track everything from class registrations in the Partner Technology Center, a training facility he founded, to Oracle Partners, to data center hardware. He left Oracle in 2006 to form his own company, Niantic Systems, which focuses on APEX development. He is a frequent speaker at NYOUG meetings and also presents at ODTUG, IOUG, and OracleWorld.

IMP 4 3:15-4:15 Herald Square "Reduce Downtime During Migrations & Upgrades"

Perhaps your company has participated in a merger or acquisition and is experiencing a related data center consolidation. Or you may be considering a change in hardware platform, operating system or application upgrade. How can you reduce

downtime in these situations and allow business to continue while the migration activities occur? You can minimize the downtime and risk required for upgrading your Oracle database, moving your Oracle database to a different hardware platform, or both using various methods. Although they reap long-term business benefits, migrations are a major availability challenge that companies face today. Each method has its own trade-offs. Attendees will learn about which method is best for their organizations. Some typical migration situations include: data center and/or server consolidation, retiring older technology, platform version, and application upgrades.

Formerly a DBA, **Jeffrey Surretsky** has been in the IT industry for over 25 years and recently has been working at Dell Quest Software for over 12 years as a Strategic Systems Consultant where he has focused on their database solutions.

Message from the President's Desk

Michael Olin

Winter, 2012

All in the Family

My son is a freshman at a well regarded engineering university in Massachusetts that sits atop the US News rankings (http://www.usnews.com/education/worlds-best-universities-rankings/top-400-universities-in-the-world). While he has not formally declared a major, based upon his interests, and the other programs he applied to, we were pretty sure that he was headed towards a degree in bioengineering. As a parent, I was quite happy with this choice, especially with such a degree projected to be the most valuable based on Bureau of Labor Statistics projections for the next decade (http://www.forbes.com/sites/jennagoudreau/2012/05/15/best-top-most-valuable-college-majors-degrees). He was interested in the field, and had already been quite successful with his biomechanics-related research during high school. Imagine my surprise as I drove him back to campus after Thanksgiving when he told me quite casually that he was considering following in his parents' footsteps and pursuing a degree in computer science.

The Allure of Shiny Things

While he has always been quite adept at using technology as a tool, until this point, my son had never expressed any interest in what makes all of that technology work. He has never written a line of code. Although he has used sophisticated software in the biomechanics lab, and to analyze the data he collected, I doubt he has ever written a simple script or Excel macro to help automate the process. Take apart an old computer or dead MP3 player to see what's inside? Not even a moment's thought about it. However, now he is talking about taking an introductory programming class and becoming a Computer Science/Electrical Engineering major. Where, I asked, did this sudden interest come from? It turns out that he was looking at a different set of projections than those produced by the BLS. While we all know (or should know) that the plural of anecdote is not evidence, it turns out that CS/EE graduates of his university (and they are the majority) end up doing quite well. They are almost all employed immediately after graduation, many by the top technology firms in the world. Alternatively, they head right off to their own tech startup, often with university provided support. There is no disputing that alumni have done incredible things such as: inventing Ethernet, starting pioneering companies like TI, HP and DEC, and quite recently founding Dropbox. I doubt that my son expects that his name will be added to this extraordinary list, but he is pretty confident that, within a year, he too can land a summer internship at Google that not only pays fairly well but also provides him with a brand new Nexus tablet and smart phone, both free of charge.

Mommas, Don't Let Your Babies Grow Up To Be Coders

My initial reaction was not exactly supportive. My son reminded me that a few years ago, his mother suggested that because he was such a logical thinker, he might enjoy programming. Today, I don't think logical thinking is enough. It turns out that his friend who lined up a great internship during the break between semesters has been coding since he was a kid and already generates a steady income from a few of the smart phone apps he wrote. Quite a few of his classmates who are heading towards their CS/EE degree have been doing similar things for years. They tinker with hardware; they write software; they root their smart phones and do the most amazing things with hacked video game consoles. This is how they have focused their creative energies for years. For these students, a CS/EE degree puts some formal structure around concepts they have been exploring on their own for as long as they can remember.

My introduction to computing was similar, although the gadgets I had access to were significantly less sophisticated. I disassembled my Telstar (http://en.wikipedia.org/wiki/Telstar_game_console), one of the first home video games that played three variations of Pong, and was disappointed to discover that all of the sounds (actually, the one sound) it made were produced mechanically, with a spring loaded plastic rod that struck a diaphragm. At school, we had a timesharing

account on a PDP-11 running RSTS/E (http://en.wikipedia.org/wiki/RSTS/E) that we accessed via a 10 character-per-second teletype with punched paper tape for backing up and loading our programs (http://www.columbia.edu/cu/computinghistory/teletype.jpg). A friend and I quickly wrote a BASIC program that emulated the RSTS/E login and used it to steal the password to our school's system account. A decade later, I was building my first database systems using the just-released Oracle V4. Back then, programming was as much art as logic. In college, we learned about bootstrap loaders that only required a few instructions, since they would have to be entered into memory by hand. We programmed using assembly language, and discovered which data structures were the most efficient for accessing varying types of data. Computers were still fairly expensive; processing speeds were a fraction of what they are today (8 MHz vs. 4 GHz); and memory was limited. The best programmers used a wide variety of creative tricks to maximize the use of fairly limited resources.

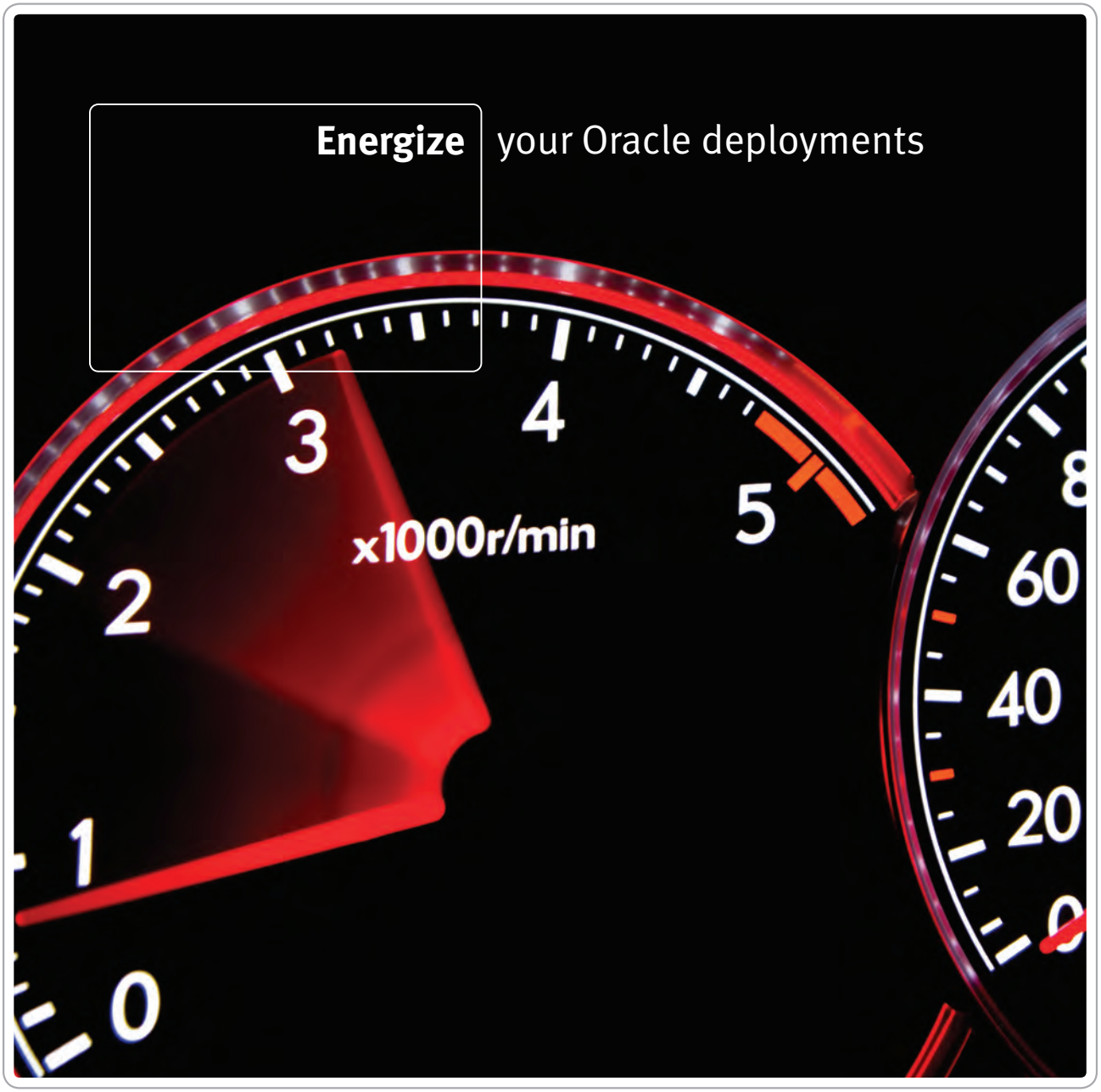
Now, I bear witness to how programming has become less of an art and more of a commodity. Computer programs today do not need to be elegant, efficient, or even particularly logical. A combination of Moore's Law and powerful frameworks have supplanted much of was fun and creative about programming with a drive to get the job done as quickly and inexpensively as possible. A whole industry has emerged with legions of commodity coders, both onshore and off, to feed the appetite for cheap programming. I have no doubts that some of my son's CS/EE classmates will be incredibly successful. I would expect, however, that their success will be more a function of their developing a brilliant idea rather than writing exquisite code. They may do the initial coding by themselves, just to develop a prototype. Once their idea takes off, they too will begin to rely on the commodity coders to keep things moving along. There is still a niche market for artful, extremely efficient coders, but I don't exactly see my son working on high-speed trading systems. It's just not the sort of thing he's interested in.

We're Number 3 (and 4)

For our children who do decide to pursue the family business, the outlook is actually still very good. According to the data, Computer Science and Software Engineering degrees are not such a bad choice after all. They don't fall far behind on the BLS list (Biochemistry comes in at number 2), but I just don't see the same opportunities that there were 20 years ago. Only a relatively small handful of Computer Science graduates will land jobs with companies that allow them to explore their full potential. Even fewer will be able to strike out on their own and bring a brilliant new idea to market. In a corporate setting, IT has become a drag on the balance sheet when, in the past, it used to drive innovation. Perhaps it's just me, but the scope of the projects I've seen lately and the vision behind them doesn't come close to what I remember from the days of the Dot Com bubble. Some of the best coders I ever met just fell into their jobs. They were writers or research scientists or dancers before they discovered programming. Now, I meet great programmers who want to be fashion designers or open a restaurant. If my son was passionate about programming, I'd tell him to go for it, but I'm not so sure that he'll go from completely uninterested to being consumed by the idea in a few short years. If he comes up with an incredible idea that requires programming to bring it to life, I am confident that he will have plenty of friends to whom he can turn. For now, I'd rather see him get the education he needs to pursue his passion in a research lab. I'm pretty sure that he'll be much happier working on a project to improve peoples' physical, rather than virtual lives. It sure seems a lot more exciting than what his dad has been doing lately.

Michael Olin

EMC²
where information lives®



Reduce risk and accelerate your application deployments by drawing on the power of our Oracle® expertise.

With EMC® Proven® Solutions, your information infrastructure accelerates towards greater productivity. Learn more at www.EMC.com/oraclesolutions.

EMC² PROVEN SOLUTIONS

Reduce Your Disk Footprint by Sharing Read-Only Tablespaces

Iordan K. Iotzov, News America Marketing

Introduction

Managing the storage needs of any rapidly growing modern database system is a major challenge. Oracle RDBMS, particularly version 11g, has many powerful compression features that allow us to reduce the size of a database. Sensible database design and strict enforcement of retention policies are also instrumental in keeping database growth under control. Our ability to cut the disk footprint by sharing read-only tablespaces among non-production database copies, however, is somehow limited and underused.

The paper provides a detailed overview of the benefits and restrictions of sharing read-only tablespaces using the transportable tablespace method – the only supported way to share datafiles and tablespaces among different databases. To overcome the limitations of the transportable tablespace method, specifically the self-containment requirement, a non-supported universal method for sharing read-only tablespaces among copies of a database is proposed and explained in detail.

Guidance on how to leverage physical design techniques, such as partitioning, to maximize the benefits of the universal method is also provided. The paper also covers version specific behaviors and workarounds, impact on backup and restore procedures, and other administrative restriction.

Basic Concepts in Sharing Data among Databases

The size of the average database has grown significantly over the years. In addition, as most companies adopt stricter SDLC processes, the number of non-production database copies has also increased. In fact, research (Delphix®, n.d. a) shows that each production database has up to ten non-production full copies. Figure 1 shows that most of the disk space is used by non-production systems. Even though constant technological improvements allowed for bigger and better storage devices, the procurement and management of storage remains a significant burden for many IT shops.

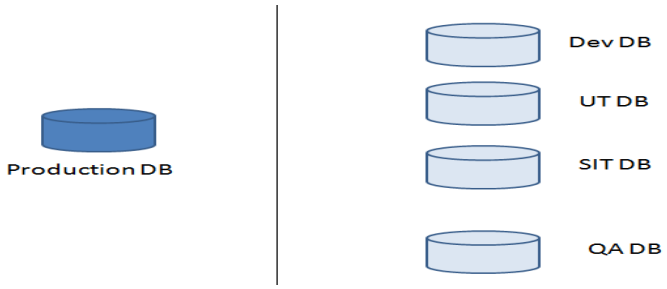


Figure 1. Disk Usage across Database Environments

Any reduction of the size of a production DB can result in significant savings across the enterprise. There are varieties of ways to reduce database size. Compression is a popular option because it is easy to implement and frequently requires no functional testing. In addition, better table and index compression algorithms are frequently introduced with new Oracle versions, allowing greater disk usage reduction as we upgrade. Adherence to good database design practices, particularly avoiding unnecessary denormalization, can bring in significant database size reduction. While retention policies are usually driven by legal and functional requirements, their strict enforcement can also help us curb the growth of the database.

Another powerful method for reducing disk usage is sharing data among multiple non-production databases. It is important to note that sharing data efforts are supplemental to the efforts to reduce the size of the production database. That is, it is possible to compress data and then share the compressed data among multiple non-production databases, getting the benefits of compression as well as the benefits of sharing.

The proprietary Delphix virtualization solution (Delphix®, n.d. b) is one way to get the benefits of sharing data among multiple non-production database copies. Clonedb, a new Direct NFS (DNFS) feature introduced in Oracle 11.2.0.2 patchset, might also be used to share a single backup set among multiple databases (Oracle-Base, n.d.).
Sharing read-only tablespaces is not a new concept. It was first introduced in Oracle 10g Release 1 and it is available in all versions and releases ever since (Oracle Documentation, n.d a). A simple and elegant method, sharing RO tablespaces allows us to cut the disk footprint without any restriction of the functionality of the involved non-production database copies.

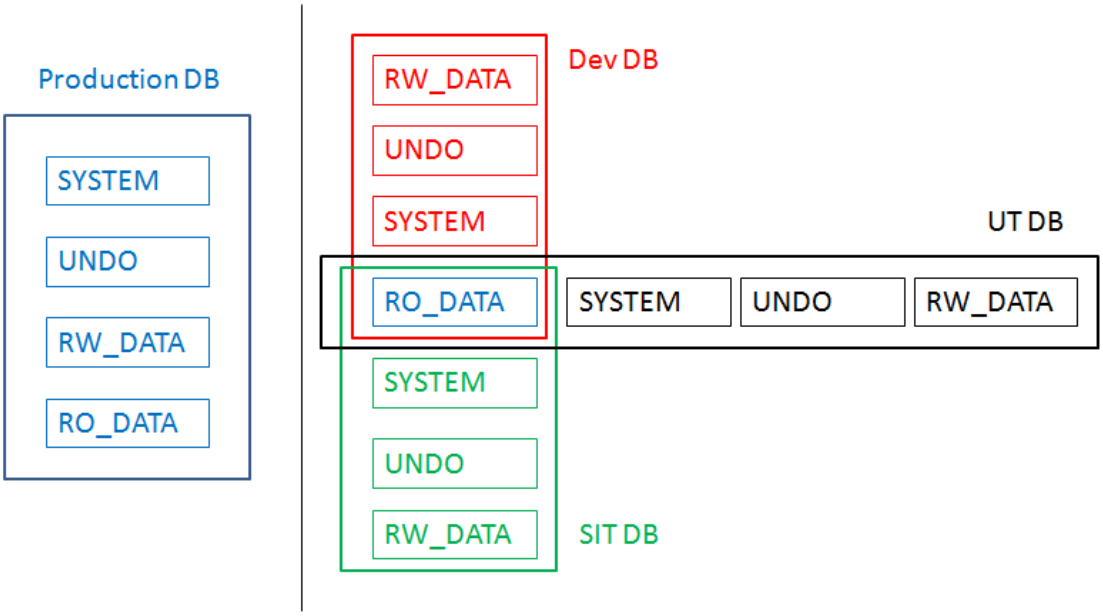


Figure 2. Basic Model of Sharing a Read-only Tablespace among Multiple Databases

In order for this method to have any significant impact on the overall disk footprint, considerable part of the database has to be placed in read-only tablespaces. In essence, the practical success of this method is predicated on how much of the data can be carved into read-only tablespaces. I believe that even though there are very few read-only entities in a database, there is plenty of read-only data in variety of different entities. Most temporal entities allow new records to be inserted, but do not allow any update of the inserted records once they have been in the system for a certain period of time. Sound data architecture dictates that most transaction and audit entities are typically insert-only. For instance, if a table stores ATM transactions, no one should be able to modify the entries retroactively. Identifying and carving out immutable data, usually with the help of the Oracle partitioning option, is vital for this approach.

Sharing Read-Only Tablespaces Using Transportable Tablespaces

Transportable tablespaces is a well-established, powerful and versatile method for moving data between different databases. This method was first introduced in Oracle 8i, and it is especially suitable for transporting large data sets and migrating big tablespaces or whole databases to a new hardware platform.
Figure 3 illustrates the basic concept behind the procedure. A transportable tablespace set (TTS) consists of data files for a set of tablespaces and an export file containing structural metadata for that set of tablespaces. Since the tablespaces to be exported into a TTS have to be in read-only mode, a quite significant restriction, an auxiliary partial copy of the source database is frequently used. The backup of the datafiles in a TTS is usually created with RMAN. If the source platform and the destination platform are of different endianness, then an additional conversion step is needed. The file with structural metadata is typically generated using DataPump export with TRANSPORTABLE_TABLESPACES option. The metadata file contains data dictionary information describing the structures in the tablespaces to be exported.

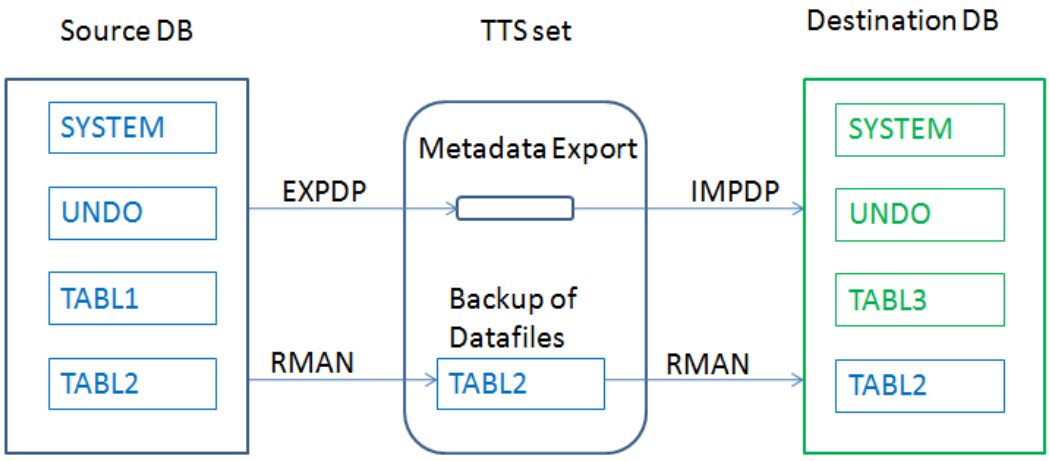


Figure 3. Migrating Data Using Transportable Tablespaces

The procedure for plugging a TTS into a database is straightforward. First, the datafiles are restored to, and if needed converted to, the destination DB server. After that, a Data Pump metadata import is done. The purpose of that import is to "register" all objects in the datafiles with the data dictionary of the new database.
While there are many useful applications of the transportable tablespace method, we are going to focus on sharing a read-only tablespace among databases. The basic idea behind this method is to plug the same datafiles into different databases. Naturally, the datafiles must be accessible by all destination databases. Also, while being part of multiple databases, the datafiles must remain read-only at all times.
Figure 4 illustrates the process. First, the datafiles that belong to RO_DATA, the tablespace in the TTS, are imported into a location that is accessible by all target databases. Next, the metadata export file is imported into Dev DB. After that, the same metadata export file is imported into UT DB. Finally, SIT DB joins the shared read-only tablespace configuration.

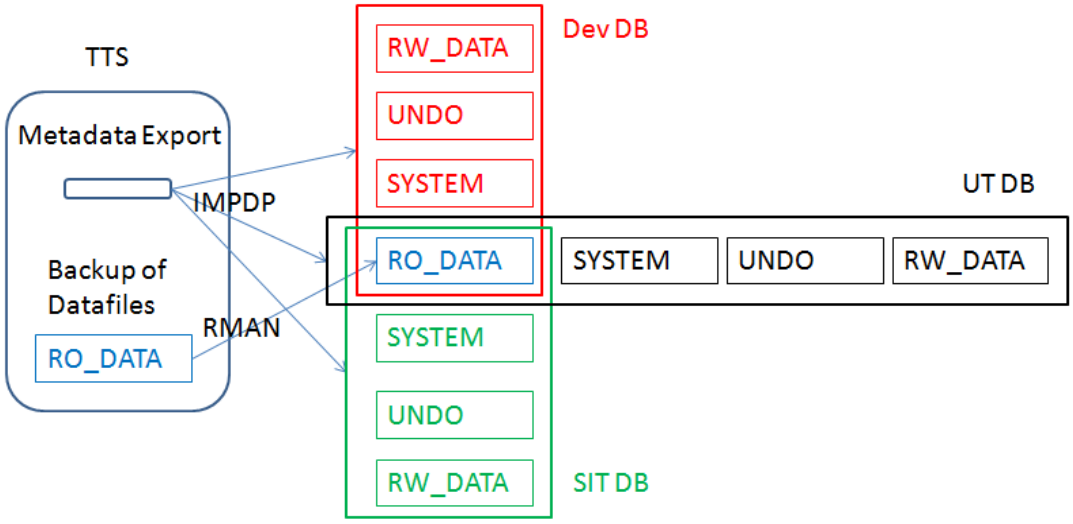


Figure 4. Sharing a RO Tablespace Using Transportable Tablespaces

While sharing read-only tablespaces using TTS is supported and simple method, it has one major restriction – the tablespaces included in the TTS must be self-contained. That means that the tablespace set to be shared should not have an index for a table outside of the tablespaces set. Neither should a partitioned table be partially contained in tablespace set. Also, every referential integrity constraint should be pointing to a table within the set's boundary (Oracle Documentation, n.d b).

The constraint restriction can be resolved by not including any constraints in the TTS. The index restriction can largely be eliminated with careful physical design. The partition restriction, however, is a difficult one. Moreover, as previously noted, usually the read-only data is a part of a table that allows inserts and possibly some updates, and table partitioning is likely the only way to carve the read-only portion into a read-only tablespace.

Universal Method for Sharing Read-Only Tablespaces

To overcome the restrictions of the transportable tablespace method, particularly the partition limitation, I propose a universal method for sharing read-only tablespaces among copies of a source DB. The copies must be on the same platform and run the same DB version as the source DB. It is important to note that the universal method is not supported by Oracle at this time.

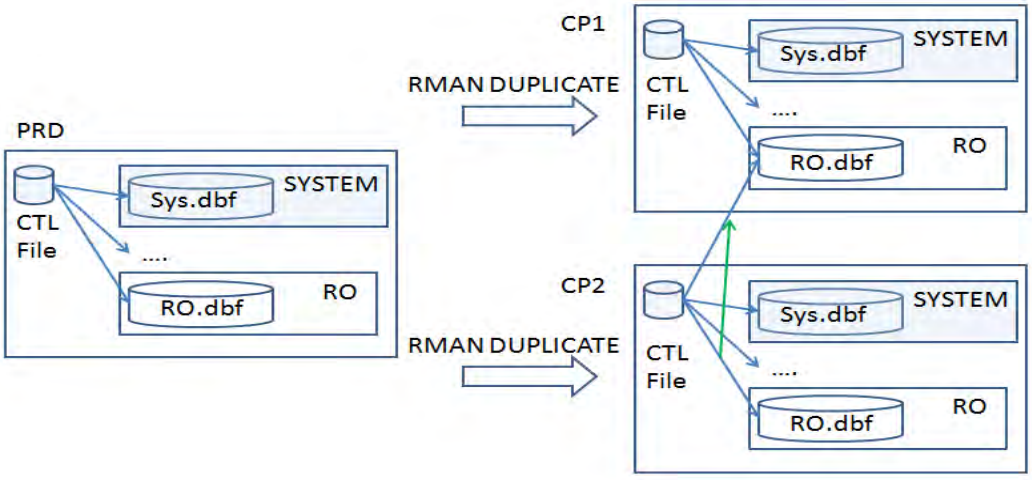


Figure 5. Universal Method for Sharing Read-only Tablespaces

Figure 5 provides a basic illustration of the universal method. To better demonstrate the method, let's set up the source database PRD. The code fragment below creates a table that would be part of a read-only tablespace, inserts a record into that table, and then makes the tablespace read-only.

```
PRD:
SQL> connect tst_user
SQL> create table ro_tab (txt varchar2(100)) tablespace ro ;
SQL> insert into  ro_tab values ('resides in RO tablespace');
SQL> connect system
SQL> alter tablespace ro read only;
SQL> connect tst_user
SQL> delete ro_tab;
delete ro_tab
       *
ERROR at line 1:
ORA-00372: file 6 cannot be modified at this time
ORA-01110: data file 6: '+DATA/prd/datafile/ro.8187.792687981'
```

Once the test table is set up, we refresh the PRD database into CP1 and CP2 databases using RMAN DUPLICATE. CP1 and CP2 databases need not be created from the same backup time. It is required, however, that RO tablespace in PRD stays read-only at all times. Figure 6 shows the datafiles in PRD, CP1 and CP2 immediately after the refresh.

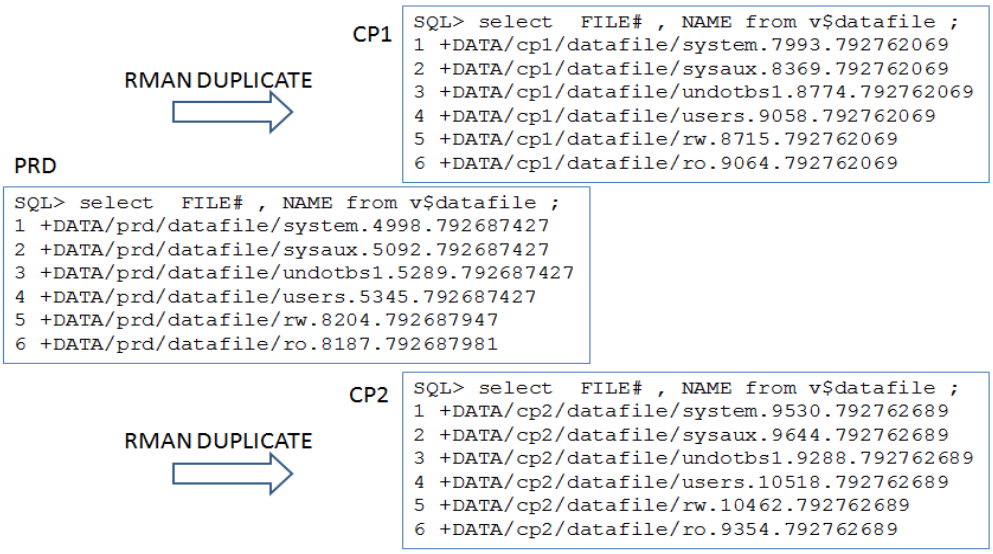


Figure 6. Datafiles in PRD, CP1 and CP2 after a Refresh

The procedure for sharing RO tablespaces between CP1 and CP2 consists of putting the RO tablespace in CP2 in offline mode, linking the pointer to the RO datafile in CP2 (`'+DATA/cp2/datafile/ro.9354.792762689'`) to the RO datafile in CP1 (`'+DATA/cp1/datafile/ro.9064.792762069'`) and putting the RO tablespace in CP2 back to online mode. The universal method is not documented nor supported by Oracle at this time.

```
CP2:
SQL> alter tablespace ro offline ;
Tablespace altered.
SQL>alter database rename file '+DATA/cp2/datafile/ro.9354.792762689'
to '+DATA/cp1/datafile/ro.9064.792762069';
Database altered.
SQL> alter tablespace ro online ;
Tablespace altered.
SQL> select * from tst_user.ro_tab;
resides in RO tablespace
```

Note that the universal procedure does not require any metadata import. The data dictionary in CP2 describes the RO datafile in CP2 (`'+DATA/cp2/datafile/ro.9354.792762689'`). Since the RO datafile in CP1 (`'+DATA/cp1/datafile/ro.9064.792762069'`) contains the same data as the RO datafile in CP2, except for few different file header bytes, then the data dictionary in CP2 can adequately describe the CP1 RO datafile. This is my opinion only - the universal method is not supported by Oracle at this time.

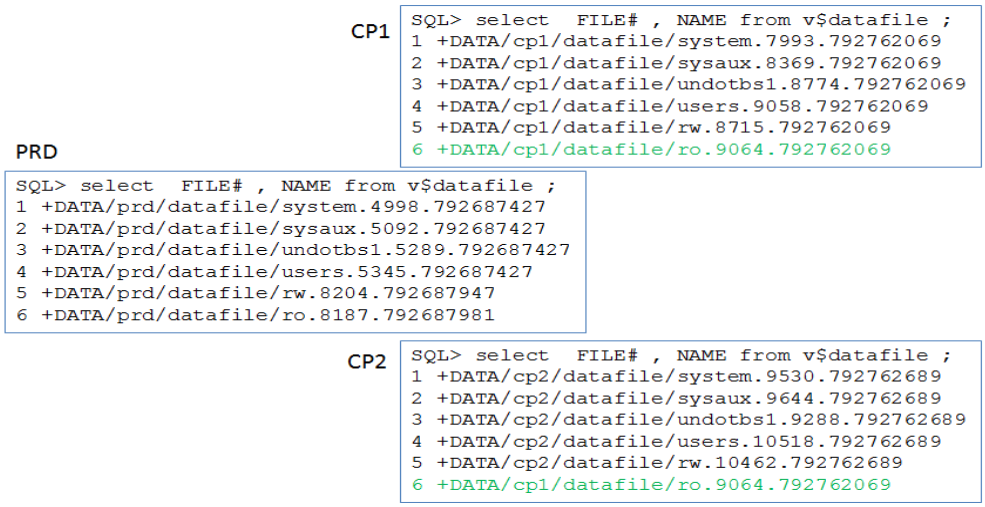


Figure 7. Datafiles in PRD, CP1 and CP2 after the Universal Method

Figure 7 shows the datafiles in PRD, CP1 and CP2 after the universal sharing method has been applied. Please note that file#6 in CP1 and CP2 point to the same location. As already mentioned, Figure 5 shows a graphical illustration of the universal procedure. After the RO tablespace in CP2 is put into offline mode, the CP2 controlfile is reconfigured to use the RO tablespace datafile that belongs to CP1. Next, the RO tablespace in CP2 is put back to online mode and the datafile associated with the "old' RO tablespace in CP2 (`'+DATA/cp2/datafile/ro.9354.792762689'`) is deleted.

Since the universal procedure is not supported by Oracle, I cannot claim with full certainly that CP1 and CP2 would function properly under all circumstances. Most basic operations, such as instance restart and open with resetlog, however, appear to be working fine.

```
CP1:
SQL> shutdown immediate;
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL> startup;
ORACLE instance started.
Database mounted.
Database opened.
SQL> select * from tst_user.ro_tab;
resides in RO tablespace

CP2:
SQL> shutdown immediate;
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL> startup mount;
ORACLE instance started.
Database mounted.
SQL> recover database until cancel;
Media recovery complete.
SQL> alter database open resetlogs;
Database altered.
SQL> select * from tst_user.ro_tab;
resides in RO tablespace
```

```
CP1:
SQL> select * from tst_user.ro_tab;
resides in RO tablespace
```

The above implementation of the universal method is not optimal though. We refresh the RO tablespace in CP2 only to delete it shortly after. We can utilize DUPLICATE SKIP TABLESPACE option to prevent transferring the data file, while still keeping the file slot in CP2's controlfile. Figure 8 shows how the improved universal method would work.

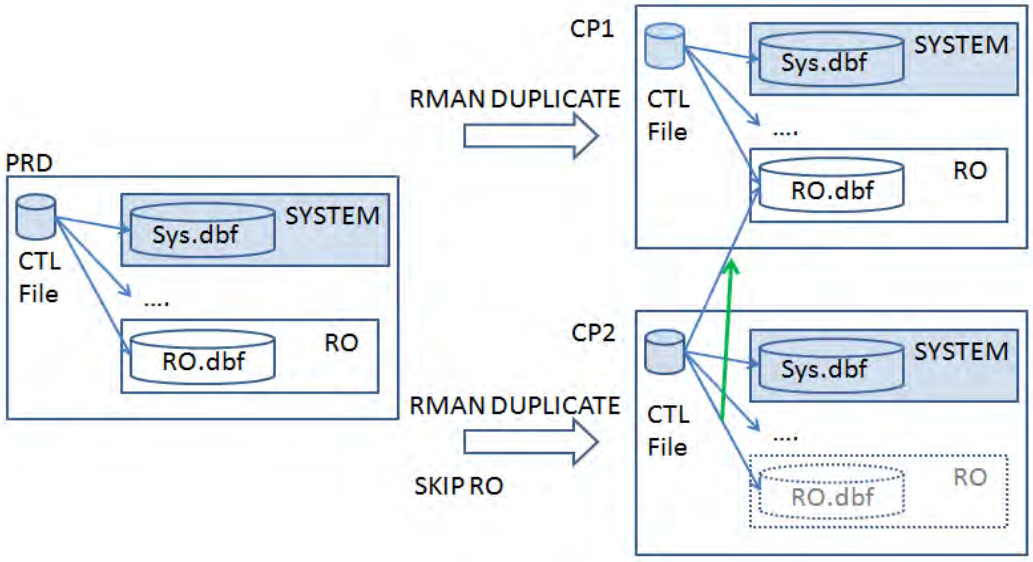


Figure 8. Improved Universal Method – CP2 RO Tablespace Is Not Brought In

The universal method only needs the slot for the RO datafile in CP2, so it can re-link it to the respective RO datafile in CP1. Figure 9 shows the datafiles in PRD, CP1 and CP2 after refresh.

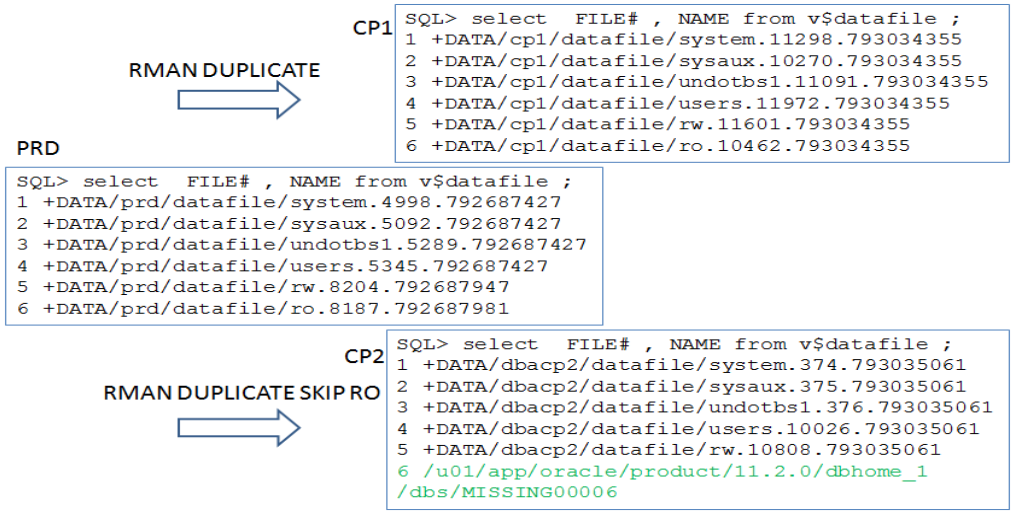


Figure 9. Datafiles in PRD, CP1 and CP2 after a Refresh (Improved)

The fact the file#6 is CP2 is missing does not change the universal method one bit. This is the code for sharing the RO tablespace for this scenario:

```
CP2:
SQL> alter tablespace ro offline;
Tablespace altered.
SQL> alter database rename file
'/u01/app/oracle/product/11.2.0/dbhome_1/dbs/MISSING00006' to
'+DATA/cp1/datafile/ro.10462.793034355';
Database altered.
SQL> alter tablespace ro online;
Tablespace altered.
SQL> select * from tst_user.ro_tab ;
Resides in RO tablespace
```

One major concern for sharing datafiles between different databases is the belief that since each datafile has a DBID, it can only be part of the database with that DBID. My Oracle Support document 1277854.1 (MOS, 2011 a) appears to share the sentiment:

"*DBID is an internal, uniquely generated number that differentiates databases.*
Oracle creates this number automatically when you create the database.
***It's used to identify the database a file belongs to**, and It's used in recovery operations to determine that a certain redo log/archived redo log actually belong to the database being recovered; Hence changing DBID requires opening the database with "resetlogs" option, and invalidates all previous archived logs.*"

My opinion is that while the DBID restrictions are valid for datafiles that belong to read-write tablespaces, they do not really apply for datafiles that belong to read-only tablespaces. Again, this is only an opinion. It is based on the following two observations.
First, sharing read-only tablespaces between databases using TTS is a documented and supported option. The shared datafile was never modified, so it has only one DBID. Since the datafile belongs to two different databases then the DBID could not have been used to identify which database that datafile belongs to.
Second, dumping the datafile headers using "`ALTER SESSION SET EVENTS 'immediate trace name file_hdrs level 10';`", a method documented in My Oracle Support document 218105.1 (MOS, 2011 b), indicates that read-only datafiles have different DBIDs than the database they belong to. Figure 10. Comparison of DBIDs (PRD, CP1 and CP2) shows the DBIDs of all datafiles in PRD, CP1 and CP2. CP1 and CP2 are refreshed from PRD using RMAN DUPLICATE. No other actions were performed on them. We can clearly see that datafile#6, the one that belongs to a read only tablespace, retained its DBID after a RMAN DUPLICATE, even though the DBID of the refreshed databases, CP1 and CP2, changed.

PRD (1864694135)	
ID	DBID
1	1864694135
2	1864694135
3	1864694135
4	1864694135
5	1864694135
6	1864694135

CP1 (1054141155)	
ID	DBID
1	1054141155
2	1054141155
3	1054141155
4	1054141155
5	1054141155
6	1864694135

CP2 (1701148445)	
ID	DBID
1	1701148445
2	1701148445
3	1701148445
4	1701148445
5	1701148445
6	1864694135

Figure 10. Comparison of DBIDs (PRD, CP1 and CP2)

Implementation Details and Tips

General Observations

None of the instances that mount the shared read-only tablespace should try to put it in read-write mode. This restriction applies to both the supported TTS method as well as the non-supported universal method. If the shared RO tablespace needs to be put in read-write mode then we first have to make a local copy of that tablespace that is not shared with any other instance, and then we can put that copy in read-write mode. Figure 11 illustrates that process. First, we copy the shared RO tablespace to CP2. Then, we point the CP2 controlfile to the newly created RO copy. Finally, we can put the local CP2 RO tablespace in read-write mode.

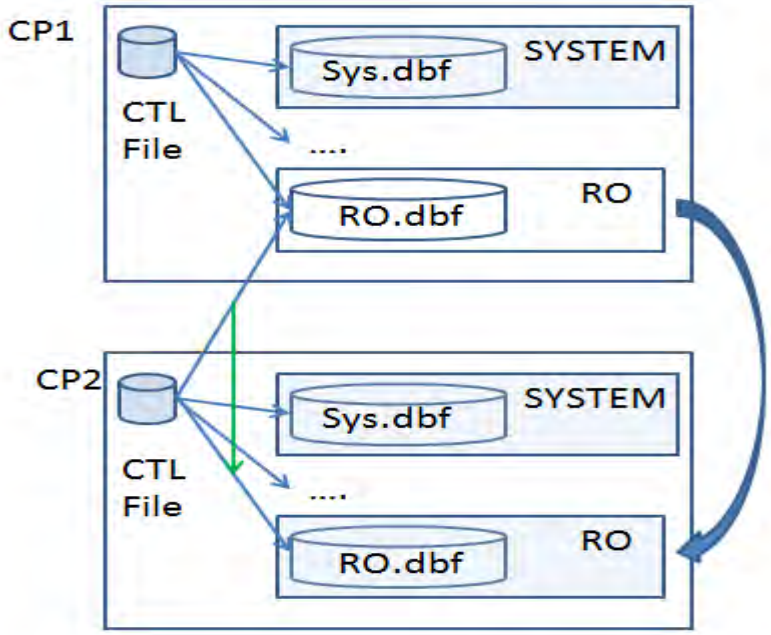


Figure 11. Putting a Shared RO Tablespace into Read-write Mode

The benefits of sharing read-only tablespaces are predicated on those tablespaces staying read-only. Do not share read-only tablespaces that you intent to make read-write, even for a moment, at any time in the future. If, for whatever reason, a shared read-only tablespace is to be temporarily converted to read-write, then the goal would be to reduce the frequency of such conversions. Finally, to minimize the impact of changing data originally slated as read-only, we can create multiple read-only tablespaces related to different business units. If data for a single business unit it to be updated, only the tablespace for that specific unit would be affected. Figure 12 illustrates that process. Initially, UAT, SIT and DEV databases all share copies (SCN 567843) of the read-only datafiles. Then, RO_U2 is switched to read-write and back to read-only. Now, RO_U2 has SCN 9998877, so the next DEV refresh would have to bring a copy of the newer version of RO_U2. A new SIT refresh would reuse the new copy of RO_U2 (SCN 9998877). We would need two copies of RO_U2 for non-production purposes for a while. Only when all non-production databases are refreshed, then we can remove the old version of RO_U2 (SCN 567843).

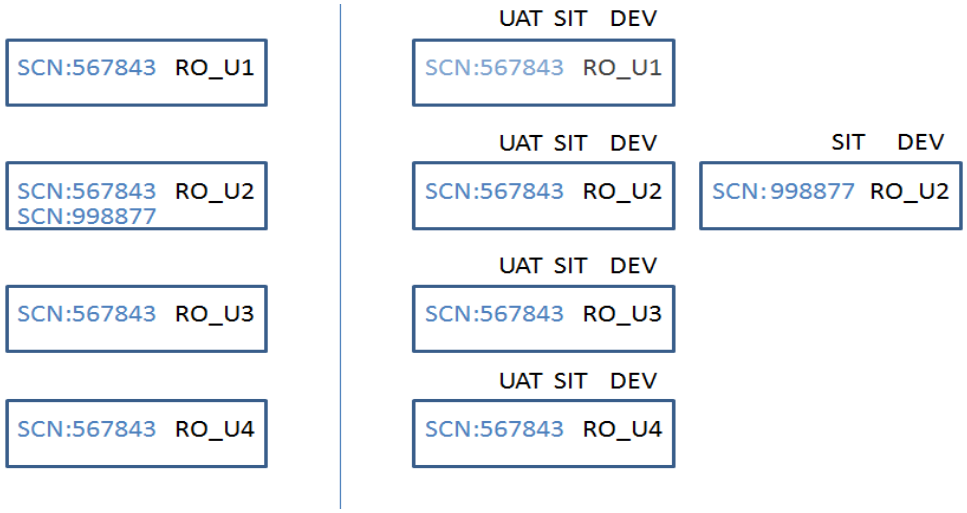


Figure 12. Impact of Temporary Switch to Read-write Mode

Universal Method Considerations

Even though we used RMAN DUPLICATE to illustrate the universal method throughout the paper, we could have also used RMAN RESTORE/RECOVER or a manual method to build the non-production databases. RMAN DUPLICATE is the preferred method though, not only because it is very simple, but also because it automatically changes the DBID of the refreshed database.

As explained earlier, skipping the read-only tablespace, when that tablespace is already in the non-production environment, is a great way to speed up the refresh. The problem is that if we use RMAN DUPLCIATE SKIP TABLESPACE, RMAN would try to drop the tablespace we requested to skip. Figure 13 illustrates that problem. If the tablespace is dropped, then we cannot use the universal method, because it requires the file slots in the controlfile to be available.

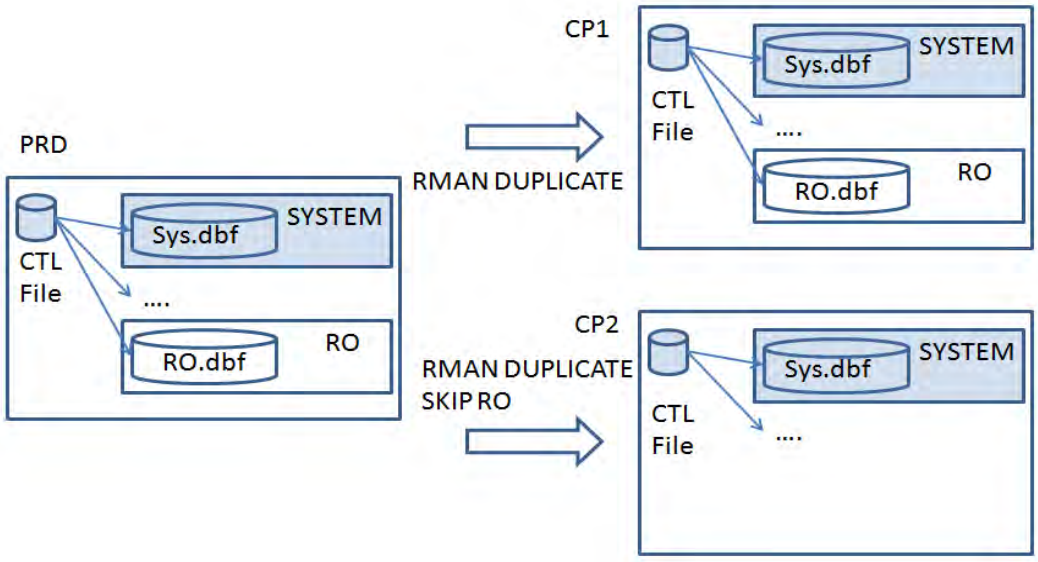


Figure 13. RO Tablespace Dropped after DUPLICATE SKIP

Fortunately, this problem has an easy workaround. If we create a partitioned table that spans the RO tablespace and a read-write tablespace, then RMAN would not be able to drop the RO tablespace. The workaround is shown in Figure 14.

```
CREATE TABLE HOOK_RO
(
  DUMMY NUMBER
)
PARTITION BY RANGE (DUMMY)
(
  PARTITION PART_RW VALUES
  LESS THAN (1000)
  TABLESPACE "RW"
, PARTITION PART_RO VALUES
  LESS THAN (2000)
  TABLESPACE "RO"
NOCOMPRESS
);
SQL> insert into HOOK_RO values (500);
SQL> insert into HOOK_RO values (1500);
```

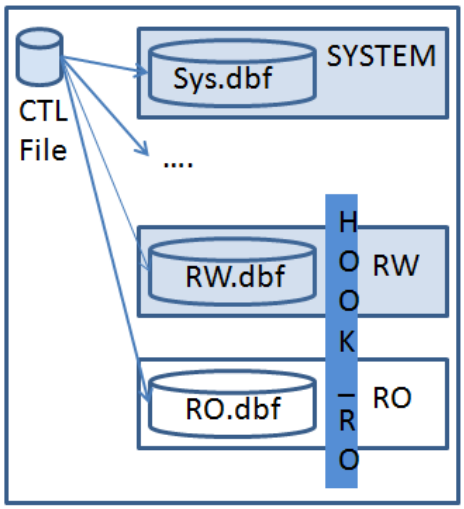


Figure 14. A Way to Guarantee That RO Tablespace Would Not Be Dropped after a Refresh

Starting in 11gR2, Oracle performs self-containment check before starting DUPLICATE, preventing us from skipping non self-contained tablespaces. A workaround for this problem can be found in My Oracle Support document 1355120.1.
Backup and recovery for databases that share tablespaces using the universal method could be a little bit more challenging. As mentioned previously, the universal method is not supported by Oracle at this time.
Databases created with RMAN DUPLICATE have no issues with backup. Since each database gets a new unique DBID upon refresh, there is no problem registering them with a single RMAN catalog. Databases created with RMAN RESTORE/RECOVER or with a manual method, carry over the DBID of the source DB. Changing it with NID is not possible, because NID requires all files to be writable. As a result, those databases should not be registered in RMAN catalogs, unless it is guaranteed that the DBID is unique within the catalog.
Restoring from a backup that belongs to a database that shared tablespaces using the universal method may require some special handling. Please note that the universal method is not supported by Oracle at this time. All steps until RMAN RECOVER are the same as with a regular restore. The RO tablespace that was shared should be skipped when RMAN RECOVER is issued:

```
RMAN> RUN
 {
 set until time "to_date(…)";
 recover database skip tablespace ro;
 }
```

After that, if needed, the controlfile should be updated with the valid path of the shared RO datafile:

```
SQL> alter tablespace ro offline;
alter database rename file '<CTL_FILE_LOCATION>' to
'<ACTUAL_LOCATION>'
SQL> alter tablespace ro online;
```

RMAN DUPLICATE works OK against a database that shares tablespaces using the universal method as long as a connection to the target database is made. RMAN DUPLICATE using backup location only does not work in this case.

Summary

There are many ways to reduce the disk footprint of a database-centered enterprise. Sharing in read-only tablespaces among non-production copies is a viable approach for saving disk space. While the supported transportable tablespace method for sharing read-only tablespaces can be applied in some cases, the non-supported universal method presented here can greatly expand the scenarios where sharing data, and consequently saving disk space, is possible.

References

Delphix ® (n.d. a) Retrieved from http://www.delphix.com/products/how-it-works/

Delphix ® (n.d. b) Retrieved from http://www.delphix.com/products/overview/

My Oracle Support /MOS/ (2011 a) Difference between DB name, DB global name, DBID and SID [ID 1277854.1] Retrieved from https://support.oracle.com/

My Oracle Support /MOS/ (2011 b) Introduction to ORACLE Diagnostic EVENTS [ID 218105.1] Retrieved from https://support.oracle.com/

My Oracle Support /MOS/ (2011 c) RMAN-05548 # skip self-contained check in rman duplicate [ID 1355120.1] Retrieved from https://support.oracle.com/

Oracle-Base (n.d) Retrieved from http://www.oracle-base.com/articles/11g/clonedb-11gr2.php

Oracle Documentation (n.d. a) Retrieved from
http://docs.oracle.com/cd/B12037_01/server.101/b10739/tspaces.htm#i1007595

Oracle Documentation (n.d. b) Retrieved from
http://docs.oracle.com/cd/B28359_01/server.111/b28310/tspaces013.htm#autoId6

ORACLE ACCELERATION

YEAH, IT'S KIND OF LIKE THAT FAST!

- Accelerate Oracle databases up to 10x
- Deploy with minimal disruption to operations
- Extend the life of your IT infrastructure

Put Your Big Data on the Fast Track.

The GridIron Systems TurboCharger™ data acceleration appliance seamlessly integrates into your existing IT environment without changes to applications, databases, servers, storage or operational processes.

Learn more at **www.gridironsystems.com/oracle**.

GRIDIRON
Big Data. Fast

Integrating XML Using Oracle SQL Developer 3.1 and Oracle Database 11g Release 2

Coleman Leviter, Arrow Electronics

For the past several years, we have been involved in a wide range of development projects using Oracle's XML DB. The types of projects include integrating a Transportation Management System[1] (TMS), transmitting an XML Manifest Document using a B2B[2] portal, and finally, communicating with Oracle's E-Business Suite[3] (EBS). The focal point of the aforementioned three projects is a Warehouse Management System[4] (WMS).
In spite of the diversity of the three projects, all are in production and for the most part, perform flawlessly. With any new technology stack, at times issues arise and alternate solutions must be developed or you find that you are unable to proceed. Consequently, the project falls behind schedule.
Some examples of issues and their resolution:

- We encountered an issue developing XML Document communications to EBS. Our Oracle XML DB technology stack was based upon Oracle 10.2.0.1.0. The EBS developers were using ***Oracle (***on 10.2.0.4***) XQuery*** a***nd requested that we (WMS) sync up with their technology stack. We ran into issues (***ORA-19114: error during parsing the XQuery expression) using XQuery. Consequently, we abandoned XQuery and used XPath for XML Document shredding. EBS continues to use 10.2.0.4. There are no current issues with communications.
- A bug was encountered using the DBMS_XMLGEN.CONVERT[5]. DBMS_XMLGEN.CONVERT is used to convert an escaped version to an unescaped version (or visa versa). As an example, the escaped form of the character ">" (without the " characters) is ">" (without the " characters). Oracle's suggested "workaround" was to 1) append a space (CHR(32) to the end of the XPath string, then 2) DBMS_XMLGEN.CONVERT the string, finally 3) TRIM the string. The workaround resolved the issue.

With the projects mentioned above as a starting point, we would like to delve into the following XML topics: XML basics, XML document construction, XML message efficiency vs. Binary data messages, XSD[6] development, Warehouse Management System (WMS) communications to Oracle's Service Oriented architecture (SOA), XML namespace and finally dbms_xmldom.

Introduction

XML DB was introduced in Oracle 8i. In Oracle 11g, XML DB reached a new maturity level, providing high-performance with native XML storage and retrieval technology. The W3C[7] XML data model is fully immersed into the Oracle Database.
How is XML DB used in a project? What must one know when using XML DB? In this discussion, we will address these issues and the design criteria one might consider when using XML in a project.
XML or Extensible Markup Language means the language can grow as required. You may include an XML declaration line for the first line (`<?xml version="1.0"?>`). You may define your own elements or tag fields in the body of an XML document (<tagfield>data</tagfield>). As long as the sender and receiver agree on the format of an XML Document, there is complete flexibility for its construct.
Throughout this paper, we will present several examples of XML constructs on XML documents. We hope the readers will become familiar with XML by reviewing the examples, modifying them and using them in their own database, whether that is 9i, 10g or 11g. Some examples demonstrate different capabilities of SQLX (SQL XML). The WEB contains a great amount of XML material. For those beginning an XML project, this establishes a good starting point. Finally, the last section presents the reader with an example using SQLX (XML Document creation) and XML Document shredding (XML Xpath). Many examples demonstrate the use of XML Documents with namespace. The examples are presented using table reference as well as in line code, which the reader may simply copy and paste in their own environment and view the results. The examples are constructed so that the reader may use them in their own Oracle database.

Project Overview

The current method of communications between the mainframe computer and the WMS are fixed length messages. Recently, the company embarked on a project retiring the mainframe computer and replacing it with Oracle's eBusiness suite (EBS). The inter-computer method of communications is XML. Therefore, the project uses Oracle 10g XML for communications between the WMS and Oracle's EBS. The WMS' pathway into eBusiness Suite is through the Service Oriented Architecture (SOA).

Using XML Messaging between EBS and WMS will accommodate a changing business model. Changing business rules may easily be incorporated into XML Messaging.

Presented is the WMS-EBS/SOA communications data flow:

WMS Appl. Step 1)

1. WMS Appl. issues an XML Request Message (XML Document) to MQ Write Queue
2. MQ Responds with a Successful Status indicating guaranteed XML Message delivery
3. EBS/SOA receives XML Request Message and processes the message

EBS/SOA Step 1)

1. EBS/SOA issues an XML Request Msg (XML Document) to MQ Read Queue
2. WMS Appl. reads XML Request Msg (XML Document) from MQ Read Queue
3. WMS Appl. Responds to MQ with Successful MQ Status indicating receipt of XML Request Msg

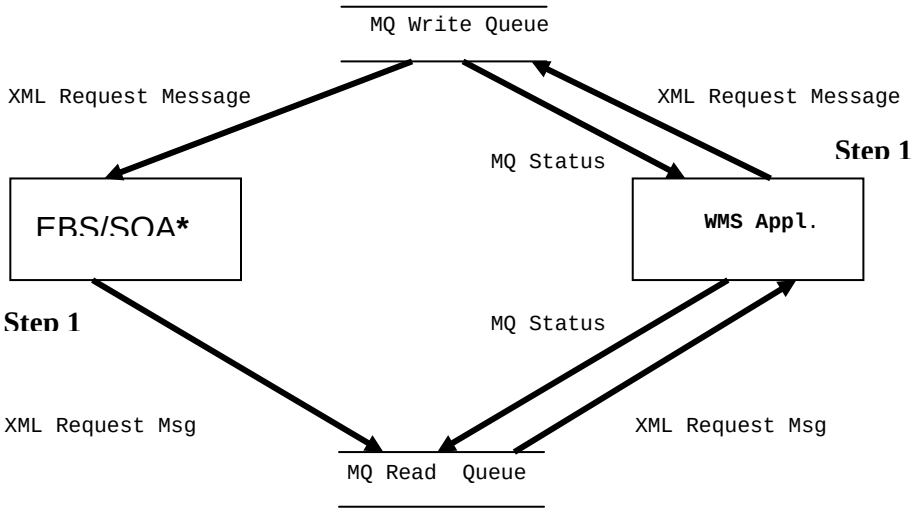


Figure 1 – EBS/SOA - WMSAppl. XML Document Data Flow

* Note: SOA is not used here. It has been replaced by DTI or Direct Memory Interface

Message Comparison

Why use XML for messaging? To answer that question, let us explore an alternate method of data transfer.

One predominant method of transferring messages between two computers (or processes) is Binary Messaging. This method continues to prevail with many legacy computers and embedded firmware systems.

Data may be compressed, so encoding and reassembly must all work in concert. If a transmitted data steam contains 256 bits, the receiving side must also be in alignment and decode those same 256 bits. All data is position dependent.

When implementing this method, maintenance costs may run very high because of the volume of software that must be managed on the sender and receiver sides. Troubleshooting adds to the cost as well.

Describing a system's efficiency is the ratio of output to input. Therefore, the efficiency of messages between two computers is shown as:

$$\text{Efficiency (\%)} = \frac{\text{Output}}{\text{Input}} \times 100 \quad \text{or} \quad \frac{\text{Message Data}}{\text{Message Data + Overhead Data}} \times 100$$

Sales Complete Message (SCP)

Field Name	Field Type	Size	Bit Position
Overhead Data	Alphanumeric	32	1 - 32
Message Code	'SCP* '	4	33-36
Entering Location	Alphanumeric	3	37-39
Sales Number	Alphanumeric	6	40-45
Version Number	Alphanumeric	2	46-47
Cartons for Shipping	Numeric	2	48-49
Shipping Charges	Alphanumeric	9/2	50-58
Date (YYMMDD)	Alphanumeric	6	59-64
Carrier	Alphanumeric	16	65-80
End	Alphanumeric	27	81-107

Table 1 – Typical binary interface data layout

Using the sample binary data layout from Table 1 we have:

```
75 BITS (POS 33- 107)
                          ----------------------                                X 100 =
70%  EFFICIENT
     (MESSAGE DATA) 75 BITS + 32 BITS (OVERHEAD DATA)
```

Let us look at a simple XML Document using namespace (437 bytes):

```
<ns1:object_group xmlns:ns1="http://www.w3.org/2001/XMLSchema/sample_namespace_1"
xmlns:obj1="http://www.w3.org/2001/XMLSchema/obj_1">
  <obj1:object>
    <obj1:thing>ball</obj1:thing>
    <obj1:thing>key</obj1:thing>
    <obj1:thing>table</obj1:thing>
  </obj1:object> ^------------------------  typical data
  <obj2:object xmlns:obj2="http://www.w3.org/2001/XMLSchema/obj_2">
    <obj2:thing>frisbee</obj2:thing>
    <obj2:thing>bbq</obj2:thing>
    <obj2:thing>switch</obj2:thing>
  </obj2:object>
</ns1:object_group>
```

Figure 2 - Simple XML Message

Here, the meaningful data consists of "ball", "key", "table", "frisbee", "bbq" and "switch" or 28 bytes. The overhead data is everything else or 437 bytes - 28 bytes = 409 bytes of metadata. (We address a bit more on metadata later on.)

Using the Efficiency Formula from above we have:

```
  28 BYTES MESSAGE DATA (FIGURE 3)
-------------------------------------------        X 100
  28 BYTES MESSAGE DATA  + 409 BYTES XML ELEMENT DEFINITION & NAMESPACE (= 437 TOTAL
BYTES)
```

= 6 % EFFICIENT

Therefore, using an XML data message with namespace results in 6% efficiency, while using binary data transfer results in 70% efficiency. It is obvious that the messages are quite different, but an important observation is the overhead ratio required to send data. XML messages far exceed binary data messages, mainly due to the verbose nature or metadata of the element names and namespaces in the XML Document.
If the efficiency of an XML data message does not fare well compared to a binary data message, then why use XML data messaging? Here are arguments for and against:

XML Arguments for Usage:

- It is platform and system independent i.e. it can work on any computer.
- It allows us to define our own tags thus making your data content understood.
- XML has adopted a standard, ISO 10646 also known as Unicode, which is a framework to encode characters. It will support most languages, thus not forcing people to use English for coding.
- Software can be developed to increase efficiency, that is, encode the element tags on the transmission as well on the receiving side. This must be balanced between easily understood tags and tags that are hard to follow.
- The code is easy to understand even for those people who do not have any prior knowledge.
- XML DB has been available with Oracle 8i and up
- XML is self-describing. For example it is obvious from <lastname>Doe</lastname> that this represents a name.
- Bandwidth issues can become non existent using data compression techniques

XML arguments against usage:

- It requires a wide amount of bandwidth.
- It may require extensive processing time to decode. The host computers must be capable of processing the messages.
- Only newer software will be able to read and understand XML. It may become costly and time consuming to retrofit legacy code with XML

A good design decision rests with one's ability to analyze a problem and choose the proper tools. The same applies when selecting XML or other data communications methods. In the previously described application, the new EBS/SOA system already used XML. Our system (WMS) used Oracle 10g in which XML DB was available. In our case, it was a perfect fit to use XML for data communications. Although the XML duty cycle is low, if the messaging frequency is low, XML is certainly a viable option. For example, if the application environment is limited to human interaction with a computer and waiting for an answer (i.e. credit card verification), XML inefficiencies would not appear to present a problem. If XML messages were used to guide a shuttlecraft into its docking station, perhaps too many messages would rapidly use up the bandwidth. Another example where XML data transfer might not work: a central station (monitoring burglar alarms) receiving video transmissions of an alarm of a potential intruder. With samplings of several seconds and several frames of a possible intruder in the area, data transmission with compression may exceed 250 Kbytes. The real time video example may not be a candidate for XML unless binary compression techniques become available.
To summarize, many real time or mission critical applications may not be candidates for XML data communications. But there are many applications that are using XML for data communications. It is important to assess the project requirements and use the proper architecture. That will ensure project success.

XML Communications

Our message queuing system for XML communications is IBM's WebSphere MQ Message Queuing, which is in use with many enterprise applications.
When we view XML Messages on MQ, we use a tool called IBM Tivoli CandleNet Portal. Here is an extract of a sample message using the tool:

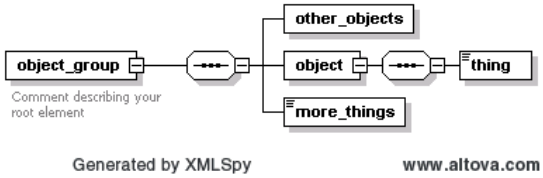
```
Hexadecimal Data              Character Data
3C3F786D 6C207665 7273696F 6E3D2231 *<?xml version=1*"
2E302220 3F3E3C69 6D70313A 574D535F *.0 ?><imp1:WMS_*"
4D51456E 76656C6F 70652078 6D6C6E73 *MQEnvelope xmlns*
3A696D70 313D2268 7474703A 2F2F7777 *:imp1=http://ww*"
772E6172 726F772E 636F6D2F 574D535F *w.arrow.com/WMS_*
4D51456E 76656C6F 70655F76 315F305F *MQEnvelope_v1_0_*
3030223E 0A202020 3C696E70 313A574D *00>.   <inp1:WM*"
535F5072 6F636573 73496E76 656E746F *S_ProcessInvento*
72794D6F 76656D65 6E742078 6D6C6E73 *ryMovement xmlns*
3A696E70 313D2268 7474703A 2F2F7777 *:inp1=http://ww*"
772E6172 726F772E 636F6D2F 574D532F *w.arrow.com/WMS/*
50726F63 65737349 6E76656E 746F7279 *ProcessInventory*
4D6F7665 6D656E74 5F76315F 305F3030 *Movement_v1_0_00*
223E0A20 20202020 203C6E73 313A5374 *>.      <ns1:St*"
616E6461 72644865 61646572 20786D6C *andardHeader xml*
6E733A6E 73313D22 68747470 3A2F2F77 *ns:ns1=http://w*"
77772E61 72726F77 2E636F6D 2F574D53 *ww.arrow.com/WMS*
2F537461 6E646172 64486561 6465725F */StandardHeader_*
76315F30 5F303022 3E0A2020 20202020 *v1_0_00>.      *"
```

Figure 3 - Partial XML Document Viewed Using IBM Tivoli CandleNet Portal

The left side of Figure 4 shows the XML message in hexadecimal; the right side shows the XML message. The asterisks are not part of the data. Viewing XML data in this manner aids in isolating problems when the XML message is on its way to the receiver or arriving from sender. Additionally, observe that the namespace definition is part of the payload.

Background: XML Schema Definition (XSD)

An enterprise project may begin with an XML Schema Definition or XSD. The XSD can be used to express a set of rules to which an XML document must conform in order to be considered 'valid' according to that schema. There are several graphical user interface (GUI) tools on the market that allow one to design an XSD. Displayed in Figure 1 below is an example of an XSD using a GUI tool. The schema is designed as well as the XML Document data types.



After the XSD is designed, one may generate the associated XML that describes the XSD:

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XMLSpy v2008 rel. 2 sp1 (http://www.altova.com)  -->
<xs:schema xmlns=" =" ="http://www.w3.org/2001/XMLSchema/sample_namespace_1 "
```

```
xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
<xs:element name="object_group">
            <xs:annotation>
                  <xs:documentation>Comment describing your root
element</xs:documentation>
            </xs:annotation>
            <xs:complexType>
                  <xs:sequence>
                        <xs:element name="other_objects">
                              <xs:complexType/>
                        </xs:element>
                        <xs:element name="object">
                              <xs:complexType>
                                    <xs:sequence>
                                          <xs:element name="thing"/>
                                    </xs:sequence>
                              </xs:complexType>
                        </xs:element>
                        <xs:element name="more_things"/>
                  </xs:sequence>
            </xs:complexType>
      </xs:element>
</xs:schema
```

Using the above XML Schema Definition, one may use it as a template to generate the XML Document. Note the "ns:" or namespace notion. We will present several examples using this notation throughout this paper.

XML Document Construction

Properly formed XML documents contain (except where noted) the following components:

- An optional declaration as the first line (generally, it is good practice to include the declaration, but it is not mandatory):

```
<?XML VERSION="1.0" ENCODING="UTF-8"?>
```

  where the mandatory first attribute identifies the XML version number and the optional second attribute is the encoding attribute which specifies to the XML parser what character encoding the text is in for translation into Unicode (Unicode is an industry standard allowing computers to consistently represent and manipulate text expressed in any of the world's writing systems).

- An optional comments section:

```
 <!--THIS IS A SAMPLE COMMENT-->
```

  A mandatory start tag and end tag as the root node:

START TAG

```
<ns1:object_group xmlns:ns1=http://www.w3.org/2001/XMLSchema/sample_namespace_1
        xmlns:obj1="http://www.w3.org/2001/XMLSchema/obj_1">
```

END TAG

```
</ns1:object_group>
```

And finally, a mandatory data element:

```
Data Element
<obj1:thing>ball</obj1:thing>
```

So, at a minimum, one root node (start tag and end tag) and one data element constitute a properly formed XML document.
Putting it all together we have:

```
<ns1:object_group xmlns:ns1=http://www.w3.org/2001/XMLSchema/sample_namespace_1
                  xmlns:obj1="http://www.w3.org/2001/XMLSchema/obj_1">
      <obj1:thing>ball</obj1:thing>
</ns1:object_group>
```

A more meaningful XML Document follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<ns1:object_group xmlns:ns1=http://www.w3.org/2001/XMLSchema/sample_namespace_1
xmlns:obj1="http://www.w3.org/2001/XMLSchema/obj_1">
  <!—- XML Document with namespace example-->
  <obj1:object>
    <obj1:thing>ball</obj1:thing>
    <obj1:thing>key</obj1:thing>
    <obj1:thing>table</obj1:thing>
  </obj1:object>
  <obj2:object xmlns:obj2="http://www.w3.org/2001/XMLSchema/obj_2">
    <obj2:thing>frisbee</obj2:thing>
    <obj2:thing>bbq</obj2:thing>
    <obj2:thing>switch</obj2:thing>
  </obj2:object>
</ns1:object_group>
```

Shortly, we will explain shortly the following syntax: “xmlns:ns1”, “xmlns:obj1”, “obj1”, “obj2”.
The XML document fits a hierarchical model as presented in Figure 4:

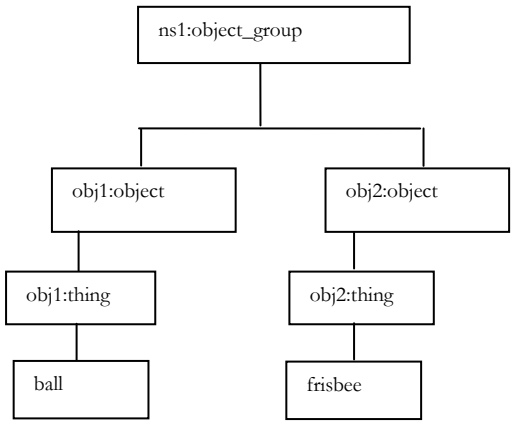


Figure 4 - Hierarchical Model of an XML Document

where ns1:object_group is the root element and obj1:object and obj2:object represent the data sections. Note: data items “key”, “table”, “bbq” and “switch” have been omitted from Figure 5 but are part of obj1:object and obj2:object.

XMLTYPE Column

To create a sample table containing an XMLTYPE column, use the following CREATE TABLE DDL:

```
CREATE TABLE MY_TABLE (ID NUMBER, XMLCOL XMLTYPE, CLOBCOL CLOB);
```

The underlying type, XMLTYPE is a CLOB, which enables storage of up to 4GB of data. Additionally, you may perform XPath queries on the XML Documents residing in the column. By simply defining the column (xmlcol) as a CLOB, XPath expression queries are not possible. When storing XML documents into an XMLTYPE column, Oracle will raise an exception if the XML document is not properly formed. If you want to store the improperly formed document for later evaluation, depending upon its length, it may be stored in a CLOB type column, which in the above sample table, is identified by "clobcol".
When using XMLTYPE columns in tables, it may be helpful to use the pragma AUTONOMOUS_TRANSACTION. When a subprogram is marked with this pragma, it is possible to perform rollbacks or commits without affecting operations in the parent transaction. Basically, this pragma works the same way as a sequence object.

XML Examples

In this section we will explore several XPath and SQLX (or SQL/XML) examples. SQLX is used for constructing XML documents. XPath is used for shredding or extracting data from the XML document. You may use the code in the following examples in your own Oracle environment. All my examples work in Oracle 10g Enterprise Edition Release 10.2.0.1.0. I cannot guarantee the results if you use an earlier Oracle edition, especially with namespaces. XPath or XML Path Language is a language for selecting parts (shredding) of an XML document and computing values (strings, numbers, or Boolean values) based on the content of an XML document. We will produce (construct) several XML fragments using functions XMLElement(), XMLAgg() and XMLForest(). We will demonstrate document shredding using the XMLSequence() function and the EXTRACT method. At the end of each respective section, the SQLX and XPath section, the reader will view other XML examples using namespace notion. To wrap it up, we will present two different types of SQLX and XPath examples: one that is using XML documents in-line, the other using Oracle table references. Finally, at the end, we provide one example using the Oracle function UPDATEXML() demonstrating an update to a single data element residing in an XMLTYPE column.

SQLX Document Construction

The following examples presented in this section, show how a few XML functions are used as building blocks to build complex XML documents.
Some of the examples reference the following relational table:

```
SQL>  select * from myobject;

THINGS                 QUANTITY         PARENT
------------           -----------------  ------------
BALL                   2                1
KEY                    3                1
TABLE                  1                1
FRISBEE                4                2
BBQ                    1                2
SWITCH                 6                2
```

Table 2 – "myobject" Relational Table

XMLELEMENT()

The simplest SQLX Query uses the XMLElement() function, which returns an XMLTYPE expression (XML fragment):

First, we will view the results of the query using SQL:

```
SELECT  'jones' EMPLOYEE FROM dual;

EMPLOYEE
---------------
jones
```

Now, let us view the SQLX version of the query:

```
SELECT XMLELEMENT("Emp", 'jones')  employee FROM DUAL;

EMPLOYEE
----------------
<Emp>jones</Emp>
```

Oracle's SQLX provides the tag fields <Emp></Emp> for the query results. The "/" notation in the last tag field signifies the element terminator.

XMLForest()

This example, which uses XMLForest(), produces a non qualified XML document: (the reason that it is not qualified is that the result set is missing the parent node):
First, we will view the results of the query using SQL:

```
select  quantity, things from myobject;

QUANTITY THINGS
-------- --------------------
2        BALL
3        KEY
1        TABLE
4        FRISBEE
1        BBQ
6        SWITCH
```

Now, let us view the SQLX version of the query:

```
SELECT XMLFOREST(obj.quantity, obj.things) "Object_list"
FROM myobject obj;

Object_list
--------------------
<QUANTITY>2</QUANTITY><THINGS>BALL</THINGS>
<QUANTITY>3</QUANTITY><THINGS>KEY</THINGS>
<QUANTITY>1</QUANTITY><THINGS>TABLE</THINGS>
<QUANTITY>4</QUANTITY><THINGS>FRISBEE</THINGS>
<QUANTITY>1</QUANTITY><THINGS>BBQ</THINGS>
<QUANTITY>6</QUANTITY><THINGS>SWITCH</THINGS>
```

The XMLForest() function produces an XML fragment that contains a set of XML elements.

XMLAGG()

Let us view an example using the XMLAgg() function. First, we will view the results of the query using SQL:

```
select things, quantity from myobject;

THINGS               QUANTITY
-------------------- --------
BALL                 2
KEY                  3
TABLE                1
FRISBEE              4
BBQ                  1
SWITCH               6
```

The first example returns an ordered set using the XMLAgg() function:

```
SELECT XMLELEMENT("OBJECT", XMLAGG(XMLELEMENT("Things",obj.things ||' '||
obj.quantity )
ORDER BY obj.things)) AS "Object_list"
FROM myobject obj;

Object_list
--------------------
<OBJECT>
<Things>BALL 2</Things>
<Things>BBQ 1</Things>
<Things>FRISBEE 4</Things>
<Things>KEY 3</Things>
<Things>SWITCH 6</Things>
<Things>TABLE 1</Things>
</OBJECT>
```

The XMLAgg() function returns an XML fragment in an XMLTYPE by assembling XML fragments, with the option of XML element sorting. The XMLAgg() function assembles all the XML elements into one XML document fragment. The outer XMLElement() function, incorporates the XML document fragment into its "OBJECT" element (parent) as child elements. As a result of using the XMLELEMENT() function, we have essentially created a fully qualified XML Document, one that has a parent node and child nodes.

XMLELEMENT - Namespace

This is an SQLX Query (anonymous block) namespace example using XMLELEMENT(), XMLAGG(), XMLTYPE.getclobval() and XMLATTRIBUTES() . In our example, XMLATTRIBUTES() is used to define the namespaces associated with the elements.
(reference Table 2 – "myobject" relational table)

```
DECLARE
lcl_obj1        CLOB;
lcl_obj2        CLOB;
lcl_full_xml    CLOB;

BEGIN
   SELECT XMLTYPE.getclobval(XMLELEMENT("obj1:object",
```

```
                xmlagg(xmlelement("obj1:thing",obj.things) ))  )
   INTO lcl_obj1
   FROM  myobject obj
   WHERE obj.parent = '1';

   SELECT XMLTYPE.getclobval(XMLELEMENT("obj2:object",
        XMLATTRIBUTES ('http://www.w3.org/2001/XMLSchema/obj_2' AS "xmlns:obj2"),
        XMLAGG(xmlelement("obj2:thing",obj.things) ) ) )
   INTO lcl_obj2
   FROM  myobject obj
   WHERE obj.parent = '2';

   SELECT ( '<?xml version="1.0" encoding="UTF-8"?>' ||
       '<ns1:object_group
            xmlns:ns1="http://www.w3.org/2001/XMLSchema/sample_namespace_1"
            xmlns:obj1="http://www.w3.org/2001/XMLSchema/obj_1">'||
      lcl_obj1|| lcl_obj2|| '</ns1:object_group>' )
   INTO lcl_full_xml
   FROM dual;
   dbms_output.put_line(lcl_full_xml);
END;
```

The results are:

```
<?xml version="1.0" encoding="UTF-8"?>
<ns1:object_group         xmlns:ns1=http://www.w3.org/2001/XMLSchema/sample_namespace_1
                  xmlns:obj1="http://www.w3.org/2001/XMLSchema/obj_1">
<obj1:object>
          <obj1:thing>ball</obj1:thing>
            <obj1:thing>key</obj1:thing>
                  <obj1:thing>table</obj1:thing>
</obj1:object>
<obj2:object xmlns:obj2="http://www.w3.org/2001/XMLSchema/obj_2">
<obj2:thing>frisbee</obj2:thing>
<obj2:thing>bbq</obj2:thing>
<obj2:thing>switch</obj2:thing>
</obj2:object>
</ns1:object_group>
```

Note: The above example demonstrates the flexibility of constructing an XML document. As in pl/sql, you may construct almost any type of query. In the above example, we have constructed an XML document using several different queries and finally constructing the XML document by concatenating the individual sections.

XPATH Document Shredding

XPath expressions are used to shred XML Documents. Shredding an XML document enables you to store data elements into a relational database.

XMLSEQUENCE()

The XMLSequence() function returns a collection of XMLTYPE. This function in a TABLE clause can be used to decompose the collection values into multiple rows. This can be further processed in a standard SQL query.

==========
Example: XML Document

```
<objects>
  <thing>ball</thing>
  <thing>key</thing>
  <thing>table</thing>
</objects>
```

XPath Query

```
SELECT  value(tab).extract('/*').getStringVal() "This Column"
FROM  table ( XMLSequence(extract ( XMLTYPE('
<objects>
  <thing>ball</thing>
  <thing>key</thing>
  <thing>table</thing>
</objects>
'),'/objects/*')  )   ) tab;
```

Result Set

```
This Column
--------------------
<thing>ball</thing>
<thing>key</thing>
<thing>table</thing>
```

The format clause ('/objects/*') indicates which child to return. Here, we request all the children under the 'object' parent. The "extract('/*')" clause in the SELECT statement requests everything from the "format" clause.

==========

Example: XML Document Deux

```
<objects>
  <thing>ball</thing>
  <thing>key</thing>   ← extract this data item
  <thing>table</thing>
</objects>
```

==========

XPath Query

```
SELECT VALUE(tab).extract('/objects/thing[2]/text()').getStringVal() "This Column"
FROM   TABLE ( XMLSequence(extract (XMLTYPE('
<objects>
  <thing>ball</thing>
  <thing>key</thing>
  <thing>table</thing></objects>
'),'*')    )   ) tab;
```

Result Set

This Column
----
key

The format clause ('*') indicates return everything. The "extract('/objects/thing[2]/text()')" method in the select statement requests return the data from the second node or 'key'. The next example demonstrates, though the use of a cursor, how to traverse the nodes to obtain all the data items.

PUTTING IT ALL TOGETHER – Shredding Example with Cursor, No Table, Namespace

(Note: this is a self contained example. An Oracle table is not used. Namespace is introduced in this example, which is presented with the "xmlns:" notation. Except for the namespace notion denoted by the prefix on each element and the namespace definition, the focus of this example is the use of a cursor to traverse the nodes to obtain each data item from its respective element.
The namespace declaration uses the following syntax. xmlns:prefix="URI". "URI" or Uniform Resource Identifier can be any Internet resource or simply a string. Its purpose is to distinguish two elements with the same name. Namespace notation is optionally the last argument in the EXTRACT method:

extract (XMLTYPE_Instance>, <XPath_string>, <namespace_string>)

```
-- Anonymous Block
DECLARE
      -- Cursor for parsing object_group XML
      CURSOR obj_cur
      IS  SELECT EXTRACT (VALUE (entire_things),'//obj1:thing/text()',
            'xmlns:obj1="http://www.w3.org/2001/XMLSchema/obj_1"').getstringval()   AS lcl_thing
      FROM   table ( XMLSequence(extract  (XMLTYPE('<?xml version="1.0" encoding="UTF-8"?>
<ns1:object_group  xmlns:ns1="http://www.w3.org/2001/XMLSchema/sample_namespace_1"
   xmlns:obj1="http://www.w3.org/2001/XMLSchema/obj_1">
<obj1:object>
<obj1:thing>ball</obj1:thing>
<obj1:thing>key</obj1:thing>
<obj1:thing>table</obj1:thing>
</obj1:object>
<obj2:object xmlns:obj2="http://www.w3.org/2001/XMLSchema/obj_2">
                    <obj2:thing>frisbee</obj2:thing>
<obj2:thing>bbq</obj2:thing>
          <obj2:thing>switch</obj2:thing>
</obj2:object>
       </ns1:object_group>'),'//obj1:thing','xmlns:obj1="http://www.w3.org/2001/XMLSchema/obj_1"' ) ) )
                     entire_things;
 BEGIN
          FOR obj_row IN obj_cur LOOP
dbms_output.put_line('each element thing  '|| obj_row.lcl_thing );
          END LOOP;
 END anonymous_block ;
```

Result Set

```
SQL> set serveroutput on
SQL> /
```

each element thing ball
each element thing key
each element thing table
PL/SQL procedure successfully completed.

Note: Once the data items are "shred" from the XML Document, they can be stored into a relational table.

UPDATEXML() Example

We thought the reader might be interested in the following example. It does not fall within the realm of XML Document Shredding or XML Document Query. It is used to update a data element within an XML Document. We will demonstrate the Oracle function UPDATEXML. We will be "upper casing" the data item "frisbee".
Consider the following table:

SQL> describe my_xml_table;

Name	Null?	Type
REF_ID		NUMBER
XMLCOL		XMLTYPE

select xmlcol from my_xml_table where ref_id = 1;

```
XMLCOL
--------------------------------------------------------------------------------
<?xml version="1.0" encoding="UTF-8"?>
<ns1:object_group  xmlns:ns1=http://www.w3.org/2001/XMLSchema/sample_namespace_1
                xmlns:obj1="http://www.w3.org/2001/XMLSchema/obj_1">
      <obj1:object>
            <obj1:thing>ball</obj1:thing>
            <obj1:thing>key</obj1:thing>
            <obj1:thing>table</obj1:thing>
      </obj1:object>
      <obj2:object xmlns:obj2="http://www.w3.org/2001/XMLSchema/obj_2">
            <obj2:thing>frisbee</obj2:thing>
            <obj2:thing>bbq</obj2:thing>
            <obj2:thing>switch</obj2:thing>
      </obj2:object>
</ns1:object_group>


SQL> UPDATE my_xml_table mxt
  2  SET mxt.xmlcol = UPDATEXML(mxt.xmlcol,
  3  '//obj2:thing[1]/text()','FRISBEE',
     'xmlns:obj2="http://www.w3.org/2001/XMLSchema/obj_2"')
  4  WHERE mxt.ref_id = 1;

1 row updated.
```

select xmlcol from my_xml_table where ref_id = 1;

```
XMLCOL
--------------------------------------------------------------------------------
<?xml version="1.0" encoding="UTF-8"?>
<ns1:object_group xmlns:ns1="http://www.w3.org/2001/XMLSchema/sample_namespace_1"
```

```
xmlns:obj1="http://www.w3.org/2001/XMLSchema/obj_1">
  <obj1:object>
    <obj1:thing>ball</obj1:thing>
    <obj1:thing>key</obj1:thing>
    <obj1:thing>table</obj1:thing>
  </obj1:object>
  <obj2:object xmlns:obj2="http://www.w3.org/2001/XMLSchema/obj_2">
    <obj2:thing>FRISBEE</obj2:thing>
    <obj2:thing>bbq</obj2:thing>
    <obj2:thing>switch</obj2:thing>
  </obj2:object>
</ns1:object_group>
```

Back to MetaData

As more projects utilize XML, we would like to add a final comment about metadata. We can present the same XML Document in two different ways:

1. Parent-child nodes containing the intelligence (for brevity, we have omitted several "room" nodes).

```
<?xml version="1.0" encoding="UTF-8"?>
<StandardHeader>
  <DateTime>12/08/2010 16:03:46</DateTime>
  <PacketNumber>1</PacketNumber>
  <AnyMorePackets>Y</AnyMorePackets>
  <TotalPackets>4</TotalPackets>
  <house>
    <room>
      <room_name name="kitchen"/>
      <room_item item="sink"/>
    </room>
            |
            |
            |
    <room>
      <room_name name="kitchen"/>
      <room_item item="table"/>
    </room>
<house>
</StandardHeader>
```

2. Attributes containing the intelligence (no "room" nodes have been omitted).

```
<?xml version="1.0" encoding="WINDOWS-1252"?>
<StandardHeader>
   <DateTime>12/08/2010 16:03:46</DateTime>
   <PacketNumber>1</PacketNumber>
   <AnyMorePackets>Y</AnyMorePackets>
   <TotalPackets>4</TotalPackets>
  <house>
    <room room_name="kitchen" room_item="sink"/>
    <room room_name="kitchen" room_item="table"/>
    <room room_name="kitchen" room_item="counter"/>
    <room room_name="kitchen" room_item="microwave"/>
    <room room_name="kitchen" room_item="oven"/>
```

```
    <room room_name="kitchen" room_item="range"/>
    <room room_name="kitchen" room_item="refrigerator"/>
    <room room_name="living_room" room_item="pictures"/>
    <room room_name="living_room" room_item="chair"/>
    <room room_name="living_room" room_item="pictures"/>
    <room room_name="living_room" room_item="HDTV"/>
    <room room_name="living_room" room_item="couch"/>
    <room room_name="living_room" room_item="chandelier"/>
    <room room_name="den" room_item="surround_sound"/>
    <room room_name="den" room_item="fireplace"/>
    <room room_name="den" room_item="table"/>
    <room room_name="den" room_item="chair"/>
    <room room_name="den" room_item="lamp"/>
    <room room_name="den" room_item="étagère"/>
   <room room_name="den" room_item="HDTV"/>
  </house>
</StandardHeader>
```

If we compare the byte count of 1) vs. 2) we have 2153 vs.1307 or a nearly 40% in reduction in the size of the XML Document using attributes to contain the intelligence of the payload. Over large XML Documents, that is a significant savings.
Observe the two different ways of shredding the intelligence:

3. Using XPATH:

```
SELECT extractValue(p.XMLtext,'/StandardHeader/DateTime') date_time,
       extractValue(p.XMLtext,'/StandardHeader/PacketNumber') packet_number,
       extractValue(value(t1),'/room/room_name/@name') room_name,
       extractValue(value(t1),'/room/room_item/@item') room_item
from testclob p,
     table(xmlsequence(extract(p.XMLtext,'/StandardHeader/house/room'))) t1
WHERE  p.id = 6;
```

4. Using dbms_xmldom8:

```
DECLARE

    dDoc            DBMS_XMLDOM.DOMDocument;
    nlNodeList      DBMS_XMLDOM.DOMNodeList;
    nNode           DBMS_XMLDOM.DOMNode;
    nNode2          DBMS_XMLDOM.DOMNode;
    nmNodeMap  DBMS_XMLDOM.DOMNamedNodeMap;
    aAttr           DBMS_XMLDOM.DOMAttr;
    cText           CLOB;
    i               NUMBER;
    j               NUMBER;
    strRoom         VARCHAR2(50);
    strItem         VARCHAR2(50);

BEGIN

    -- Get the Clob from the table
    SELECT XMLTYPE.getclobval(XMLText)
    INTO cText
    FROM testClob
    WHERE Id = 5;
```

```
    -- Create the xml document from the Clob
    dDoc := DBMS_XMLDOM.NEWDOMDOCUMENT(cText);

    -- Get the nodes corresponding the the 'room' tag
    nlNodeList := DBMS_XMLDOM.GETELEMENTSBYTAGNAME(dDoc, 'room');

    -- Loop through the 'room' nodes
    FOR i IN 0..DBMS_XMLDOM.GETLENGTH(nlNodeList)-1 LOOP

        -- Get the ith node
        nNode := DBMS_XMLDOM.ITEM(nlNodeList, i);

        -- Get the attributes of this node
        nmNodeMap := DBMS_XMLDOM.GETATTRIBUTES(nNode);

        -- Find the room_name attribute
        nNode2 := DBMS_XMLDOM.GETNAMEDITEM(nmNodeMap,'room_name');
        aAttr := DBMS_XMLDOM.MAKEATTR(nNode2);
        strRoom := DBMS_XMLDOM.GETVALUE(aAttr);
        DBMS_OUTPUT.PUT('Room_Name = '||RPAD(strRoom,15)||CHR(9)||CHR(9));

        -- Find the room_item attribute
        nNode2 := DBMS_XMLDOM.GETNAMEDITEM(nmNodeMap,'room_item');
        aAttr := DBMS_XMLDOM.MAKEATTR(nNode2);
        strItem := DBMS_XMLDOM.GETVALUE(aAttr);
        DBMS_OUTPUT.PUT_LINE('Room_Item = '||strItem);

    END LOOP;

    -- Free ressources
    DBMS_XMLDOM.FREENODE(nNode);
    DBMS_XMLDOM.FREENODE(nNode2);
    DBMS_XMLDOM.FREEDOCUMENT(dDoc);

END;
/
```

XPath requires less pl/sql to extract the intelligence than dbms_xmldom. DBMS_XMLDOM traverses the nodes of the XML Document. XPath requires path definitions. Each has its own merits and shortcomings.

Conclusion

XML development began on this project a number of months ago. We recently completed the majority of the SQLX (XML document creation) and XPath (document shredding) programming and began testing. Except for adjusting the SQLX queries and changes to XPath (parsing out similar tag fields under one parent), we made no major revisions. As far as the project is concerned, Oracle's XML DB is a stable environment and meets all the standards according to W3C. However, programming in XML DB does require a bit of some effort. The constructor XMLTYPE creates an instance of an XML object. We used it to convert a CLOB into a properly qualified XML document for storage into a table with an XMLTYPE column. When we shred XML documents we made extensive use of the function EXTRACT (with methods getstringval and getnumval). Once we developed the primitives for XML fragment creation and XML document shredding, we found ourselves using the same type of code over and over again. The iterative process for developing a project using Oracle's XML DB is quite similar and an extension to PL/SQL programming. We hope the information presented here establishes a good starting point for those embarking on a project using XML.

About the Author

Coleman Leviter, OCP is employed as an IT Software Systems Engineer at Arrow Electronics. He has presented at IOUG's Collaborate and at Oracle Open World. His articles have appeared in Select Magazine, IOUG's Tips & Best Practices Booklet for Oracle professionals and ODTUG Journal. He is the WEB SIG chair and sits on the steering

committee at the NY Oracle Users Group (www.nyoug.org). Additionally, he was the 2012 IOUG COLLABORATE Conference Chair. He has worked in the financial services industry and the aerospace industry where he developed Navigation, Flight Control and Reconnaissance software for the F-14D Tomcat. Coleman was recently elected to the IOUG Board of Directors. Coleman has a BSEE from Rochester Institute of Technology, an MBA from C.W. Post and an MSCS from New York Institute of Technology. He may be contacted at cleviter@ieee.org

References

[1] Wikipedia, http://en.wikipedia.org/wiki/Transportation_management_system
[2] Wikipedia, http://en.wikipedia.org/wiki/B2b
[3] Wikipedia, http://en.wikipedia.org/wiki/Oracle_E-Business_Suite
[4] Wikipedia, http://en.wikipedia.org/wiki/Warehouse_management_system
[5] Oracle® Database PL/SQL Packages and Types Reference 11*g* Release 1 (11.1), Part Number B28419-03
[6] Wikipedia, http://en.wikipedia.org/wiki/XML_schema
[7] Wikipedia, http://en.wikipedia.org/wiki/XML_Schema_(W3C)
[8] http://download.oracle.com/docs/cd/B19306_01/appdev.102/b14258/d_xmldom.htm

RAC One-Node: Reduce Costs and Replace Third-Party Cluster Services with Grid Infrastructure and RAC One-Node

John Larkin, Advanced Database Services

Introduction

There are several vendors that can provide clustering capabilities for databases. The beauty of using Oracle's RAC One-Node is that it is integrated with the Oracle stack and can help to reduce overall software costs. This gives you one vendor that can handle both the clustering software and the database. Additionally, you gain the ability to upgrade to a fully RAC-enabled database at a later date with relative ease.

I will discuss the basic Grid Infrastructure required to implement RAC One-Node and how to avoid some of the pitfalls inherent in constructing a RAC environment. Additionally I will touch upon some of the utilities and home-grown scripts that can be used to manage this environment. I'll discuss the high-level installation steps and some options that have made my life easier.

The genesis of this paper is a multi-year project to migrate 140+ databases on Veritas-managed clusters to RAC One-Node Databases, backed by Grid Infrastructure clusters. I will conduct a high-level survey of the functionality of Oracle RAC offering.

As with most projects, it's the work that you do up-front that can save you time when you finally get to the implementation phase.

Scope

This is intended to be an exploration of some basic concepts and tools that you can use to build and run an 11g R2 Oracle database in an 11gR2 Grid Infrastructure clustered environment. I will discuss the ASMLib driver, Linux RPM's, Filesystem layout, Grid Infrastructure installation and a patching workaround, Oracle RDBMS installation, Oracle RAC One-Node Database installation, RAC One-Node migration from one node to another and some basic care and feeding instructions. Oracle's Grid Infrastructure support for applications is not within the scope of this paper. I will assume for the purposes of this paper that I am working on a 2-Node cluster with a RAC One-Node database.

Hardware

The initial cluster configuration was designed as a starting point for the eventual migration of more than 40 databases. The hardware and software list follows:

- two - HP Blade servers x86-64, 8 CPU, 48G Ram
- 6 GigE NIC's
- RedHat Enterprise Linux Release 5.3
- Oracle 11g R2.02

Clustering Technologies

There are many vendors that provide Clustering services : Veritas/Symantec(VCS), IBM(HACMP), Microsoft(Clustering Service), Oracle (Grid Infrastructure). Clustering can be broken down into two large categories, based on how resources are shared throughout a cluster. Today, two of the more popular systems are shared-nothing and shared-disk.

Shared-Nothing clusters, like IBM's HA-CMP and Microsoft's MSCS separate resources, with each server have a dedicated set of resources. Typically when information is needed from more than one node, the clustering software will use the network to communicate.

Veritas Clustered File system and Oracle's Grid Infrastructure implement a shared-disk cluster. The disks containing the database files are shared among all of the nodes. With Oracle's Grid Infrastructure, inter-instance communication occurs

via the interconnect private network. In-memory changes to data blocks in one instance are available to other instances in the cluster via Oracle's Cache Fusion. This essentially makes a single database buffer cache from the memory allocated to the database buffer caches across all active instances in the cluster.

Network Setup

The Grid Infrastructure (GI) makes use of both public and private networks to support high availability.
The Virtual IP's (VIP) are part of the public network and are used to reference the nodes, providing a level of abstraction for host and database access.
The SCAN (Single Client Access Name) feature allows the use of a single name to connect to a RAC-enabled database, regardless of which node(s) the database is using. It uses the public network.
The Interconnect (or heartbeat) uses a private network and is used by GI for healthcheck/cluster information and by a RAC-enabled database to transport data blocks via Cache Fusion from one instance to another.
Most likely your SA or Network Administrator will define these IP addresses for you (preferably before you begin your installation)

Grid Infrastructure

The backbone of Oracle's clustering capabilities is the Grid Infrastructure. It is the framework which is used to implement Oracle's High Availability (HA) features. During installation of GI (formerly named Clusterware) you create the HA services that will monitor the cluster and support RAC-enabled databases (and clustered applications which are outside the scope of this paper). Another point of confusion when talking about RAC is the distinction between Oracle Database, or RDBMS, binaries and Oracle Grid Infrastructure, or Clusterware, binaries. Grid Infrastructure executes the GI binaries from the GRID_HOME and RAC-enabled databases execute the Oracle Database (or RDBMS) binaries from ORACLE_HOME.
For the purposes of this paper I will reference the Grid Infrastructure ORACLE_HOME as "GRID_HOME" (or GH). This should help to distinguish between that and the ORACLE_HOME used by the RDBMS.
As the RAC product has matured and become more commonplace in the datacenter, Oracle has continued to develop GI. In 11gR2, Grid Infrastructure is the main component of Oracle clustering and is tightly coupled with ASM. If you have watched the development of Oracle's product line over the last several years, you can recall that ASM has moved from an optional method for storage management to the preferred method. Also I have seen that the Clusterware stack has moved from an install that was separate from ASM to become one installation, using ASM as the storage manager. While at first I was skeptical of using a database company's software to manage storage, ASM has been proven to be an effective and safe tool. As I continue in this paper I will show how these two once-separate products are now wrapped into one.

The RAC One-Node Database

A RAC One-Node database instance really has a split personality. In looks much like any non-RAC database when it is not in a state of flux(static), but can look more like a full RAC database when it is relocating. The main property visible on a database that might indicate it employs RAC One-Node is the presence of a second UNDO tablespace and the more than one thread. While this is also visible on full-fledged RAC databases, you can make a further determination by querying gv$thread. The values from the inst_id and status columns should help to determine the type of database. For a One-Node database you will only see INST_ID=1 with once STATUS=OPEN and another STATUS=CLOSED. For a full RAC database you will (normally) see more than 1 INST_ID and all STATUS=OPEN.

```
RAC One-Node:

SQL> select * from gv$thread;

INST_ID          THREAD#   STATUS        ENABLED             GROUPS   INSTANCE
----------   ------------   ------------  ------------   -----------   ------------
         1              1           OPEN        PUBLIC             2   PCT002Q_1
         1              2         CLOSED        PUBLIC             2   PCT002Q_2
```

```
Full RAC :

INST_ID          THREAD#   STATUS        ENABLED             GROUPS   INSTANCE
----------   ------------   ------------  ------------   -----------   ------------
         1              1          OPEN        PUBLIC             2          BRNYT1
         1              2          OPEN        PUBLIC             2          BRNYT2
         1              3          OPEN        PUBLIC             2          BRNYT3
         2              1          OPEN        PUBLIC             2          BRNYT1
         2              2          OPEN        PUBLIC             2          BRNYT2
         2              3          OPEN        PUBLIC             2          BRNYT3


SQL> select tablespace_name, contents, status
          from dba_tablespaces

TABLESPACE_NAME                          CONTENTS         STATUS
---------------------------------      ---------------  -----------
ORACLE_TRN_UNDO_1                        UNDO             ONLINE
ORACLE_TRN_UNDO_2                        UNDO             ONLINE
…
(one tablespace for each instance)
```

Also the REGISTRY has information that can be used to see what features are installed.
select comp_id, status, version, comp_name from dba_registry;

COMP_ID	STATUS	VERSION	COMP_NAME
CATALOG	VALID	11.2.0.2.0	Oracle Database Catalog Views
CATPROC	VALID	11.2.0.2.0	Oracle Database Packages and Types
JAVAVM	VALID	11.2.0.2.0	JServer JAVA Virtual Machine
CATJAVA	VALID	11.2.0.2.0	Oracle Database Java Packages
RAC	VALID	11.2.0.2.0	Oracle Real Application Clusters

I will also look at the behavior of a RAC One-Node database that is relocating to another node in the cluster. (srvctl relocate database -d DBNAME -n RacNode2) once I get past the installation section.

Network Requirements

RAC One-Node has the same basic network requirements as a full RAC database, it's just that some of the addresses will not be fully exercised until you relocate the One-Node database to another node. Typically you will use 3 NICs, one for the Private network (Interconnect), one for the Public network (SCAN, VIP's), and one for I/O if you are using Network-attached storage.

SCAN (Single Client Access Name) is a feature that allows the use of a single name to connect to a RAC-enabled database, regardless of which node(s) the database is using. This is configured prior to installation by adding the name into DNS and having it resolve to 3 IP addresses using a round-robin scheme. The Network administrator can help with this configuration. Typically I'll add the SCAN entries to the /etc/hosts file as comments, just so that I have them available for reference. (temporary workaround to complete installation: Define the SCAN in /etc/hosts with a single IP). During cluster configuration several SCAN resources are created : a SCAN Vip and SCAN Listener for each of the 3 IP's defined in the DNS SCAN entry. Each SCAN Listener runs from the Grid Infrastructure home.

Let's clarify some of the terms used to this point. When using the term VIP by itself, it refers to the virtual IP that represents a node in the cluster server1-vip. SCAN, SCAN-vip and SCAN_Listener are sometimes used interchangeably but they really refer to very distinct items. SCAN is the facility that implements a redundant location-independent connection to a database. The SCAN-vip and SCAN_Listener are the means to implement that connection. In an OUI-guided GI installation, there are 3 SCAN-Vip's and associated SCAN_Listeners, running on separate nodes. If there are

only 2 nodes in the cluster, then 2 of the SCAN instantiations will run on one of the nodes and one SCAN listener on the other.
By looking at the behavior of SCAN it will show us one of the High-Availability features in the Grid Infrastructure stack : Run nslookup multiple times and you should see the order of the IP's change. If this is not reflecting the round-robin attributes, then your Network administrator will need to adjust the configuration.

```
#>nslookup pctcrs1a-prd
Name:   pctcrs1a-prd.ads.com
Address: 169.83.244.108             <==== First IP used on first lookup
Name:   pctcrs1a-prd.ads.com
Address: 169.83.244.109
Name:   pctcrs1a-prd.ads.com
Address: 169.83.244.107

#>nslookup pctcrs1a-prd
Name:   pctcrs1a-prd.ads.com
Address: 169.83.244.107             <=== Second IP used
Name:   pctcrs1a-prd.ads.com
Address: 169.83.244.108             <=== First IP not used on second call
Name:   pctcrs1a-prd.ads.com
Address: 169.83.244.109
```

So, one can see SCAN rotating through the IP addresses, spreading out the network load and the database load across the various nodes(assuming you have the IP's bound to NIC's on multiple servers). SCAN information is available from the cluster registry:

```
#=>srvctl status listener
Listener LISTENER is enabled
Listener LISTENER is running on node(s): pctcrs11,pctcrs10

#=>srvctl status scan_listener
SCAN Listener LISTENER_SCAN1 is enabled
SCAN listener LISTENER_SCAN1 is running on node pctcrs11
SCAN Listener LISTENER_SCAN2 is enabled
SCAN listener LISTENER_SCAN2 is running on node pctcrs10
SCAN Listener LISTENER_SCAN3 is enabled
SCAN listener LISTENER_SCAN3 is running on node pctcrs10

#=>srvctl config scan_listener
SCAN Listener LISTENER_SCAN1 exists. Port: TCP:1657
SCAN Listener LISTENER_SCAN2 exists. Port: TCP:1657
SCAN Listener LISTENER_SCAN3 exists. Port: TCP:1657
```

Typically, you will also configure the normal /etc/hosts entries for the servers, comment out the SCAN entries, add the Interconnect entries and the VIP entries.

ASM – Automatic Storage Management

At the beginning I had mentioned that ASM is now integral to the function of the Grid Infrastructure. Perhaps Oracle's creation of the AsmLib (ASM Library driver) RPM is an indicator of how much they want you to migrate to ASM. It simplifies the configuration of disks to be used by ASM and helps to eliminate some of the problems inherent in using raw disks for ASM *. Combined with disk aliases, it helps you to define and name your ASM disks from the start and eliminate ambiguity in which are in use and their intended purpose.
AsmLib is highly dependent on the OS kernel and so, especially after an OS upgrade or a new install you need to verify that the AsmLib rpm is compatible with the O/S kernel level. I recently went through an unplanned O/S upgrade in the

middle of a server/database build. When I went to configure the AsmLib driver, I received no errors from the scandisks and listdisks commands, the configure command ran without error. “Only” the status command showed an issue. After working with the configuration for quite some time it dawned on me that perhaps the rpm for AsmLib had not been updated. The following check (and also cluvfy) pointed me in the right direction.

```
      rpm -qa --queryformat "%{NAME}-%{VERSION}-%{RELEASE} (%{ARCH})\n"| grep oracleasm |
sort
          **      oracleasm-2.6.18-194.3.1.el5-2.0.5-1.el5 (x86_64)  **
            oracleasm-support-2.1.3-1.el5 (x86_64)
            oracleasmlib-2.0.4-1.el5 (x86_64)
```

When I compared this to another system that was working correctly I found the difference in versions. Apparently Yum had not been updated with the AsmLib rpm that matched the kernel. You can also look at Metalink to get information on compatible kernel/rpm version. Once the correct rpm was installed, I was able to successfully configure and load the AsmLib driver module.
The error messages produced by AsmLib/oracleasm do not provide a lot of detail, but at least they can indicate that a problem exists.:

```
vi /var/log/oracleasm
      Cleaning any stale ASM disks...
      Loading module "oracleasm": failed
      Unable to load module "oracleasm"
      ..
      Creating /dev/oracleasm mount point: /dev/oracleasm
      Loading module "oracleasm": failed
      Unable to load module "oracleasm"
```

Once the ASMLib rpm is installed, it must be configured to your environment. This is accomplished through the /usr/sbin/oracleasm configure command. Running it with no parameters will display the current configuration. Typically it will look like this on a fresh install:

```
            ORACLEASM_ENABLED=false
            ORACLEASM_UID=
            ORACLEASM_GID=
            ORACLEASM_SCANBOOT=true
            ORACLEASM_SCANORDER=""
            ORACLEASM_SCANEXCLUDE=""
```

Running with the –i parameter will allow you to set the values for your particular disk configuration:

```
      /usr/sbin/oracleasm configure -i
```

Configuring the Oracle ASM library driver.

This will configure the on-boot properties of the Oracle ASM library driver. The following questions will determine whether the driver is loaded on boot and what permissions it will have. The current values will be shown in brackets ('[]'). Hitting <ENTER> without typing an answer will keep that current value. Ctrl-C will abort.

Default user to own the driver interface [grid]:
Default group to own the driver interface [asmadmin]:
Start Oracle ASM library driver on boot (y/n) [y]:
Scan for Oracle ASM disks on boot (y/n) [y]:
Writing Oracle ASM library driver configuration: done

This will create the /etc/sysconfig/oracleasm configuration file, and the /dev/oracleasm mount point, and also mount the ASMLib driver file system. There are still several more parameters that you will most likely need to set, but you must editing the configuration file to change these :

```
vi /etc/sysconfig/oracleasm
              ORACLEASM_SCANORDER="/dev/mapper/asm"   <= get the devices prefixed by...
              ORACLEASM_SCANEXCLUDE="sd"              <= exclude these devices( otherwise
                                                         you will see duplicates for
                                                         multipath'ed devices. )
```

There are actually 2 implementations of the oracleasm command. The base executable is /usr/sbin/oracleasm, but there is also a scripts that implements high-level functions. This takes simpler parameters and can be handy for certain functions - /etc/init.d/oracleasm, using parameters like stop, start. The executable requires a different syntax : exit, init to perform the same functions. (/usr/sbin/oracleasm init, /etc/init.d/oracleasm start) The stop function might alarm you the first time you try to run it because of the messages it displays

```
      sh-3.2# /etc/init.d/oracleasm stop
      Dropping Oracle ASMLib disks:                          [  OK  ]     ***
      Shutting down the Oracle ASMLib driver:                [  OK  ]
```

The disks are not permanently dropped, they just become inaccessible while the drive is not running.
Once the driver is in place and configured, you can build the ASM disks that you will use during the install of the Grid Infrastructure. There had been an issue reported in our group if you define more disks than is required for ASM before installation is complete. During the software installation and ASM creation the installer appeared to hang. To get around the issue I only defined the disks needed for ASM at this point. I performed additional tests with more luns and it appears that the time required to complete the configuration is related to the number of disks defined via asmlib.
To maintain consistent naming across the luns used for ASM, I utilize the /etc/multipath.conf and /etc/multipath/bindings files. Both the Blacklist and Multipath parameters are useful. If you have a LUN presented to more than one cluster, the blacklist parameter can be used to exclude specific luns from view on the server. I would recommend that the Multipath parameter be used as it permanently associates a WWID (Lun identifier) with an alias that can be used when defining disks for use with the oracleasm command. You can also use the /etc/multipath/bindings file to permanently relate mpath99 names to a WWID. Both of these help to maintain the same names for luns regardless of changes that a SAN engineer might implement that could change mpath names.

```
multipaths {
        multipath {
                wwid    999999999999999999999999
                alias   asmocr01
                }
      …
```

Now that ASMLib has been configured I can add some disks for use by ASM. Initially I'll add OCR and System disks. First, I need to check on the sizes of the lun's available and the names associated with them. My asmmap_inactive script works for Linux systems if you have access to the root account:

```
          echo "( asm raw device in /dev/mapper/ OR dm- in /dev/)"

          for i in `/sbin/multipath -v 1 -l | grep "^asm" | sort` ;
          do
          v_asmdisk=$i
          v_size=`/sbin/multipath -ll $v_asmdisk| awk -F[ '{print $2}'|awk -F] '{print
$1}' | grep size`; #lines w size
```

```
        v_device=`ls -alF /dev/mpath/$v_asmdisk|awk '{ print $11 }'|awk -F"." '{print
$3 }'|awk -F"/" '{print $2}'`;
        echo $v_asmdisk $v_size "/dev/"$v_device |awk '{ print "ASM candidate "$1 "   "
$2 "    " $3  }';
        done

sh-3.2# asmmap_inactive
( asm raw device in /dev/mapper/ OR dm- in /dev/)
ASM candidate asmocr01   size=10G    /dev/dm-5
ASM candidate asmsys01   size=10G    /dev/dm-10
ASM candidate asm1p001   size=30G    /dev/dm-18
ASM candidate asm1par1   size=30G    /dev/dm-13
ASM candidate asmacf01   size=10G    /dev/dm-8
ASM candidate asmbackup01   size=125G    /dev/dm-25
```

These will be the disk names used to create the asm disks. You can also see the mountpoint created by asmlib.

```
sh-3.2# ls -alF /dev/oracleasm/disks
drwxr-xr-x 1 root root           0 Aug 24 16:57 ./
brw-rw---- 1 grid asmadmin 253, 10 Aug 24 16:51 ASMQM001
brw-rw---- 1 grid asmadmin 253, 30 Aug 24 16:58 ASMD003
```

The basic oracleasm commands are: (arranged in order of use)

```
    init                    Load and initialize the ASMLib driver
    configure               Configure the Oracle Linux ASMLib driver
    createdisk              Allocate a device for Oracle ASMLib use
    scandisks               Scan the system for Oracle ASMLib disks
    status                  Display the status of the Oracle ASMLib driver
    listdisks               List known Oracle ASMLib disks
    querydisk           Determine if a disk belongs to Oracle ASMlib
    exit                    Stop the ASMLib driver
    deletedisk              Return a device to the operating system
    renamedisk          Change the label of an Oracle ASMlib disk
    update-driver Download the latest ASMLib driver

/oracle/dba-work.390=>/usr/sbin/oracleasm deletedisk
Usage: oracleasm-deletedisk [-l <manager>] [-v] <label>|<device>
```

I am now ready to create the asm disks. First run scandisks to see if any already exist.

```
sh-3.2# date;  /usr/sbin/oracleasm scandisks ; date
Reloading disk partitions: done
Cleaning any stale ASM disks...
Scanning system for ASM disks...
```

No disks were present, so I can define those needed for the GI installation. Here I map the multipath'ed luns to ASM disks. The ASM disks will show up during the Grid Infrastructure installation and the execution of ASMCA.
They will be used for the ASM instance. ASMOCR01 will hold the Cluster Registry and Voting Disk files. ASMSYS01 will hold the infrastructure tablespace files (system, sysaux, etc).

```
/usr/sbin/oracleasm createdisk      ASMOCR01    /dev/mapper/asmocr01
/usr/sbin/oracleasm createdisk      ASMSYS01    /dev/mapper/asmsys01
```

After creating the disks I can scan again and see that they are now defined via AsmLib:

```
sh-3.2# date;  /usr/sbin/oracleasm scandisks ; date
Reloading disk partitions: done
Cleaning any stale ASM disks...
Scanning system for ASM disks...
      Instantiating disk "ASMOCR01"
Instantiating disk "ASMSYS01"
```

For future reference : Once the Database is created you can query the database to see which ASM disks are associated with a diskgroup.

```
select vadg.NAME vadg_name, vad.name
from
v$asm_disk      vad,
v$asm_diskgroup vadg
WHERE
   VADG.group_number = VAD.group_number
order by vadg.name, vad.name
/

VADG_NAME                       NAME
------------------------------  --------------------
OCRDG01                         ASMSYS03
ARCHDG01                        ASM1PAR1
INFRADG01                       ASMSYS01
```

Now run the CLUVFY utility to confirm that hardware and os settings are correct. Review the CVU report. All checks performed by CVU should be reported as "passed" before continuing with the Oracle grid infrastructure installation.

```
./runcluvfy.sh stage -post hwos -n pctcrs10,pctcrs11 -verbose
Checking node reachability...

Check: Node reachability from node "pctcrs10"
  Destination Node                      Reachable?
  pctcrs10                              yes
  pctcrs11                              yes
Result: Node reachability check passed from node "pctcrs10"

Checking user equivalence......
Checking node connectivity...
```

All checks should be passed. At that point the Grid Infrastructure installation can begin.

Install the Grid Infrastructure

Change to the Grid installation directory , /oracle/Oracle11gR2.11.2.0.2/grid.
Startup you Xwindows server and start the OUI (Oracle Universal Installer):

```
 ./runInstaller  &
```

The prompts and suggested responses follow:

Screen Name	Response
Download Software Updates	skip - Yes I wish to remain UNINFORMED !!!

Select Installation Option	Select " Install and Configure Grid Infrastructure for a Cluster"
Select Installation Type	Select " Advanced Installation"
Grid Plug and Play Information	Un-check the option to "Configure GNS". ***
Cluster name:	clst-pctcrs1-pd (< 15 characters)
SCAN NAME	pct-dev-scan.ads.com (overlayed generated name)
SCAN PORT:	1577 (978686 Ora forums thread - svrctl modify scan_listener -p <port#>)
Cluster Node Information	add ALL additional NODES to the cluster. EDIT to COPY original text "Add" button to add additional nodes "EDIT" all "virtual IP Name"s change generated "pctcrsxx-vip" to "pctcrsxx-vip1" for each node to match your vip naming standards
Test SSH connectivity	Enter password for Grid user, [TEST] button. You should see the following message: "Psswordless SSH connectivity between the selected nodes already established."
Specify Network Interface Usage (7 of 16)	Identify the network interface for "Public" and "Private" network. BOND0 = VIP/Public BOND1 = for backups BOND2 = INTERCONNECT/PRIVATE
Storage Option Information - (9 of 16)	"Automatic Storage Management (ASM)". This option configures OCR and voting disk files on ASM storage.
Create ASM Disk Group	Create an ASM Disk Groups to store the Oracle Clusterware files : OCRDG01, OCRDG02, OCRDG03. REDUNDANCY NORMAL (for OCR) requires 3 disks.
Privileged Operating System Groups	Oinstall
Installation Location (STEP 13 of 17)	Place BASE outside of software directory to ease management of audit log files.
“performing pre-req checks...." (15 of 18)	BUG : 1210863.1, NOTE:1267569.1 - PRVF-5449 : Check of Voting Disk location "ORCL:(ORCL:)" failed see also : How to Collect CVU Trace / Debug Output Generated by RUNCLUVFY.SH [ID 986822.1] Run: /etc/init.d/oracleasm status and /etc/init.d/oracleasm listdisks If both of these pass, continue with install.
INSTALL	When you get “oraInstRoot.sh and root.sh” prompt, STOP.
PATCH the Grid Home.	By waiting here to run the root scripts you can patch the HOME directories during the installation. This is much simpler that patching after the software is completely installed.
On ALL NODES :	Run OPatch

. export MEDIA_HOME=/oracle/Oracle11gR2.11.2.0.2
apply patch #1

. ${GRID_HOME}/OPatch/opatch napply -oh ${GRID_HOME} -local ${MEDIA_HOME}/12419353 -
invPtrLoc /oracle/orafs/grid_base/oraInventory/oraInst.loc

leave email addr blank... (OCM security)
. enter NONE when prompted for email/ nothing for pw / NONE http url.

now patch #2
. ${GRID_HOME}/OPatch/opatch napply -oh ${GRID_HOME} -local ${MEDIA_HOME}/12419331 -
invPtrLoc /oracle/orafs/grid_base/oraInventory/oraInst.loc

Run the ROOT scripts.
. /oracle/grid_base/oraInventory/orainstRoot.sh
. /oracle/grid_base/products/11.2.0.2/root.sh
This took a while to get past the "located 3 voting disks" message 5+minutes.

If you have problems with the install you can run the new deinstall script to cleanup the installation for a fresh rerun.

$GRID_HOME/deinstall/deinstall

As the GRID user :

asmca &

Screen Name	Response
. Disk Groups	From the "Disk Groups" tab, click the " Create" button.
. Create Disk Group	The "Create Disk Group" dialog should show two of the ASMLib volumes I created earlier. If the ASMLib volumes created earlier do not show then click the "Change Disk Discovery Path" button and check the path ."Redundancy" section, choose " External (none)" for all but OCRDG01. Do NOT select QUORUM. (only for voting disks)

You now have a functioning ASM instance and can install the RDBMS software and create the RAC One-Node database.

Install the Oracle/RDBMS Binaries

Logon as the Oracle user (the one that will own the RDBMS software.)
Change to the Oracle installation directory , /oracle/Oracle11gR2.11.2.0.2/database
Startup you Xwindows server and start the OUI (Oracle Universal Installer):

```
./runInstaller  &
```

The prompts and suggested responses follow:

Screen Name	Response
Security Updates	Skip
Installation Option	"Install database software only".(create the database later)
Grid Options	Select the "Real Application Clusters database installation" radio button (default) and verify that ALL Oracle RAC nodes are checked in the "Node Name" window.

[SSH Connectivity] button.	Enter Oracle password
Database Edition	Select " Enterprise Edition". (STEP 6 of 12)
Software Location	see message "Checking specified locations on remote nodes.".
Operating System Groups	Database Administrator (OSDBA) Group: dba Database Operator (OSOPER) Group: oinstall
Install	When get “root.sh” prompt, STOP.
Patch the Oracle Home On ALL NODES :	Run OPatch

. export MEDIA_HOME=/oracle/Oracle11gR2.11.2.0.2
run pre-patch
${MEDIA_HOME}/12419353/custom/server/12419353/custom/scripts/prepatch.sh -dbhome
${ORACLE_HOME}
apply patch #1
. ${ORACLE_HOME}/OPatch/opatch napply -oh ${ORACLE_HOME} -local
${MEDIA_HOME}/12419353 -invPtrLoc /oracle/orafs/grid_base/oraInventory/oraInst.loc

leave email addr blank... (OCM security)
. enter NONE when prompted for email/ nothing for pw / NONE http url.

now patch #2
. ${ORACLE_HOME}/OPatch/opatch napply -oh ${ORACLE_HOME} -local
${MEDIA_HOME}/12419331 -invPtrLoc /oracle/orafs/grid_base/oraInventory/oraInst.loc

run POST-patch script
${MEDIA_HOME}/12419353/custom/server/12419353/custom/scripts/postpatch.sh -dbhome
${ORACLE_HOME}

Run the ROOT scripts.
. /oracle/products/11.2.0.2/root.sh

You will see the message "The installation of Oracle Database was successful". But you just have the RDBMS software installed at this point.

Create the RAC One-Node Database

As the Oracle user, start an Xwindows server if not already running. Start the installer.

dbca &

Screen Name	Response
Welcome Screen	Select Oracle RAC One Node database.
Operations	Create a Database.
Template	Custom database

Database name	Upper case!
Database File Locations	Storage Type: Automatic Storage Management (ASM) (cluster file sys is the only other opt)
Storage Locations:	Use Oracle-Managed Files
Database Area:	+ORACRS
Recovery Area	UNCHECK "specify Flash Recovery Area
CHECK "Enable Archiving"	
Database Content	
Custom Scripts	
Initialization Parameters	
Database Storage	Default settings
FINISH buttom	

When the DBCA has completed, you will have a fully functional Oracle RAC One-Node database cluster.
Change default port : Doc ID: 359277.1 Changing Default Listener Port Number
How To : Changing Default Listener Port Number
NOTE:99721.1 - Listening Port numbers

srvctl modify listener -l LISTENER -p "TCP:1560"

Add Application tablespaces as needed.

Relocate the RAC One-Node Database to an Alternate Node

NODE1 :
/home/grid=>srvctl status database -d pct002t
Instance PCT002T_1 is running on node pctcrs10
Online relocation: INACTIVE <<===== relocate inactive

srvctl relocate database -d PCT002T -n pctcrs11

NODE1 :
/home/grid=>srvctl status database -d pct002t
Instance PCT002T_1 is running on node pctcrs10
Online relocation: ACTIVE <<===== relocation active
Source instance: PCT002T_1 on node pctcrs10
Destination instance: PCT002T_2 on node pctcrs11

NODE2:
/home/grid=>srvctl status database -d pct002t
Instance PCT002T_2 is running on node pctcrs11
Online relocation: INACTIVE <<===== w/ no reloc active

/home/oracle=>psg pmon

```
oracle    2544     1  0 10:06 ?        00:00:00 ora_pmon_PCT002T_2
grid     28488     1  0 Jan31 ?        00:00:00 asm_pmon_+ASM2

:/home/oracle=>export ORACLE_SID=PCT002T_2
/home/oracle=>sqlplus / as sysdba

SQL*Plus: Release 11.2.0.2.0 Production on Wed Feb 1 10:09:13 2012
Oracle Database 11g Enterprise Edition Release 11.2.0.2.0 - 64bit Production
With the Partitioning, Real Application Clusters, Automatic Storage Management, OLAP,
SQL> show parameter undo

NAME                                 VALUE
------------------------------------ -----------------------------------
undo_management                      AUTO
undo_retention                       900
undo_tablespace                      UNDOTBS2

---- /oracle/dba-work/e169506>srvctl config database -d pct002t
Database unique name: PCT002T
Database name: PCT002T
Oracle home: /oracle/products/11.2.0.2
Oracle user: oracle
Spfile: +INFRADG01/PCT002T/spfilePCT002T.ora
Domain: card.jpmchase.net
Start options: open
Stop options: immediate
Database role: PRIMARY
Management policy: AUTOMATIC
Server pools: PCT002T
Database instances:
Disk Groups: INFRADG01,ARCHDG01,INFRADG02
Mount point paths:
Services: PCT002T_dflt
Type: RACOneNode
Online relocation timeout: 30
Instance name prefix: PCT002T
Candidate servers: pctcrs10,pctcrs11
```

Cost Differences of RAC One-Node and Full RAC

The price difference between a Full-fledged RAC database and a RAC One-Node database can be a consideration in choosing the type of database to create. Typically the RAC One-Node savings is approximately 55% over the cost of a fully RAC-enabled database. Sample list prices are $23k for Full RAC, $10k for RAC One-Node.
The licensing limitation for RAC One-Node allows a user to migrate to another node for 10-days per calendar year without the need for additional licenses.

Conclusion

There are so many features available in the Oracle Clusterware and RAC software stack that you could write a book describing all of the options. I have attempted to provide an overview of the RAC installation process and some options to make it a little easier. I have reviewed some of the basic components required for building a RAC One-Node database.

Having a database running on a cluster provides you with many options to reduce downtime when maintenance will necessitate a server outage. The failover capabilities of RAC One-Node also provide an automated method to restart a database in the event of hardware problems and can even allow queries to be restarted after an instance has relocated to a remaining node. Oracle will continue to enhance the capabilities as the product matures, providing us with an even more robust high availability system.

Reference Material

Oracle 11g Adminstrators guide
MOSC – Bug reports, Technical Notes, How-To articles
Jeffrey Hunter – Articles for building a RAC system for less than $2000.
Vincent Chan – Moving to RAC One-Node

Modernizing Workflow and Data Integration: The Present and Future of Data Warehousing

Anthony D. Noriega, MSCS, MBA, BSSE, OCP

Oracle Data Integration model provides support over 10,000 different concurrent data sources independently from their type, including a variety of structured and unstructured data sources, such as, relational and object relational databases, XML, flat files (csv or tab-delimited among others), various multi-dimensional data sources, message queuing and various others, using Oracle Data Integrator, Oracle Streams AQ, and Oracle Golden Gate (OGG), encompassing significant heterogeneous workflow from both virtual and physical databases, including RAC. Oracle Data Integration model is flexible and an ideal to work with Oracle Sun appliances such as Big Data, ZFS, Exadata, Exalogic, and Exalytics appliances, and Oracle Solaris SuperCluster T4-4, as it provides seamless designing capabilities through fast bulk transformations. Besides, the Oracle integration model is quite versatile in interaction with other tools such as Oracle Golden Gate, allowing for fast Advanced Replication, including both full replication and thin provisioning. This research will both demonstrate modern E-LT ODI and OGG capabilities live and discuss the enormous benefits of adopting this model. As part of the Oracle Fusion Middleware, this modern approach becomes extremely hot-pluggable. And Oracle Data Integration model works well along with Oracle WebLogic, BPEL Process Manager, Hadoop, TopLink, JDeveloper, and Oracle Identity Management.

1. Business Framework: Concepts and Concerns

1.1 Workflow

According to the Workflow Management Consortium, workflow "represents the automation of a business process, in whole or in part, during which documents, information, or tasks are passed from one participant to another for action, according to a set of procedural rules." Some other somewhat related standards include the expansion of the static modeling IDEF1X into dynamic business process modeling standards such as IDEF2 and IDEF3.

Indeed, this dynamic perspective of what workflow represents is quite congruent with Oracle Grid Computing paradigm, which considers process automation as one of its important components in conjunction with provisioning, and resource pooling, among others. These principles are critical to the determination of the structured and analytic business intelligence workflow, typically associated with data integration for data warehousing purposes.

1.2 Workflow and Data Integration Modernization goals

The key goals to attain workflow optimization in an explicit and strategic fashion encompass not only the dynamics of process automation, but also the approaches to deriving and channeling data sources in an agile way to build data warehouses that are meaningful in nature, and consistently representative of the aggregated and integrated overall data, for statistical, analytics, metrics, forecasting, data mining, and business intelligence purposes, in general, while attaining an optimal level of data quality.

Furthermore, big data workflow and integration should convey the idea of being congruent with the organizations needs to manage data in a more agile way, congruent with other business processes and related metrics, including the Software Development Life Cycle (SDLC), Information Life Cycle Management (ILCM), and the ability to plan on capacity planning, tiered storage archive model, and other resource management that can either be recycled or subject to a round-robin algorithmic or other bounded cyclic resource control model.

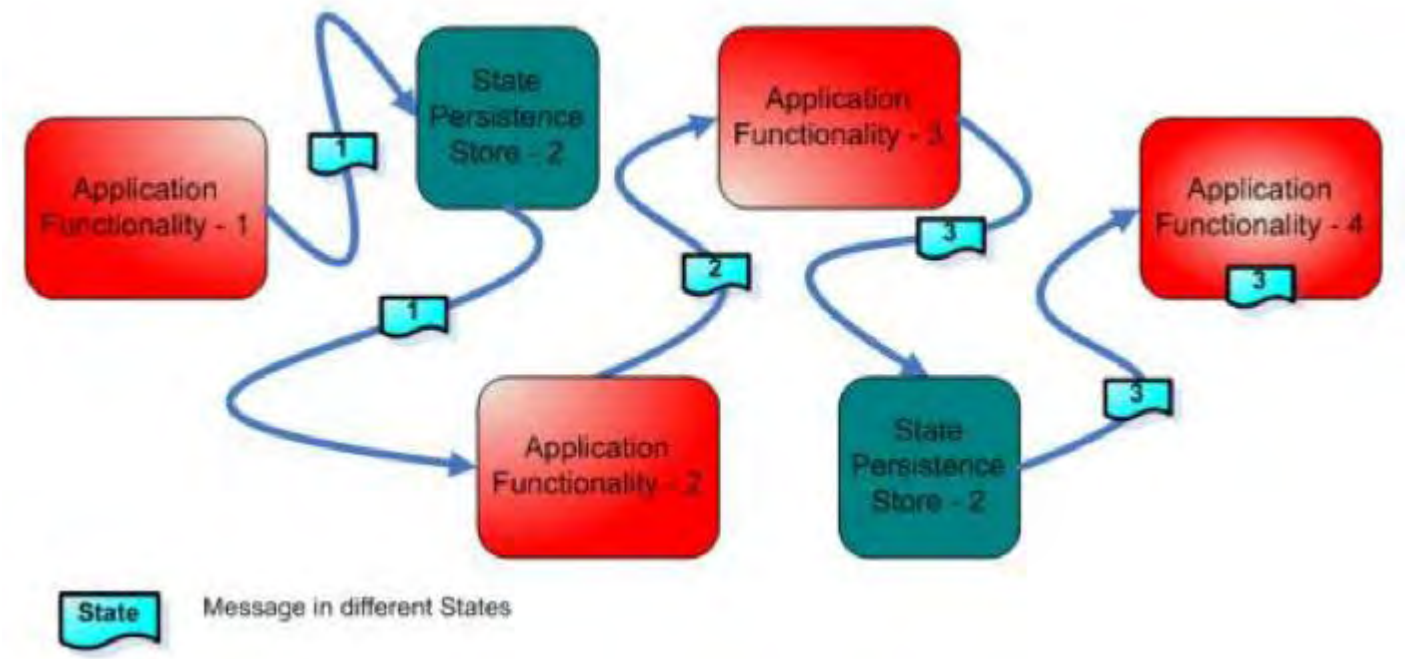


Exhibit 1. Workflow Dynamics

1.3 Private Cloud vs. Public Clouds Workflow Integration

The customization of existing resources are driving cloud computing in a rather transparent way, due to the appropriate usage of file systems, physical, virtual, and hybrid, in alignment with storage networking innovation, clusters, and related topologies and protocols. So, the only criteria from deciding whether a corporation should decide on private cloud over a public cloud is very likely to be data security and privacy, since workflow can included archiving data logging possibly proprietary and intelligence information.

1.4 Private Cloud vs. Public Clouds Data Integration

The integration of data that is apparently volatile within public, private, and hybrid clouds can actually be simplified by the usage of MapReduce, message queuing and data connectors models, which enable transparent gateways through the diversity of file system, operating system platforms, DBMS, and storage models. So, only the planning and costs associated with the actual architecture can determine a comprehensive integration model.

1.5 Brainstorming on Workflow and Data Integration Convergence

Due to the impact of cloud computing, technical challenges on innovating workflow and data integration suggest that it is no longer possible to think only on the software technologies and business processes involved, but also that the hardware involved and their resources dimensions, in terms, of storage, virtualization and other provisioning features, as well as their integration with key middleware components. This is not only because of the implementation and customization of appliances solutions, but also the expected effect of subsequent upgrades and migration in conjunction with the convergence of heterogeneous technologies, as well. The goal is to be able to create a business integration workspace capable building the necessary orchestration to allow for streamlined workflow communication in order to attain true data integration through collaboration, with enhanced coordination and comprehensive overall control through the cloud landscape. This means that all tiers at each stage propagate consistent streamlined coordinated workflow generated by big data applications, and allowing integration via business intelligence and smart data warehousing processes. Whether private or public the cloud workflow and data integration does not depend on the restrictions of firewalls and security protocols and settings, but in the continued, comprehensive interaction of data manipulation, automation, provisioning, and coordination and integration processes, which transparently permit true data integration. Besides, message queuing

technologies, such as Oracle AQ and IBM Websphere MQ, and others relevant such as Oracle Golden Gate, Hadoop MapReduce, as well as others comparable, allow for the structured mapping and propagation of big data workflow within a large organization and beyond, in particular, in inter-organizational processes, such as those associated with B2B and B2C e-business processes, with a variety of topologies supporting those business processes and their actual routing. Technologically speaking, it is expected that new tools and utilities will serve as resources enabling hybrid file system integration with virtual file system through transparent and procedural gateways, and channels; and also these allow the integration of already existing technologies such as ASM (including ACFS), OCFS, GPFS, Veritas Cluster, CIFS/SMB, NFS, and several others through the appropriate protocol integration. Thus, physical and virtual databases, and involved middleware can attain visibility, coordination, and collaborate in truthful data integration. So, modernizing workflow and data integration does involve envisioning collaboration and coordination through cloud computing environments with flexible topologies, hybrid file system, message-oriented middleware, networking storage, and overall resource networking with a clearly defined goal-seeking mission. This structured approach can drive the implementation of both public and proprietary methodologies with models that are both bounded and controlled.

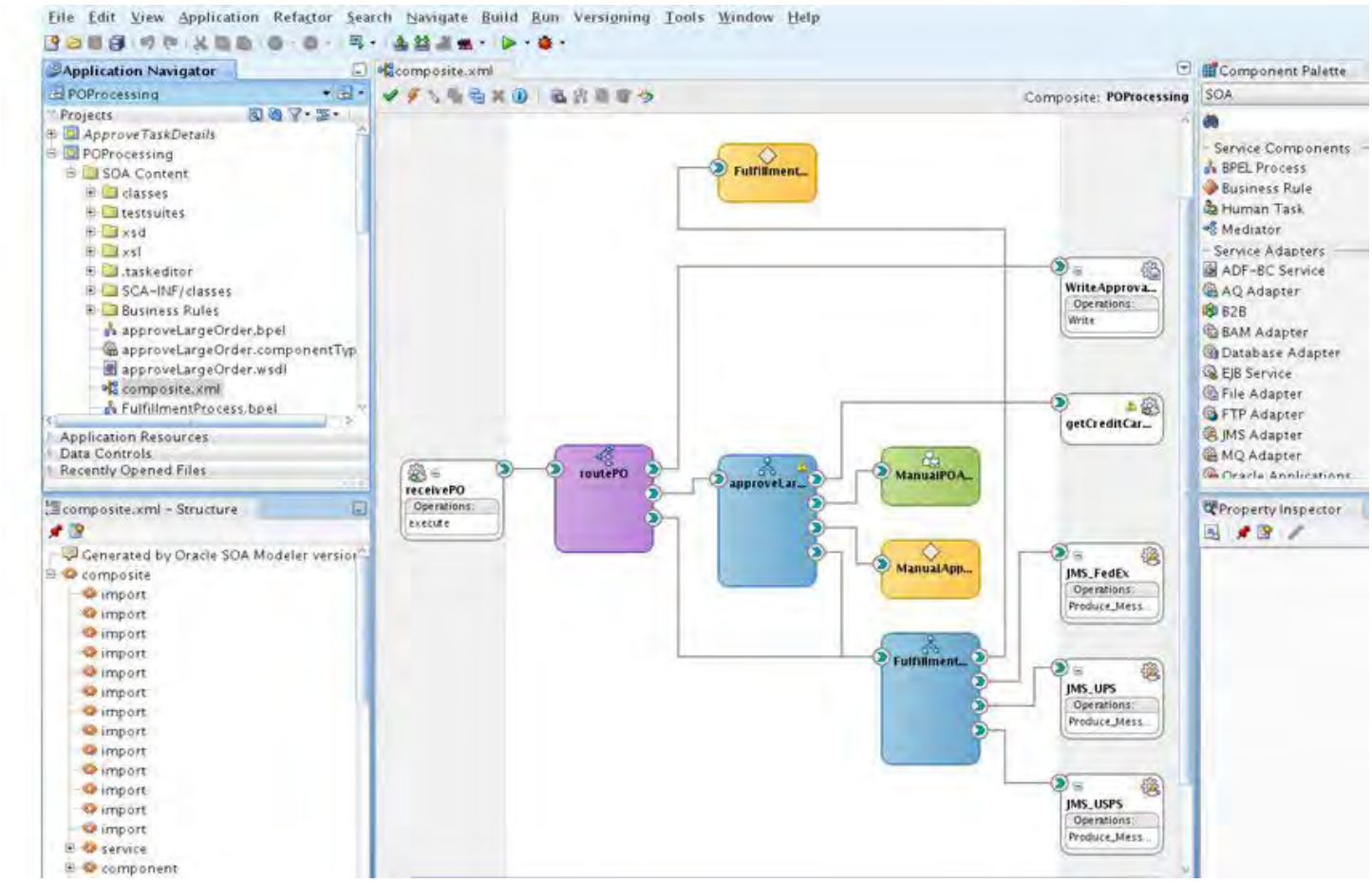


Exhibit 2. A Business Process Model Can Be Established as BPMaaS

1.6 E-Business Process Innovation

E-Commerce will be highly impacted by the modernization of workflow and data integration. Thus, B2B and B2C will provide the ability to make direct usage of SaaS, PaaS, or even IaaS in some specific scenarios, beyond the scope of this paper. Furthermore, BPMaaS will be in itself a strong component of the application of this modernization directly targeting e-business process innovation.

2. Technical Framework

The following concepts are fundamental to understanding workflow and data integration in the perspective of the Oracle Fusion Message-Oriented Middleware and the underlying relevant data integration technologies, including ODI, OGG, and AQ, as follows:

2.1 Oracle Data Integrator

Oracle Data Integrator is an Oracle Fusion Middleware component that allows to overcome challenges common to all data integration and transformation projects such as Checking and improving the quality of your application data and accurately and easily exchanging data between your applications in traditional ETL and bulk E-LT while respecting the business rules of your information system. ODI supplies an analytical framework to:

- Implement data quality controls to check data in a database and ODI repository
- Create unification interfaces to perform ETL and E-LT tasks, namely, move and transform data
- Automate the execution of these interfaces into packages
- Execute the package and review its outcome
- Prepare the developed components for deployment

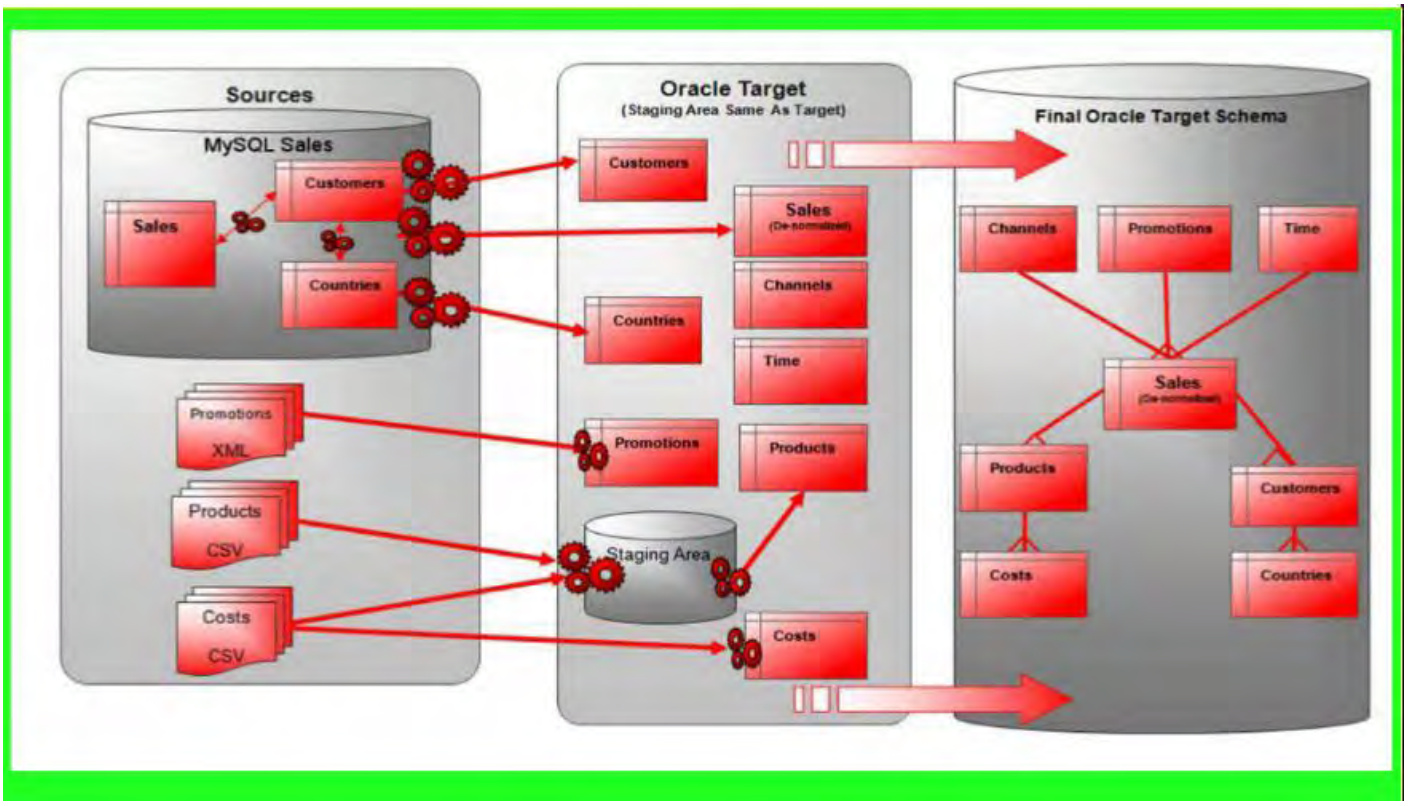


Exhibit 3. Oracle Data Integrator Basic Workflow

2.2 Oracle Golden Gate

As a component of Oracle Fusion Middleware, Oracle Golden Gate provides real-time data integration to mission-critical applications supporting improved business process models, query offloading to maximize OLTP performance, zero-

downtime data migration, backup and recovery, and active-active database synchronization for continuous high-availability.

2.3 Message Oriented Middleware (MOM)

A middleware architecture and infrastructure that utilizes advanced message queuing, and other messaging technologies, as part of its architecture core components, integrating information through both front-end and back-end architectures.

2.4 Streams AQ

An Oracle technology integrating message queuing and capture-propagate-apply downstream replication, event/alert management, and workflow models.

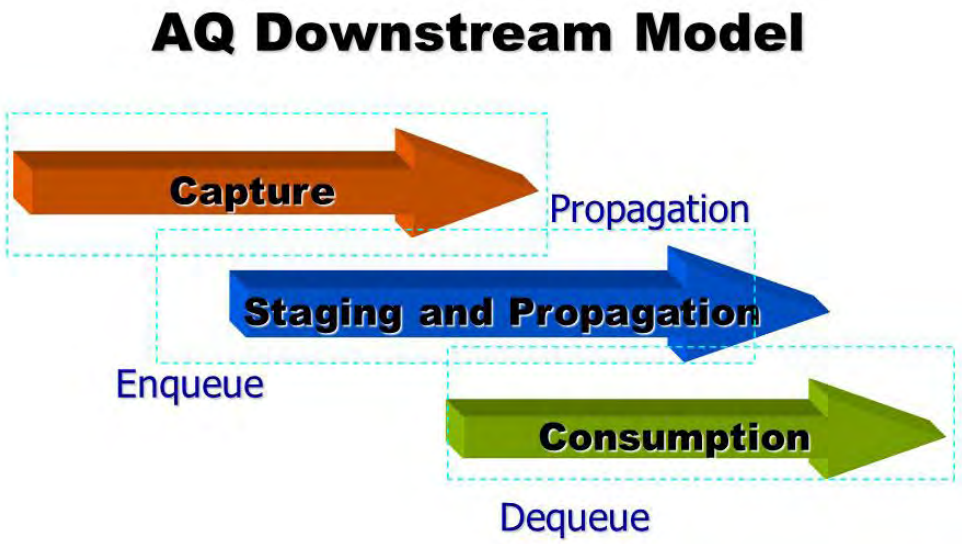


Exhibit 4. Oracle Streams AQ Basic Workflow and Replication Model

2.4.1 Queue

A First In First Out (FIFO) data structure, i.e., specifically, a type of list. In advanced message queuing technology, queues can be persistent or non-persistent; normal or exception; transactional or non-transactional; with or without priority; regular messaging or buffered messaging; and based on their payload datatype, they can be of anydata, xml, lob or raw (default) data types. The ANYDATA datatype is very useful since it requires no validation, such that neither a producer enqueuing the message nor a consumer dequeuing the message needs to deal with any payload strongly-typedness concerns at any propagation stage.

2.4.2 Message Queuing

A technology that permits message transmission via a virtual channel from one queue to another queue, or a database link, thus allowing information integration to occur in different e-business models.

2.4.3 Header

A set of features associated with the message set atop and accompanying its payload, by passing the first and most necessary information identifying the message.

2.4.4 Payload

The actual content of the message placed in the queue, *i.e.*, the message content itself.

2.4.5 Channel
A virtual communication path tuning a queue connection, sometimes called a topic. The channel is more relevant when using foreign queues, such as Websphere MQ or Tibco.

2.4.6 Port
The port associated with the queue listener and the involved network connection.

2.4.7 Enqueue
The activity through which a producer application places a message onto a queue. The enqueue process is the mechanism producing the message and setting it into the queue.

2.4.8 Propagation
The process through which a message stages and is moved from a producer queue, where it has been enqueued, to a destination queue, where is consumed by a consumer or recipient. Propagation is sometimes called *Staging* generically and imprecisely, which is not significant, for it does not convey the idea of travelling from one queue to another one.

2.4.9 Dequeue
The process through which a consumer application removes a message from a queue. Usually after dequeuing from a normal queue, the mechanism places the removed (consumed) message onto an exception queue. Therefore, the dequeuing process actions are associated with the mechanism consuming the message.

2.4.10 Producer
An application placing messages onto a queue, *i.e.*, performing an *enqueue* operation.

2.4.11 Consumer
An application that removes or consumes a message from a queue, *i.e.*, it dequeues the message from that queue.

2.4.12 Recipient
An explicitly-designated consumer. The term is widely used in *publish/subscribe mode* when utilizing multicasting, and more explicitly when an inbox or outbox is explicitly established for the recipient.

2.4.13 Agent
An intelligent connectivity entity associated with a listener, a database link, and a relevant queue acting as a liaison to database and both local and foreign queue technologies. The agent establishes a binding relationship with a queue manager as well, via a protocol such as IPC using the gateway-established channel.

2.4.14 Queue Manager
The queue administrator handling inbound or outbound traffic through an established gateway interconnect channel. The concept is usually associated with foreign queues as well.

2.4.15 Logical Change Records (LCRs)
An LCR is a message with a specific format that describes a database change. There are two types of LCRs: row LCRs and DDL LCRs. A capture process, a synchronous capture, or an application can capture LCRs.
The following types of LCRs can be captured with Oracle Streams:
A captured LCR is an LCR that is captured implicitly by a capture process and enqueued into the buffered queue portion of an ANYDATA queue.
A persistent LCR is an LCR that is enqueued into the persistent queue portion of an ANYDATA queue.

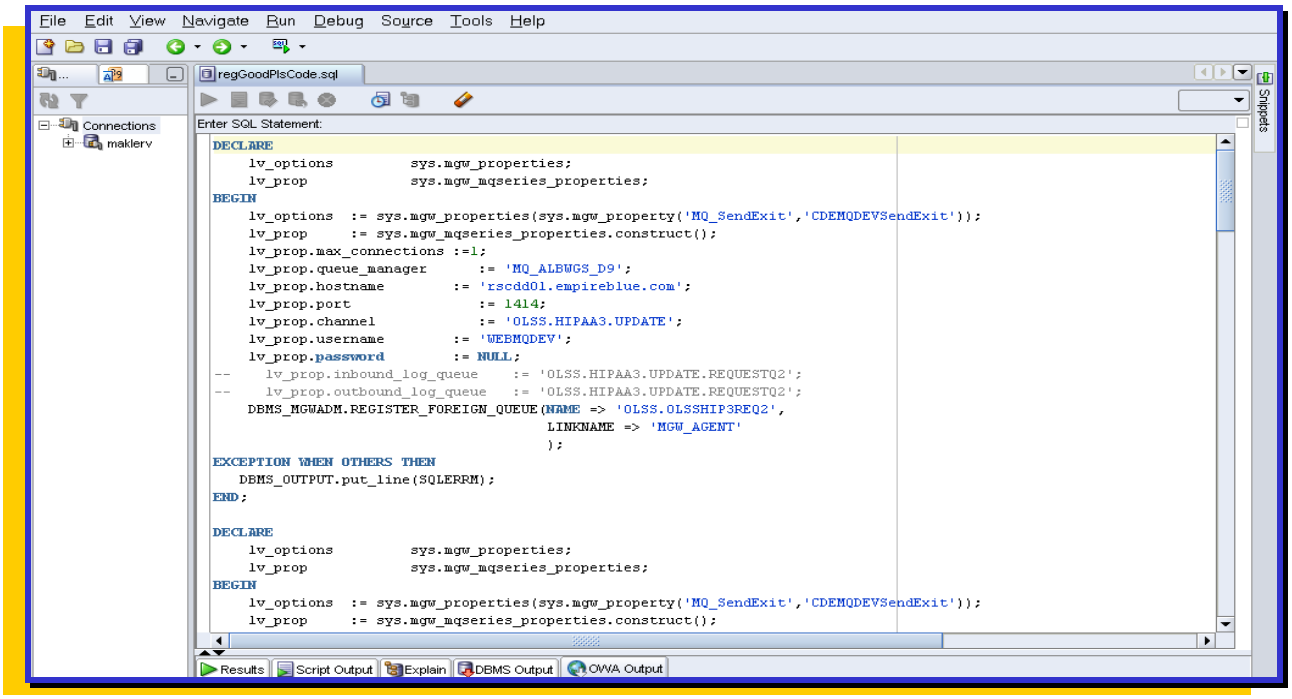


Exhibit 5. Streams AQ to Websphere MQ PL/SQL Code

2.5 Architecture and Queuing Models Highlights

In particular, the following MOM architectural models are introduced in accordance from their messaging approach, and further characterization is discussed about manageability, security, and advanced capabilities and administration. Although queuing models can be classified based on many issues, such as workflow characterization, payload type, replication type, data capture type, and many others, which could easily be determined by simple analysis.

2.5.1 Cardinality-based Queuing Models

On the basis of the queues' relationship cardinality model used, it is possible to derive the following taxonomy, as shown in Table 1:

QUEUE MODE	DESCRIPTION	CAPABILITY
Peer-to-Peer (P2P)	The message is sent from one queue to another or to foreign queue via a database link. There could be more than one receiving queue but only one can dequeue the message at one time.	Only one consumer application can dequeue the message.
Publish-Subscribe	This is a one-to-many model, which uses a subscription for recipients.	When the number of consumers is unlimited it is called broadcasting. When that number of recipients is limited it is called multicasting.

Table 1. Cardinality-based Queue Mode Models

Likewise, with the introduction of the Enterprise Service Bus and Exabus in Exadata technology, application architectural consideration can project the Model View Controller (MVC) more closely by breaking them down into *domain*, *data model*, *and data source* (Model); its *presentation* (front-end view); and its *controller or mediator*, thus establishing architectural fundamentals for grid computing implementation.

2.5.2 Administration and Manageability

According to the available Oracle literature, an application (*producer*) can enqueue messages that represent events into a queue explicitly, or a Streams capture process can capture database events and encapsulate them into messages called LCRs, resulting from DML or DDL changes. Propagations can spread messages in a stream through multiple queues. Finally, a user application (*consumer*) can dequeue messages explicitly, or a Streams *Apply* process can dequeue messages implicitly. Similarly, an Apply process can re-enqueue these messages explicitly into the same queue or a different queue, if necessary.

A database schema user can be a conventional user with the AQ_USER_ROLE privilege or a privileged administrator with the AQ_ADMINISTRATOR_ROLE privilege.

Similarly, Oracle10g and Oracle 11g databases handle and tune the *qmnc* master process and their *qxxx* worker processes, in particular, when the AQ_TM_PROCESSES initialization parameter is not set. Besides, the STREAMS_POOL_SIZE parameter default setting is about 10% of the shared pool size.

An agent is established in connection with an *extproc* listener and a gateway home, when foreign queues or heterogeneous services are in place. The configuration of the LD_LIBRARY_PATH, and the installation of the Oracle Procedural Gateway are required with MQ and Tibco, while the Oracle Transparent Gateway is needed in Heterogeneous Services, like Microsoft MSMQ and SQL Server. The Oracle Procedural Gateway will normally allow for the start of the gateway without further configurations in a peer-to-peer mode, but it requires, otherwise, specific gateway home settings for each foreign queue established concurrently. For further details concerning the Oracle Procedural Gateway, refer to Metalink notes 188833.1, 212587.1 at http://support.oracle.com. Other recommended notes include 198523.1 and 238070.1.

Among the most important PL/SQL supplied packages for this purpose, we can include DBMS_AQ, DBMS_AQADM, DBMS_AQJMS, DBMS_MGWADM, and DBMS_MGWMSG. Equivalently, a good number of Java Messaging Services (JMS) Java supplied packages is available and can be used concurrently or in lieu of PL/SQL.

2.5.3 Security Framework

Oracle's queuing security framework is based on Streams rule-driven security at the object, schema, and global levels. However, it can also provide Virtual Private Database (VPD) and security can be further enhanced with realms set via Oracle Database Vault. Encryption models based on asymmetric authentication and the Public Key Infrastructure (PKI) can allow Oracle PL/SQL supplied packages such as DBMS_CRYPTO encrypt payloads prior to being enqueued and allow for their decryption via the public key upon consuming the message at the target queue (dequeue). This is widely supported primarily via anydata, xml, and lob data types. Oracle Advanced Security can provide tier-based security through several layers in the Oracle stack and the networking level overall. This represents a significant advantage over Oracle Golden Gate constraints, and another solid possible data source integration resource for Oracle Data Integrator instead custom enhancements to this technology involve the appearance of the One-Time-Public Key Infrastructure (OTPKI), which uses a one-time short-term certificate model for environments with mission critical privacy and security requirements. Besides, when used with Oracle11g Transparent data encryption (TDE), TDE stores the master key in an encrypted software wallet and uses this key to encrypt the column keys, which in turn encrypt column data, as needed, which beneficial for the model security.

3. Data Warehousing with Heterogeneous Data Sources

For many years already, data warehousing has utilized many different approaches to accommodate its workflow through diverse types of data sources and a variety of strategic transformations, and using many technologies, tools, and interfaces. Historically, the data warehousing business model evolved originally from many brainstorming approaches to what has been called ETL (Extract Transport Load), which enticed the implementation of a data warehousing and in some instances an analytical workspace as an expansion to business intelligence. With the innovation of bulk processing, it is most commonly practical to use the Extract, Load, and Transform (E-LT) model, which provides unlimited capabilities to

support many data sources, and how, and where, and when the data can be either loaded or transported, and in how many different ways it is possible to accomplish this process. Oracle Data Integrator provides seamless capabilities beyond its interaction with the entire conglomerate of possible data sources, and it is geared towards the intelligent dynamic manipulation of conceptual, logical, and physical data models, and their expansion into a multi-dimensional analytical workspace landscape, supporting BI for general analytics, forecasting, and decision making, and consistently integrated with Oracle intelligent middleware, driven by message queuing applications, including Weblogic application server, J2EE, web services, and various others.

4. ETL vs E-LT Techniques and Applications

While ETL provides capability for single row-by-row extraction and transformation processing, E-LT provides an unlimited capacity for bulk load and transformation, stages that can occur concurrently. The advantage of E-LT over ETL is that the aggregated transformations and loads can occur in many possible ways and in any convenient order, as required by the business process model.

ETL and E-LT techniques can be both driven through batch processing via the Oracle Golden Gate command line interface, as shown in exhibit 6, or through an IDE tool, such as, Oracle Data Integrator Studio.

Exhibit 6. Replication-Driven Data Integration with Oracle Golden Gate

4.1 Seamless Bulk Transformations

Seamless bulk transformation means that the load can proceed using the bulk usage of methods capable of multiprocessing many rows at one time. This is significantly important since single row processing tends to become time consuming and deal with various systematic I/O scenarios, most of which are naturally overcome using bulk processing, and the appropriate timeliness to control and commit a transaction.

4.2 Supported Data Sources

The supported data sources included text flat files, MIME format files such as Microsoft Excel, Access data files, XML, Oracle dump files, Oracle distributed databases, including but not limited to ASM-driven and RAC, involving database running on both physical and virtual machines environments and networked and integrated via hybrid file system technologies. Oracle Streams AQ message queuing technology also supports any possible data type. This is attained through either a user-defined data type definition or using the ANYDATA, as specified for the payload, which has essentially no exclusions.

4.3 E-LT Bulk Transformation Capabilities

The usage of Extract-Load and Transform or E-LT, provides the capability to perform bulk data extraction followed by bulk load followed by loading and transformation as needed. Oracle Data Integration approaches using Oracle Data Integrator and Oracle Golden Gate.

5. Integrated Use of Oracle Golden Gate and Oracle Streams AQ

While Golden Gate provides a variety of user-oriented options it has some data type limitations that do not exist with Streams AQ, which excels by supporting essentially any data type through its message queuing capacity, within Oracle technology and in interaction with other technologies, such as IBM Websphere MQ, Tibco Rendez-Vous, and other Tibco products, and Microsoft MQ, among other third-party vendors. However, the ability to automate, provision, and replicate in a substantially easier way is led by Golden Gate. The ability to coding and interact to other technologies in a more granular way from the programming perspective is in the advantage for Streams AQ message queuing technology. Deciding whether to use one of Oracle Streams AQ, Oracle Data Integrator or Oracle Golden Gate or any possible combination is a decision that relies of important constraints such a budget, company technology direction, scope of planning, investment and ROI, and specific company needs in relation to their size, security requirements, infrastructure and topology in place, and approaches to their corporate information technology convergence.

6. Oracle Golden Gate Data Integration with other Middleware Tools

The workflow generated by Big Data in many fashions, whether application workflow, such as transactional workflow, and that generated by logs, mail, social media, mobile communications, document processing including images, and various others can expand to the variety of files formats, markup languages, and processing protocols. Oracle middleware technologies in conjunction with business intelligence tools are the key players in the custom implementation of methods and business process models to allow big data integration, organization, analysis, and mining for optimal coordination and agility.

Similarly, as an Oracle Fusion Middleware component, Oracle Golden Gate features support for high-volume systems, global implementations and heterogeneous platforms, while enhancing data capture through Extract-Load-Transform (E-LT) in association with Oracle Data Integrator and enhanced security. Besides, Oracle Golden Gate works well in conjunction with Oracle Exadata, Oracle Enterprise Manager 12c (Management Control as a Service or MCaaS), and ERP, through Oracle Applications, as applicable to the Oracle Integration Architecture (AIA), and third-party, like SAP. Some of Oracle Golden Gate 11g R2 release new features involve:

- **Integrated Capture**
  Enabling customers to efficiently capture changed data in high-volume and high-throughput implementations.
- **Oracle Advanced Compression Support**

Permitting the capture and delivery of compressed objects within Oracle Database 11g and Oracle Exadata environments, an important feature for virtual and cloud environments.

- **Best in class performance**
  Providing the following features:
  - More Powerful Conflict Detection and Resolution, thus, enabling customers to reduce the time spent on conflict resolution.
  - Increased Security, support for the Federal Information Protection Standard (FIPS) and Blowfish encryption algorithms enabling secure data movement across systems and regions through clouds.
  - Increased support for business-critical and heterogeneous systems, including multi-byte Unicode Support
  - Expanded Platform Support: Provides support for capture from, and delivery to, IBM DB2 on iSeries and enhanced support for MySQL, Microsoft SQL Server, Sybase, Teradata, and IBM DB2 z/OS.
  - Expanded management capabilities via its Plug-In for Oracle Enterprise Manager 12c (MCaaS).

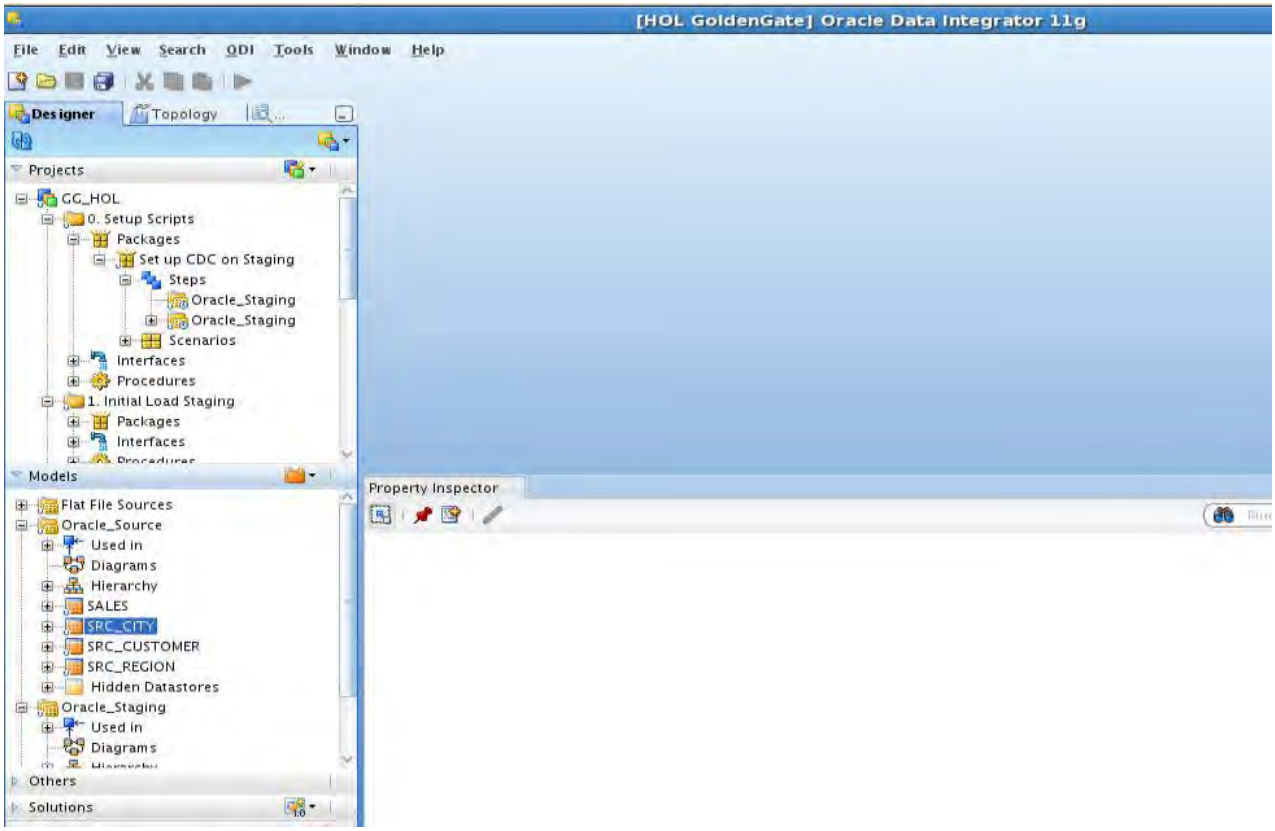


Exhibit 7. A Data Integration Model Using Oracle Data Integrator Studio

7. BI Datamarting with Oracle Data Integrator and Oracle Golden Gate

The conjoint usage of Oracle Data Integrator and Oracle Golden Gate provides a major data integration model, whose impact and benefit on business intelligence are rather unlimited. Some of the most important areas of application involve, namely:

- Business Intelligence Analytics
- Data Mining
- Forecasting
- Cloud Virtualization resource management

Finally, virtualization approaches to apply both workflow and data integration, mostly derived from big data applications, conveys a variety of possible selections, from the leading quadrant including VMWare, Microsoft VM, and Citrix, and Oracle Virtual Box from the challenger quadrant in today's virtualization market. Thus, virtualization and provisioning are the driving force of this workflow through cloud at a fraction of their traditional historic cost.

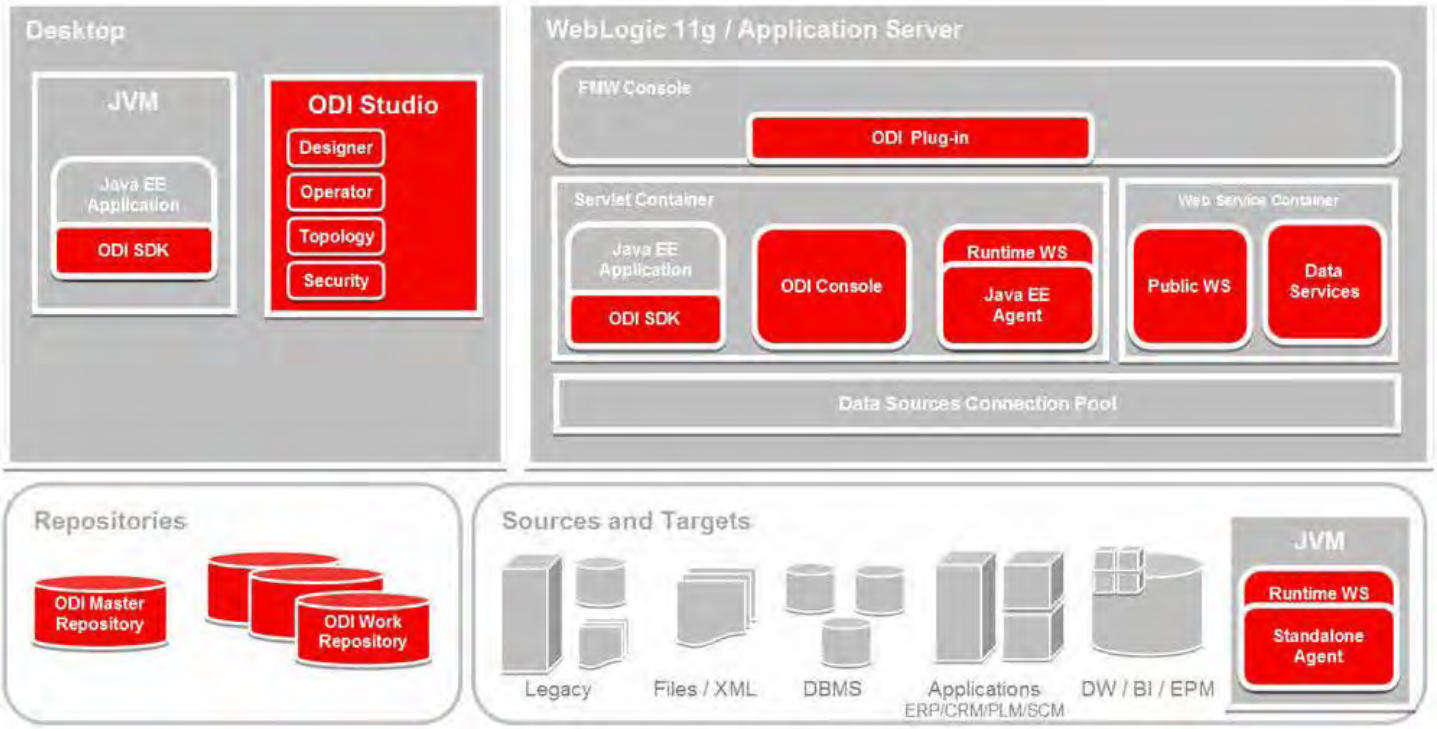


Exhibit 8. Data Integration Workspace Using Oracle Data Integrator Architecture with ODI Studio

8. IT Convergence: Appliances and Storage Networking

Providing guide to hardware selection to match and optimize computing, storage networking and networking resources on Oracle Sun product line, in particular, with workflow and data integration as a goal, is a task that requires product knowledge and extensive experience on Oracle and Sun product line, and expertise with a specific objective and independent perspective. Enhancing the matched resources and capabilities, whether you use a comprehensive middleware, extensive virtualization, or emerging files systems technologies, such as Hadoop HDFS, and underlying technologies such as partitioning and Hybrid Column Compression (HCC), you have the opportunity to analyze, establish, and further research an expert's opinion with a different viewpoint with a comparable potential for a successful infrastructure and implementation for optimal performance, reliability and availability, and subsequent deployment. Therefore, it is certainly a good time for a major investment on information technology to walk on the cloud. It is time for a change! Thus, providing a custom independent guide to match and optimize the use of Oracle Sun systems including appliances, high-availability, and storage networking solutions with respect to OLTP or Data warehousing and Business Intelligence Solutions, with a futuristic view on technology sustainability and predictable economic trends.

If you information technology infrastructure hinders a focus upon data warehousing, there are various solutions all of which could meet the DBA desires to have the best storage, storage networking and main computing resources, such as CPU power, enough RAM memory, and high quality flash and solid state storage (SSS) with convenient capabilities to attain the optimal custom solution. Whether you are looking at Exadata, the Oracle Database Appliance, the NoSQL Appliance, or the Oracle ZFS Storage Appliance as a potential solution, your investment will pay back on reliability and performance for the next few years, opening a door to easy upgrades to new versions, editions, and the next generations of database technologies. Exalogic is appropriate to attain the perfect middleware goals, and Exalytics can drive the appropriate built-in resources to attain the best desirable outcome for business intelligence, data mining, data discovery, and analytics, in general. The implementation of *in-memory* database machines (using Coherence or Times-Ten), with

Oracle SQL or NoSQL, enhances the ability to integrate with new file systems, such as Hadoop HDFS, through database connectors, and the ability to implement associate *MapReduce* maps sets and derive intelligent reports. The amount of hardware resources customizing high-end resources at any level, and the ability to leverage those resources for the best return on investment (ROI) and extended return on assets (ROA) at the lowest possible total cost of ownership (TCO) in today's market.

Because, these appliance and engineered systems have been designed to work together both CAPEX (Capital Expenditure, i.e., investment) and OPEX (Operational Expenditure) are significantly optimal in today's information technology market. Equivalently, the same remarks could be applicable to online transaction processing (OLTP) systems, where the Exadata database machine and the Oracle Appliance can provide a significant level de customization for a dedicated or hybrid purpose.

In general, in their search to optimize cloud big data integration, companies regardless of its size can successfully consider an important investment in IT at the moment, which could bring this sector to its best moment in ten years, and have more confidence than ever that their ROI.

In essence, what is important is to plan a recommended acquisition quite ahead of time, considering, importance concerns such as capacity (memory and storage), and for each one customize every technology available; as such for the former, what amount of conventional RAM and flash memory will be required, so you could decide on a quarter, half or full Exadata machine; or customize your RAC on a third-party hybrid solution choosing the appropriate Solid State Storage (SSS) technology, but always looking about the convenience of systems that are engineered to work together; likewise, for the latter, carefully research what are your options for secondary storage (e.g. the ZFS Storage Appliance) and tertiary storage (e.g., tape libraries), including customizable archiving models. In terms of storage capacity, Exadata has the ability to easily expand and interconnect the enhanced machines storage networking via Infiband, the fastest and most flexible providing the best bandwidth as well. On the other hand, processor power is no longer the only main factor in attaining optimal database performance, and architects, DBA, IT Managers, and other influencers or recommenders need to look at a variety of factors before a final decision is made. Indeed, there will be plenty of time to make a decision.

Exadata Machine X.2-8

Currently, the most powerful and scalable machine with capabilities for both data warehousing and OLTP databases with leading TPC-H and TPC-C benchmarks.

Therefore, this is the latest Exadata machine version in production environments today.

Exadata Machine X.2-2

Exadata X.2-2 is the traditional Oracle database machine solution, most widely used in the market. The Exadata machines can be either expanded through an Expansion Rack and several machines can be networked through Infiniband, which offers capabilities to an open cloud networking model, in contrast with some market opinion.

Oracle Database Appliance

Exhibit 9. Oracle Database Appliance

The Oracle Database Appliance is the natural path to upgrade your RAC or RAC OneNode environment and utilize a variety of built-in features on the fly, simplifying your future upgrade path as well.

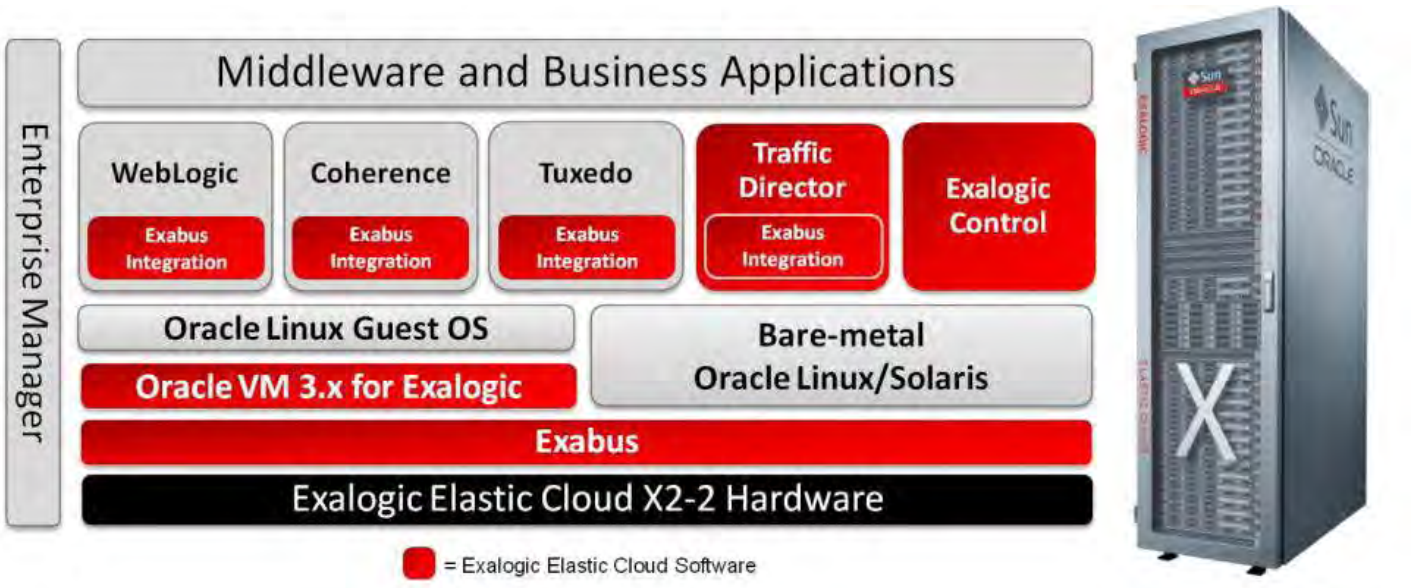


Exhibit 10. Oracle Exalogic Elastic Cloud Software Enhances Exadata/Exalogic Technology

Oracle Sun Servers

Companies that wish to customize database solutions from ground up with native OS, and build their own virtualization environment, either bare-metal or on top of the native OS-layer can do so as well.

9. Future Expectations

Based on the issues previously highlighted, it is expected that the modernization of big data workflow to attain optimal data integration for decision making purposes and business intelligence processes innovation, in general, will involve the following BPaaS concerns:

1. Define an integration model and underlying infrastructure topology
2. Define the set of data sources and data capture methods
3. Define the staging and propagation methods, including channels and workflow buses and baselines, if any is relevant
4. Predict model changes and relevant analytics for best integration
5. Provide a consumption model that is transparent to the cloud platform, operating systems, physical and virtual database environments, file systems, and even storage networking technologies involved. This model should support any of SaaS, PaaS, IasS cloud paradigms.
6. Establish the collaboration, coordination, and enterprise management control models.
7. Maintain a secure big data workflow and integration model with a solid quality of service (QoS) and quality assurance (QA) approach.

But in the end, even if a corporation IT infrastructure, platform, or other software resources is late to implement a cloud-oriented business process model, then the cloud will finally come to them, if they do not reach the cloud soon.

10. Recommended Use Matrix

Category	Feature	AQ	ODI	OGG	ODI-OGG	AQ-ODI-OGG
USER'S SKILLS	Programming Skills	Recommended				Recommended
	Admin Skills		Recommended	Recommended	Recommended	
DATA CENTER SIZE	Large Data Center and Cloud	Recommended			Recommended	Recommended
	Midsize IT environments	Recommended	Recommended	Recommended	Recommended	
WORK-FLOW TYPE	Big Data from Social Media	Recommended	Recommended		Recommended	Recommended
	Transactional	Recommended			Recommended	Recommended
	Archiving and Logging	Recommended	Recommended	Recommended	Recommended	Recommended
	Mail and Document Management	Recommended				Recommended
	Data Type Requirements	Recommended				Recommended

Table 2. Recommended Use for Workflow and Data Integration Middleware Tools

Oracle statement of direction is suggesting that more complex event workflow characterized from Big Data and corporate data integration is placing their focus on Oracle Golden Gate and other middleware components such as Oracle Data Integrator rather than just on Oracle Streams AQ. The matrix shown in Table 2 represents solely the author's opinion, based on his experience and knowledge of this topic, and not necessarily those of Oracle or third-party vendors.

In addition to these selective scenarios, a DBA might recommend Streams AQ over any other solution, when the *capture-propagate-apply* process implies a complexity of datatypes, since there are essentially no exceptions that Streams AQ would not support, and in any scenario data can capture and propagated as ANYDATA type, and then transformed accordingly. The following data types are supported by a Streams AQ capture process, namely:
When capturing the row changes resulting from DML changes made to tables, a capture process can capture changes made to columns of the following data types:

- VARCHAR2
- NVARCHAR2
- FLOAT
- NUMBER
- LONG
- DATE
- BINARY_FLOAT
- BINARY_DOUBLE
- TIMESTAMP
- TIMESTAMP WITH TIME ZONE
- TIMESTAMP WITH LOCAL TIME ZONE
- INTERVAL YEAR TO MONTH
- INTERVAL DAY TO SECOND
- RAW
- LONG RAW
- CHAR
- NCHAR
- UROWID
- CLOB with BASICFILE or SECUREFILE storage
- NCLOB with BASICFILE or SECUREFILE storage
- BLOB with BASICFILE or SECUREFILE storage
- XMLType stored as CLOB

11. Concluding Remarks

Workflow and big data integration innovation and modernization convey the implementation of an agile business process model using ETL and E-LT for faster processing in a sustainable fashion through its life cycle.
While the message queuing paradigm, encompassing many middleware applications, is still in place in many cloud oriented infrastructures, in most instances, due to the extraordinary datatype support, the convergence information technology resources, including hardware, and the need to facilitate the use to attain big data integration and workflow management can inspire and direct larger organizations to focus on Oracle Fusion Middleware components such as Oracle Data Integrator and Oracle Golden Gate for optimal data quality and improved agility.
The advantage of utilizing Oracle Data Integrator in conjunction with Oracle Golden Gate not only facilitates the bulk extractions, transformation, transport, and loading of data from many heterogeneous data sources to and from many databases within the data warehouse, but also enables the strengthening of an already solid data quality model and its underlying methodology.

Two Million Database Professionals Count on One Solution.

Simply the best for Oracle database professionals - Toad 11. Supported by over a decade of proven excellence, only Toad combines the deepest functionality available, extensive automation, and a workflow that enables database professionals of all skill and experience levels to work efficiently and accurately. Countless organizations empower their database professionals with Toad. The best just got better.

Watch the video at **www.quest.com/Toad11SimplyBetter**.

QUEST SOFTWARE®
Simplicity At Work™

NYOUG 2012 Sponsors

The New York Oracle Users Group wishes to thank the following companies for their generous support.

Apress (www.apress.com)
CCH (www.cch.com)
Confio (www.confio.com)
Dell/Quest Software (www.quest.com)
EMC (www.emc.com)
F5 (www.f5.com)
GridIron Systems (www.gridironsystems.com)
Oracle Corporation (www.oracle.com)

Contact Caryl Lee Fisher for vendor information, sponsorship, and benefits