Fusion Forward - an update from Oracle Open World 2011

Lucas Jellema - AMIS Services, Nieuwegein, The Netherlands

The yearly Oracle Open World (OOW) conference that was held last October is an excellent moment to look back, evaluate and then look forward. Questions I try to get answered at the OOW conference include: How are products doing - and product managers? How is technology progressing - which trends are emerging? Where is Oracle on the roadmaps it had announced a year ago? What are the visions and roadmaps it currently plots into the future? How should Oracle's activities guide me in daily life and in tactical decisions? And a fairly generic: Do I like what I see?

It is interesting to see how all of those 40,000+ people attending OOW seem to have a slightly different conclusion to take away from it. Given the various backgrounds, skill sets, ambitions and experiences people bring to the conference and the fact they attend different sessions, listen to different Oracle product managers, visit with different Oracle staff in the booths on the demo grounds and have their opinion influenced by a unique set of bias, anxiety, desire and selective perception, it is only to be expected that widely varying conclusions are drawn.

ODTUG's President Mike Riley commented in an interview on The Server Side that the announcement of Forms 11gR2 was the most important fact to take home from the conference. Given the continuity and ongoing validity of Forms as development platform that this demonstrates, crucial to many ODTUG members, I can see his motivation for that conclusion. Forms is part of Fusion Middleware - and even though it is not the most central element at this moment in time, it is still relevant. As well as a vehicle for many organizations to get started with WebLogic Server - through Forms 11g - the foundation for many more FMW products.

This installment of the Fusion column discusses the past, present and future of Fusion Middleware in view of Oracle Open World 2011. It focuses on the development of application components, user interfaces, business processes and data services.
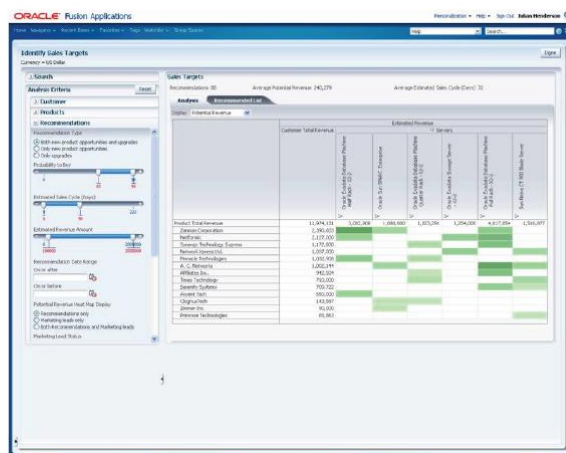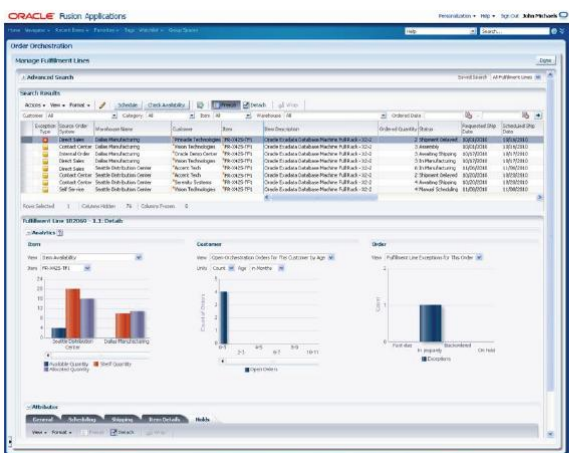
Fusion Applications

The main reason for being for many of Oracle's development tools is the need of its own internal product development groups. The teams that create products such as Fusion Applications. In order to create these massive enterprise applications - that need to bring in the big bucks for Oracle - there has to be an application development platform that enables huge development teams to work productively and consistently and produce an attractive, modern, functionally rich user interface. That application development platform is ADF - and we get to use it too for our application development. This year, even more than on previous occasions, made it perfectly clear that ADF is absolutely crucial to Oracle. Not

only Fusion Applications, but the user interface of almost every other user interface to come out of Oracle - including GoldenGate Monitor, Enterprise Manager 12c, OBI EE, the Identity Management components and the new Composer tools - is created using ADF.

This year's big announcement around Fusion Applications was a little bit disappointing as it was very similar to last year's - at least to the untrained ear. Last year, limited availability was announced to a controlled group of 200 early customers. It sounded however like "Fusion Applications are ready". This year, the announcement was 'general availability to all customers'. Again, it sounded a lot like 'Fusion Applications are ready' - but apparently they are even more ready than last year.

The roll out of Fusion Applications - to on premise installations and cloud deployment - is based on the 11gR1 branch of Fusion Middleware - and it will stay on that branch for some time to come. Fusion Apps will not adopt the 11gR2 release of ADF but probably embrace 12c at some point.

Fusion Applications - as well as the dozens of other user interfaces created by Oracle - set an interesting example of what ADF technology can and perhaps should be used for. Common across the user interfaces are the visually appealing screens, frequent combinations of more traditional data elements and modern data visualizations and a focus on tasks, notifications and exceptions.

 The user is guided to what he should do, what he should know about or should make a decision about. Classic CRUD screens (for standard Create, Retrieve, Update and Delete operations) are becoming far less common than previously in Forms applications. Oracle Open World hosted a substantial number of sessions where Oracle shared its Fusion Architecture and the internal Fusion development experiences and best practices. These sessions are a great help in understand how Oracle thinks its FMW products are best used - and prove how they have been put to the most extensive test any doubtful architect, development manager or developer could ask for.

12c

This year, Oracle staff started to talk openly about the next major release of all things red and shiny: the 12 release. Product releases labeled 12 start appearing late this calendar year - starting with Enterprise

Manager. The beta of the 12 release of the database is commencing this Fall and most product lines start targeting the end of 2012 (perhaps even 12/12/12) as the release time frame for 12.

Ever since Oracle started using the appendix to its release numbers with the *i* first appearing as decoration for the Oracle 8*i* database, there has been some speculation around the next release's letter. After *i* (for internet) came *g* (for grid). The 12 release will be adorned with the letter *c* for chocolate. No, just kidding. It is *c* for *cloud* obviously. Note that in Oracle's mind the grid really also was the cloud - yet primarily focused on scalable clusters of fairly simple hardware units to together form massive systems in the on site data center, not so much on 'pay per use' IT facilities from some place somewhere out there. With the letter c decorating the next generation of products, Oracle's intention to be part of the cloud hype is underlined.

Public C(loud)

Oracle goes to the cloud - the public cloud. At last. The cloud not only provides the tagline for the 12 releases. It is also something will do in a more concrete way. Oracle will offer its own public cloud - a Platform as a Service where users from more or less anywhere can use specific modules from Fusion Applications or deploy their own Java/JEE, ADF and APEX applications to Oracle's cloud environment. The Database Cloud can be reached through APEX, from the Java Cloud and through RESTful data services. Note that REST(ful) services were one of the very popular themes at both Oracle Open World and the co-located JavaOne conference.

The Oracle Cloud will physically be located in Austin Texas, Virginia, near Edinburgh in Scotland (UK) and in data centers in Singapore, India and China. Its logical address is http://cloud.oracle.com. It launch date should be early November - but last minute arrangements could delay some aspects of the cloud offering.

The Oracle Cloud will work with monthly subscriptions, in three tiers of usage patterns: light, medium, heavy. Oracle does not encourage enterprise usage of its cloud and even the *heavy usage* profile is not meant to replace enterprise wide on premise private cloud initiatives. What will be allowed for each profile in terms of storage, data transfer, cpu load, concurrent sessions and Oracle Software product usage is not yet clear. However, the Oracle Public Cloud offers a 30-day free trial. Available exactly once to any organization.

ADF

As discussed above was ADF omnipresent at OOW. In dozens of dedicated sessions, a full and well attended day of the ADF Enterprise Methodology Group and as support act in hundreds of other sessions. And that is probably the most important news around ADF from this conference: ADF's maturity, the strategic position it has acquired within Oracle and the quickly growing demand for ADF expertise outside of Oracle. The fact that ADF applications can be deployed from within JDeveloper (as well Eclipse) directly onto the Oracle Public Cloud is probably another boost for ADF - as this will allow smaller organizations and individual developers to publish rich ADF based applications without the hassle of organizing and administering middleware and supporting infrastructure.

ADF will continue to evolve, for example surfing the Java wave. For ADF 12c this means support for example for Java SE 7, JEE 6 and HTML 5.0 plus CSS 4.0. Continuous improvements in performance, stability and footprint for design time and run time are on the horizon. Probably inspired by the X-ray plugin for Eclipse, JDeveloper will have dependency visualization. It will also have additional Maven (3.0) support. Maven has really been discovered and embraced by the JDeveloper team. JDeveloper will have an embedded Database to allow for even lighter weight development, not requiring an Oracle (XE) database. JDeveloper 12c will be able to use Git for source control, as alternative for Subversion. There is talk of a community edition of ADF that would be free, that would run on Tomcat, Glassfish or other open source JEE containers - with a slightly trimmed down feature set. ADF is expected to gain more support for RESTful services - XML and JSON based. For example the ability for ADF BC Application Modules to expose a RESTful service interface and a more feature rich RESTful/JSON Data Control. JDeveloper will be extended with a real Groovy Editor and Debugger. ADF Faces will have some facilities that will make it better suitable for external web-sites.
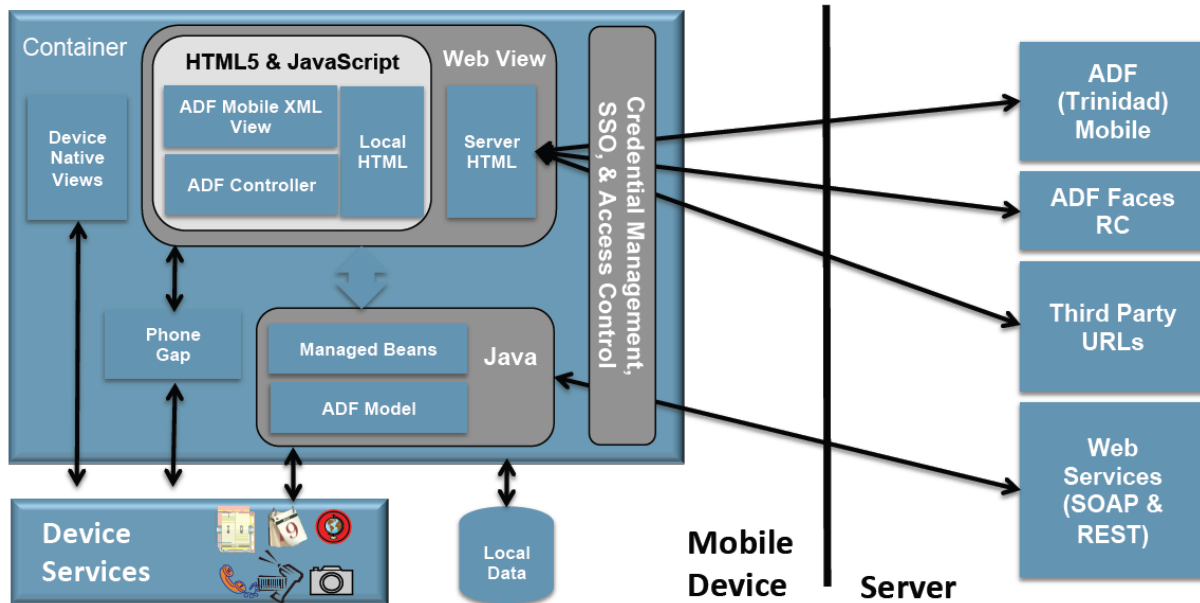
ADF Mobile

The major announcement regarding ADF during OOW was [the new direction for] ADF Mobile. In the past few years, Oracle has provided support for ADF applications on Mobile devices in two ways. One is through browser based applications that use a slimmed-down version of HTML that works well across browsers on mobile devices with varying display sizes and features. The other is using native applications running on Windows 6 and BlackBerry, using Oracle Mobile Server for data synchronization after off-line usage.

This latter approach is dropped and replaced with the ADF Mobile new style. Oracle concludes that in order to provide the type of mobile extensions of enterprise applications that organizations using ADF are typically looking for, it is not necessary to really go entirely native on each mobile platform. It is important to offer a mobile application technology that is appealing and attractive with a near-native look and feel, one that provides access to device specific features such as telephone, GPS, photo camera, contacts and email and one that easily integrates with an ADF back end and allows for productive development with support for all popular mobile platforms, including iOS (iPhone and iPad) and Android. Mobile applications need to be able to function in off-line mode and to recover gracefully once the connection is restored.

Oracle claims to offer all of the above. By adopting the open source PhoneGap framework, Oracle has the foundation for cross-platform applications with generic APIs for access to device specific features. Oracle ADF Mobile comes with a client side framework that runs on the mobile device and works with PhoneGap. This client also contains a light weight Java Virtual Machine with limited functionality. This JVM provides the support for local file system access and on-device persistency (in SQL Lite database) and also for HTTP (and WebSocket?) connectivity to a remote back end, typically in the form of WebServices.

In the ADF Mobile Client, pages can be rendered in multiple ways, supporting hybrid applications that are part native, part local HTML, part locally rendered ADF Mobile XML (AMX) based pages and part remote [server side generated] HTML. The locally rendered AMX pages are very similar to ADF Faces RC pages, but use a lighter weight set of components. An essential part of the visual experience and the underlying rendering is HTML 5 along with CSS 3.0 and the latest JavaScript functionality. Note that the locally running ADF application also uses ADF Controller - with for example task flows - and ADF Model with data bindings - resolving to local beans and remote web services. Eventually there may even be ADF BC running on the ADF Mobile Client.

The development experience for developers creating mobile applications running in the ADF Mobile framework will be very similar to the regular ADF Web Application development experience. JDeveloper as IDE, Data Controls and Drag and Drop Data Binding of UI components as well as use of Task Flows in the visual editors and diagrammers. EL expressions, managed beans and snippets of Java code to fine tune the declaratively created behavior. Very similar, this mobile development experience. The device specific functions that developers will want to leverage in their applications will even be available through an ADF Data Control for drag and drop or programmatic access.

Deployment of such ADF Mobile applications is a somewhat different matter. There will be deployment to the application server - that is known territory - and to the device. That is where device specific actions will be required - but only then, when all development is done. Oracle staff hinted at discussions with Apple that would allow for a distribution model that would by-pass the AppStore for enterprise mobile applications, not targeted at the general public. We will have to await the outcome of such discussions that are of great importance for the success of ADF Mobile. Part of these discussions undoubtedly is the ADF Mobile Client infrastructure and the Java Virtual Machine that Oracle plans at its core. Even though Apple is a firm partner in the OpenJDK project, it may not necessarily welcome Java on iOS.

What certainly is not welcome on iOS is the Flash Player. Why is that relevant? Skip to the next section on ADF DVT.

A story on the value of a framework based approach

This is a true story. And a wonderful proof of the validity of the ADF way of developing applications where developer are typically shielded by the framework from underlying technologies.

One of the Oracle Fusion Applications team decided to certify a module on the iPad. There was an issue with this: the module was rich with Data Visualizations - wonderful and interactive graphics and

animations. These visualizations were created using ADF DVT (Data Visualization Tags) and render using the Flash Player. When the Flash Player is not available - such as is the case on iPads - DVT automatically degrades to static images using SVG or PNG. That is elegant, but lacks animation and interaction. This Fusion Apps team started a conversation with the ADF team about how to achieve the same richness for Data Visualizations on iPad as on desktop browsers - without major rewrites of the application.

The ADF DVT team took on the challenge. They added one additional render engine for the DVT components: one that renders in HTML 5. By having this render engine selected when the user agent mandates it - for example on iOS - Data Visualizations are presented with the same interaction and animation as before, using HTML 5 instead of the Flash Player.

The Fusion Applications team achieved its objective: an iPad ready module with rich and interactive graphics. And it did not change a single line of code in the application. The ADF framework was upgraded (a library replaced) and that was all it took to adopt this new technology.

Interestingly enough: we benefit from that same upgrade of ADF to create data visualizations that run on Apple's mobile devices. And in a similar fashion, we will benefit from a WebSockets based implementation of ADF Active Data Service for Server Push [notifications] that will replace the current long poll implementation in the near future. And other technology upgrades that Oracle will introduce inside ADF.

Data Visualizations with ADF DVT

One of the most visible parts of ADF obviously is the rich collection of Data Visualization Tags. Components to present data in frequently more telling ways than the classical table and form layouts. DVTs in ADF can be created in same simple drag-and-drop data bound fashion as tables and forms. They come in many different shapes and forms - from the basic bar chart, pie chart and line chart to far more advanced displays such as funnel, stock and bubble charts and gantt chart. In addition to traditional charts, the DVT library also contains Map components, the Hierarchy Viewer, Pivot Table, various Gauges and inline Spark Charts. Using DVT it is easy to integrate charts and other visual representations of data in ADF applications - as Fusion Applications proves in every other page. Implementing a rich, visual dashboard as part of an ADF application is very easy to accomplish. Enterprise Manager 12c provides another great examples of this, as does OBI EE that also thrives on the ADF DVT components.
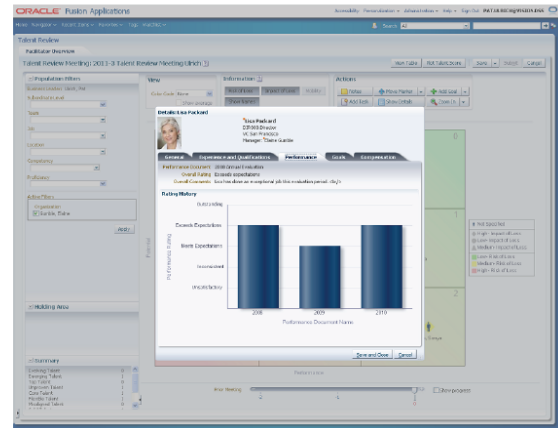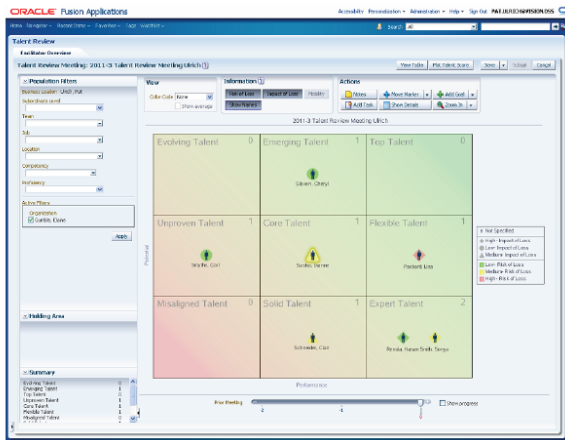
This conference demonstrated the ongoing evolution of the DVT library, driven largely by requirements from various product development teams within Oracle, such the teams working on Golden Gate, Social Network and WebCenter, BPM Process Composer and Human Capital Management. Several interesting new DVT components were announced during OOW 2011, including a Diagrammer, a Timeline, a Treemap and a Sunburst component.

The Diagrammer seems to be a very versatile presentation of objects (icons, shapes) and their relationships on a canvas that allows manipulation by the end user through drag and drop. In fact, very much like any visual editor you work with in JDeveloper or any other tool. The objects, relations and presentation thereof are driven by structured data and can be manipulated programmatically. Typical applications of the Diagrammer include Topology diagrams in Enterprise Manager, Process diagrams in BPM Process Composer and visual representations of social networks.
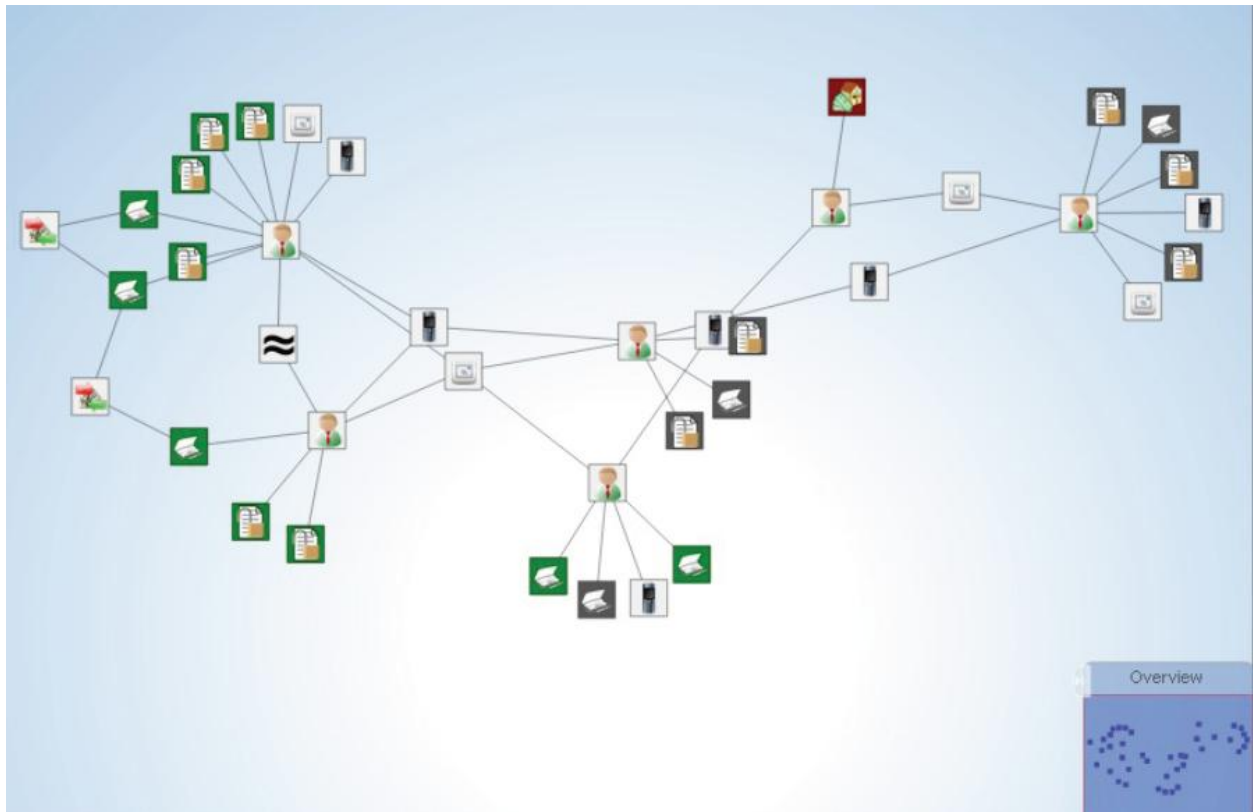
The Timeline - a personal favorite of mine - can be used (as is frequently done in newspapers when VIPs die or major events take place) to present a chronological overview of data. The user can scroll through the timeline, interact with events and zoom to an appropriate level of detail.
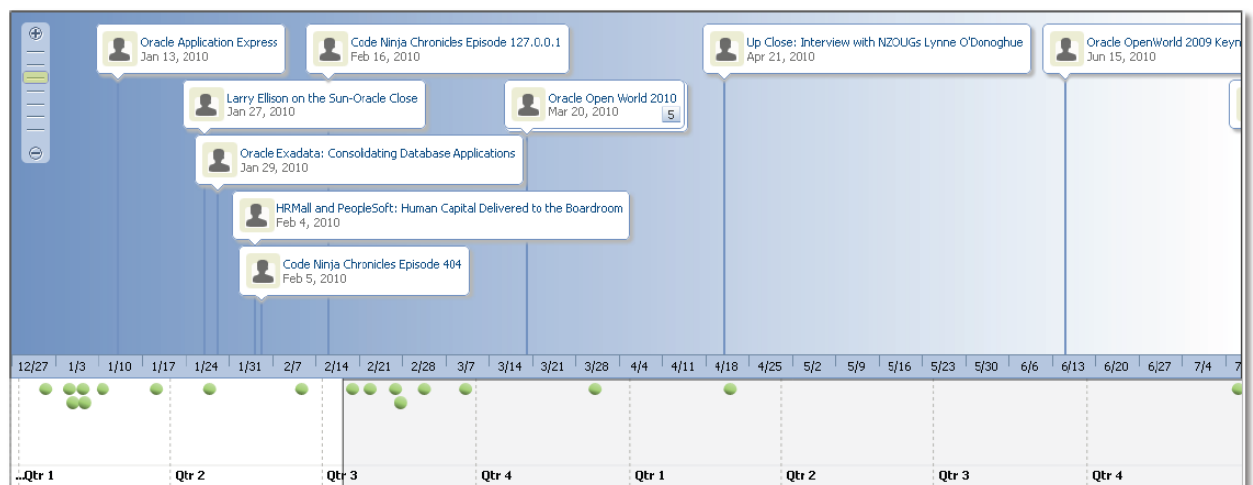
Another avenue explored by the DVT product development team is the further 'activation' of DVT components. Currently, the DVT components support server push as mechanism for instant refresh through notification. However, this is only supported for DVTs bound to a BAM (Business Activity Monitoring) data control. Work is now being done to also support push based refresh on more regular data controls.

Design Time at Run Time through the FMW Composers

In a previous installment of this column, we discussed the concept of design time at run time in Fusion Middleware to give 'power to the end user'. The change persistence and customization features in ADF and FMW at large are exposed through composers like the Web Center Page Composer, the BPM Process Composer and the SOA Composer. This OOW edition made it clear that this Fusion Extension Platform is regarded by Oracle as an important differentiator compared to other technology stacks and as the foundation for a key facility in Fusion Applications. Additionally, design time at run time in combination with customization is essential for multi-tenant SaaS applications where a single instance of an application is used by different user groups - departments, organizations, roles - with each slightly different requirements from the application, its look and feel, its business logic and the business processes.

In the near future, two more composers are to be expected: the Data Composer and the Report Composer. Furthermore, there could be a new product offering from Oracle, sitting somewhere between ADF and Web Center Portal, that contains the Page, Data and Report composer products. Currently, a WebCenter license is required to benefit from Page Composer - which is a huge show stopper for many organizations.

The Data Composer allows run time modification and extensions of Business Components as well as creation of entirely new Business Components. This in turn provides the foundation for adding and modifying fields, adding new details and lookup-popups. Through the Data Composer and Page Composer, a database backed OLTP application can be created almost completely at run time! Note: that is neither Oracle' intention nor a good practice. However, the ability to refine applications at run time, especially with the objective to tailor the application through customizations for specific user groups is potentially very valuable, especially for SaaS style applications.

The Report Composer can be used to assemble fairly simple reports, using the data available from Business Components - either defined at design time or through the Data Composer at run time - and written to various output formats.

JavaOne: Java 7, JEE 7, JavaFX. Java 8 and Beyond

While the Oracle crows was having a ball, only a few blocks away the JavaOne 2011 conference was taking place. This year's tagline for the conference was Moving Java Forward. And that was absolutely the message: Java is continuing to evolve - or in reality after a few years of coma-like contemplation and hibernation, the Java Platform is now on the move, on almost every aspect, tier and application. From JEE (Enterprise Edition) to JavaCard (JVM running inside SIM-cards), everywhere were new releases and ambitious plans for the coming year.

JavaOne 2011 proofs that the apparent decline of Java – as a language, a platform and a community – has been turned around into a steady growth again. Attendance numbers for JavaOne were double those of last (very poor) year. More importantly: the atmosphere was very different from last year. From last year's fairly low spirits with fear, uncertainty and doubt and substantial dose of skepticism with regard to Oracle's stewardship and willingness to invest in areas not (immediately) generating revenue, there now seems to be a growing confidence in Oracle's ability to deliver on promises as well as a relieved realization that Java is rapidly evolving, catching up with current trends – multi touch UI, mobile devices and their native functions, modern programming language constructs, cloud based deployment, today's attractive user interface design, modularization, integration with dynamic JVM languages – and is not yesterday's news and tomorrow's legacy.

The delivery of Java EE 6, Java 7 SE and Java FX 2.0 (GA on Windows, Preview on MacOS), the ongoing strengthening of Glassfish, the convergence of Hotspot and JRockit JVM on OpenJDK, the next steps with Java Card (3.0.4) and ME are all concrete proofs of progress. They in turn vouched for the many new ambitions laid out during JavaOne:


- Java 7 EE – project initiated early 2011, completion planned for end of 2012 – with many goals around cloud deployment, HTML 5 and WebSockets support, new JMS 2.0 spec, increased support for RESTful

services, and many more architectural improvements (note: initial Glassfish 4.0 are already available for download!)

- Java 8 SE – completion planned for Summer 2013 – with improvements from JigSaw (modularization), Lambda (closures), Date Time, a new JavaScript engine in the JVM (project Nashorn); integration of JVMs into OpenJDK; JavaFX 3.0 integrated into Java 8 SE.

The explosion in the number of dynamic scripting languages - (j)Ruby, Scala, Groovy, JavaScript, Jython, ... - that run on the JVM seems to have abated. In past years, developers seemed to want to distance themselves as much as possible from Java proper with their favorite dynamic language. Now it seems that the Java platform itself has been rediscovered and reevaluated. Dynamic languages will be used, but only in niche areas where they deliver special value. They will not replace Java itself, but merely complement. And for its part, the Java platform will be extended to even better integrate with these specialty languages.

- JavaFX future plans- FX is definitely staying! Oracle continues on the investments made by Sun, to provide the next generation of Java Client technology. Multiple platforms, closer integration with SE (already started with the replacement of FX Script with the Java APIs), integration with Dynamic Languages (ScalaFX and GroovyFX); FX is planned for Linux (and Android) and demos were even shown of FX applications on iPad. Any cool User Interface developed in Java and running on desktop or in Applet will be developed in FX, is the objective.
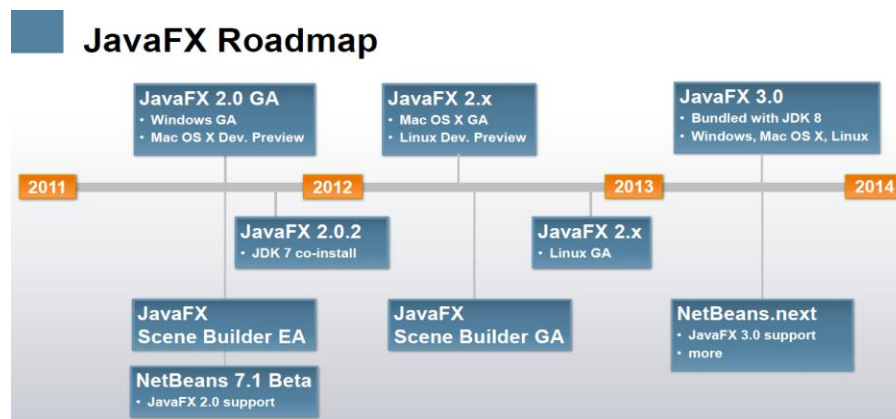
SOA Suite and Oracle Service Bus

It was a relatively quiet conference for the SOA Suite and the OSB (Oracle Service Bus). Big things are brewing - especially for the Oracle Service Bus. But most of the brew will only appear as part of the 12c release. By that time, OSB will be further integrated into the rest of the stack. JDeveloper will take over as design time IDE from Eclipse, currently the home for the OSB development environment. The OSB console, a browser based interface for both development and operational monitoring & administration will in part merge into the Enterprise Manager Fusion Middleware Control. The development facilities in OSB may end up as a stand alone composer or as part of SOA Composer.

The OSB design time will be made to look a lot like the SCA (Service Component Architecture) based approach adopted for SOA Suite, with services, references and wires. OSB will adopt the XSLT based mapping and transformation used in SOA Suite and the XQuery approach currently favored in OSB will also be introduced for SOA Suite. The SOA Suite Event Delivery Network (EDN) will be extended to also hook into the Oracle Service Bus.

JDeveloper will be extended with a (Visual) Debugger that supports debugging both SOA Suite composite applications - including Mediator, BPEL and Business Rule - as well as Oracle Service Bus. This debugger will be similar to regular Java debuggers in as many was as possible. That includes breakpoints, step-by-step walk-through, inspection and probably manipulation of variables. Note that this debugger will connect to a remote SOA Suite and OSB run time environment to perform remote debugging.
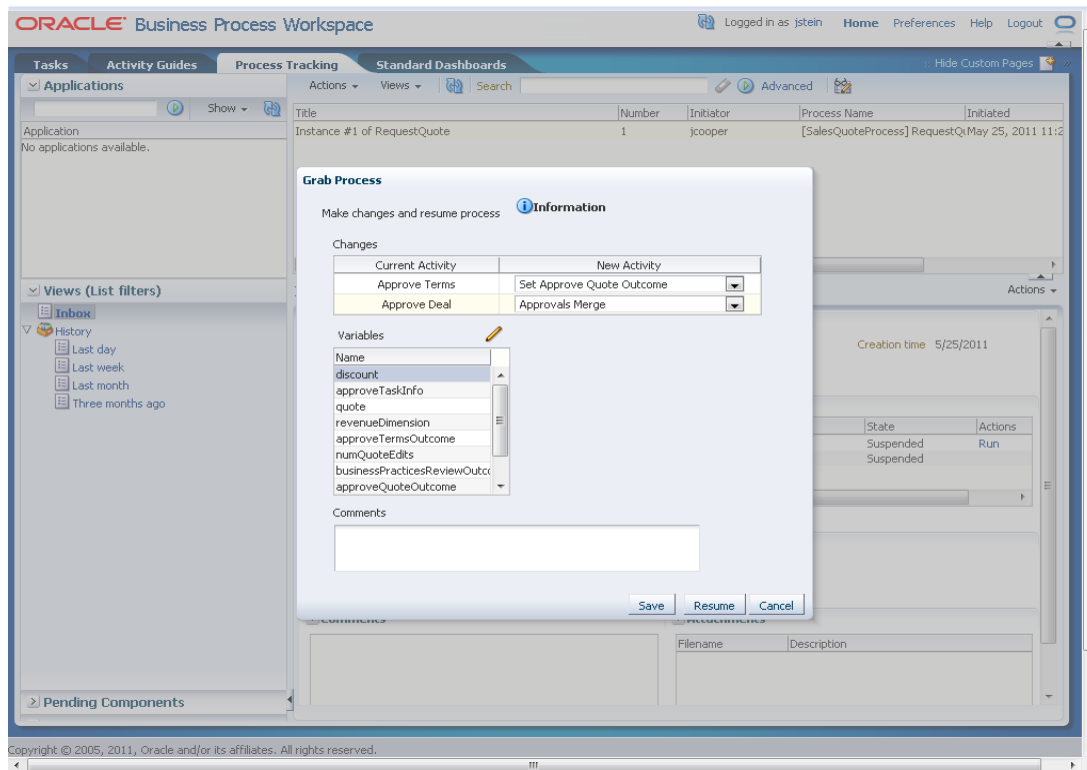
Another step in making the SOA development process more like regular application development is support for refactoring with regard to SOA Suite applications. That includes for example name changes of components and operations - currently quite hard to accomplish.

It is not like nothing is happening in the short term. Before the year is out, we are expecting Patch Set 5 (11gR1 PS5) that will introduce several interesting bits and pieces. This includes better purging support to remove instance data, BPEL component level testing, a UMS adapter that provides support for email interaction and allows instances to be triggered by the reception of an email, support for the Aggregation Pattern in BPEL, better testing facilities for Business Rules and support for CEP (Complex Event Processing) on WebLogic.

BPM

An area that is tremendously important to Oracle is BPM or Business Process Management. A perfect example of business meets IT, BPM is an architecture approach supported by the BPMN notation and technology like the Oracle BPM 11g Suite, that puts strong emphasis on business processes in looking at, analyzing, organizing and improving organizations. In terms of IT systems and applications, BPM drives a move away from departmental, data-oriented applications to cross-department, cross-system process oriented systems where users and automated services interact together working on the execution of business processes.

Prior to Oracle Open World, the so called Feature Pack release for BPM was published, containing a wealth of new functionality. Among the new features is the much sought after option to edit a running process instance - for example to resolve issues with stuck processes, improvise in exceptional circumstances and manipulate process state when the predefined process does not cater for the current situation.

ORACLE Business Process Workspace

Logged in as jstein   Home   Preferences   Help   Logout

Tasks   Activity Guides   Process Tracking   Standard Dashboards   Hide Custom Pages

Applications
Actions ▾   Views ▾   Search   Advanced

| Title | Number | Initiator | Process Name | Initiated |
| Instance #1 of RequestQuote | 1 | jcooper | [SalesQuoteProcess] RequestQt | May 25, 2011 11:2 |

Application
No applications available.

Grab Process

Make changes and resume process   ⓘ Information

Changes

Current Activity	New Activity
Approve Terms	Set Approve Quote Outcome ▾
Approve Deal	Approvals Merge ▾

Views (List filters)
Actions ▾

Inbox

History
Last day
Last week
Last month
Three months ago

Variables

Name
discount
approveTaskInfo
quote
revenueDimension
approveTermsOutcome
numQuoteEdits
businessPracticesReviewOutc
approveQuoteOutcome

Creation time   5/25/2011

	State	Actions
	Suspended	Run
	Suspended	

Comments

Save   Resume   Cancel

Filename   Description

Pending Components

Another important facility is the ability to update running process instances with a new definition of the process - to upgrade long running processes to the latest process design.

BPM 11g PS4 with FP4 also supports correlation of events and messages into running instances, easier manipulation and association of documents from UCM and better collaboration and design time at run time support in the BPM Process Composer.

A huge ambition for the 12c release of BPM is support for Case Management - a much more flexible approach to workflows where professional players have large freedom to define how a process instance - a case - is handled. Other objectives for 12c include 'BPM on the cloud' and the Process Accelerators. These accelerators are pre-built business processes that many organizations will be able to take on, customize and roll out. Typically the process accelerator provides generic support for supporting processes such as leave or vacation requests, event coordination, travel request, expense reporting, sickness notification etc.
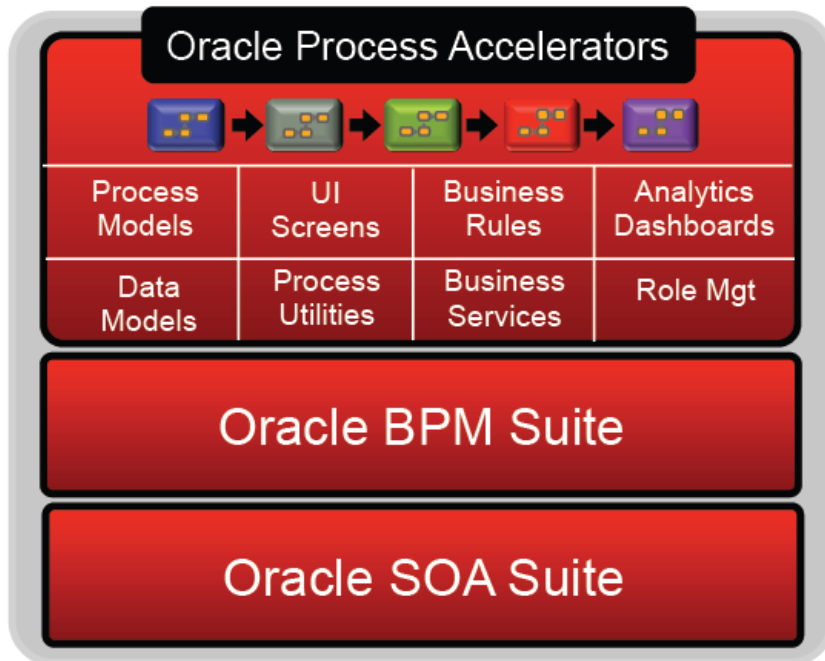
The process accelerators serve as demonstrations of what BPM Suite together with SOA Suite and supporting components can do and should also have real business value for organizations - either as is out of the box or after customization.

Oracle is looking to extend BPM Composer with areas previously found in BPA - regarding enterprise architecture. BPM Composer will be enhanced and extended to support a broader user community of business analysts, process owners, enterprise architects and other parties involved with defining, re-designing, implementing and executing processes. Additionally, round trip process metrics becomes available: simulation of process models can be based on actual metrics collected during the execution of business processes.
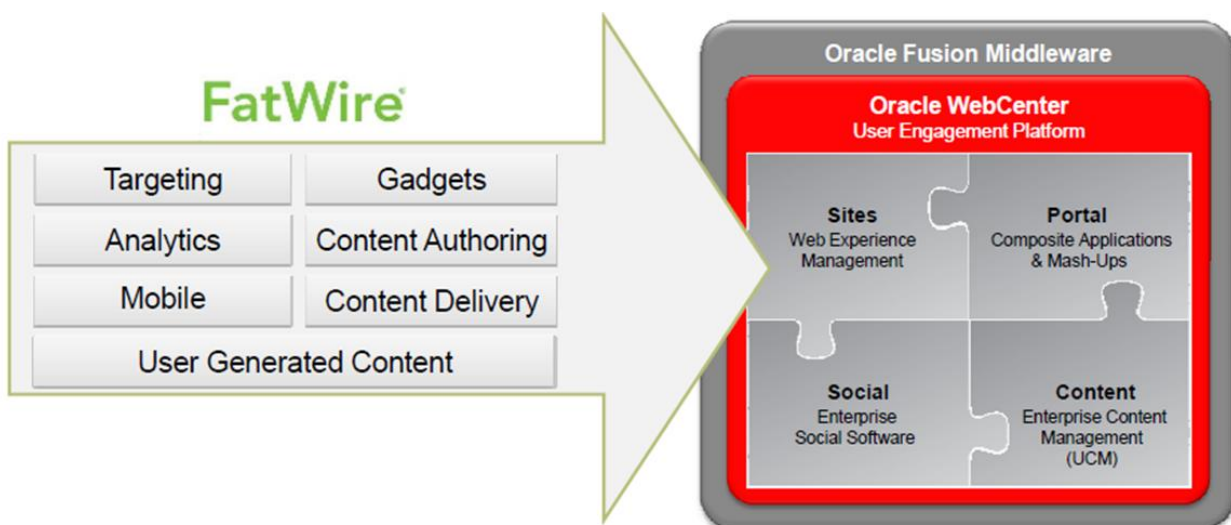
WebCenter

Oracle has been struggling with WebCenter. On the one hand, it is great product with interesting features and a great story to tell. On the other, it is a combination of things - some targeted at end users and business representatives and others at developers and architects. It has no clearly identified target audience and Oracle staff have found it difficult to find the right sales pitch. WebCenter has been around for four years and has definitely attracted a following - but things could have been better, given all the expectations, opportunities and investments.

Lately - Summer of 2011 - Oracle has somewhat restructured WebCenter. That goes for both the internal product management organization, as well as the product offering. FatWire was acquired - now to be offered as WebCenter Sites - to provide a solution for externally facing web sites. Oracle has recognized that there is a distinction between an enterprise portal with internal or at least know, logged

in users with clear needs for collaboration and interaction with [information from] enterprise systems and involvement in business processes and an external web site with more casual, typically not logged in, browsing and buying interaction. The previous WebCenter offering did an excellent job for the enterprise portal, but fell somewhat short for the external web site experience.

The FatWire product - or WebCenter Sites as it is now labeled - provides WebExperience Management. It allows marketing personnel to organize content, targeted at specific categories and interests. WebCenter Sites tracks user interests - searches, navigation - and adapts the content offered to the personal interests of users. It provides the site masters with detailed insight in navigation and search patterns. It caters for content contributed by visitors - in the form of comments, ratings, blogs and tags. It can render to a plethora of different devices - with content optimized per device.

Using the gadget mechanism, extensions can be created - with content from WC Sites - that can be embedded in Facebook or LinkedIn pages or mashed up in iGoogle. A similar mechanism is planned to allow for mutual integration between WebCenter Sites and Portal. This would allow interaction with enterprise systems in WebCenter Sites and usage of Sites' gadgets in the enterprise portal. Note that a lot of work needs to be done within Oracle to make this interaction a reality.

After the restructuring, WebCenter is defined a four different products: Sites, Portal, Social and Content. These work together, but can be used by themselves too.

WebCenter is the foundation for Oracle Social Network. This is the cloud service offered by Oracle that provides collaboration and communication for the enterprise, similar to a combination of Twitter, Facebook, LinkedIn and Yammer - targeted at enterprises and their communities of professionals.
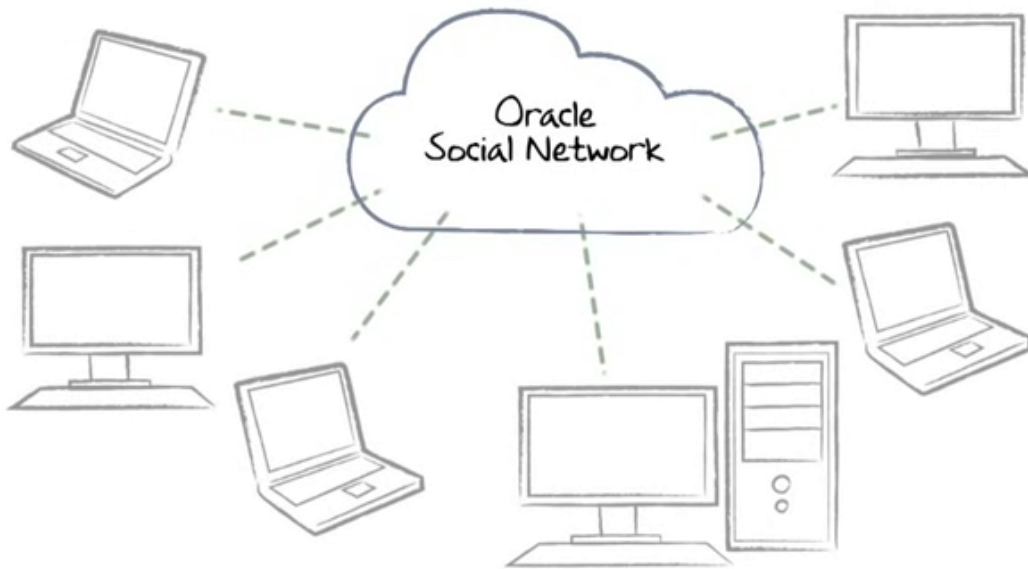
Some elements from Beehive - Oracle's previous attempt at collaboration software that seems to have left the stage - for communication and web conferencing and many social services from WebCenter provide the core for Oracle Social Network.

NoSQL

The very last announcement expected to come out of Oracle one might say is "No SQL". Yet here we were at Oracle Open World 2011, with Oracle proudly announcing its NoSQL database.


NoSQL, as Oracle staff quickly will explain, does not mean literally "no sql" but stands for Not Only SQL. And this is meant to say that the rich functionality, referential integrity and high reliability of relational databases may not always be required for every form of data storage. Not all data produced and processed in some way by enterprises needs to be stored in a relational Oracle database. For example because it is data in some intermediate state that should be further refined, enriched and processed before it makes sense to persist it. Or data that has been retrieved from a relational database to be used to present in a user interface and therefore has been cached in memory structures.

A NoSQL database has been around at Oracle for many years, every since Oracle acquired SleepyCat back in 2006. The Berkeley DB product happens to be a no-sql (or non-relational) database that is probably the most widely shipped database in the world (part of every Linux distribution). It is used to store potentially huge volumes of data as key-value pairs. No referential integrity or data constraints are

enforced by the database. The application is responsible for the data it stores. Data can be stored by key and very rapidly be retrieved by key. Data can be structured and nested. The Berkeley DB database has a C++, Java, PHP, Perl and an XQuery API and now also a SQL API. It now forms the core of the Oracle NOSQL Database (see http://dbmsmusings.blogspot.com/2011/10/overview-of-oracle-nosql-database.html for many interesting details), that is positioned as the container for the data produced by the 'digital exhaust' - the multitude of systems and channels, sensors and monitors that continuously produce measurements, tracking data, click analysis and other forms of digital notifications. This data may render valuable insights and analysis outcomes, after having been processed. These valuable results will end up in the relational database.

To efficiently process the data in the NoSQL database, organizations can leverage Hadoop (see http://en.wikipedia.org/wiki/Hadoop). Hadoop is a protocol - inspired by the MapReduce algorithm - for executing jobs in a parallel fashion using a cluster of nodes that can be as simple as laptops or PCs. Or, somewhat more formal: "Hadoop is a generic processing framework designed to execute queries and other batch read operations against massive datasets that can be tens or hundreds of terabytes and even petabytes in size. The data is loaded into or appended to the Hadoop Distributed File System (HDFS). Hadoop then performs brute force scans through the data to produce results that are output into other files. "

Oracle provides tools to transfer data from the NoSQL database to an Hadoop network that will process the data: the Oracle Data Integrator Application Adapter for Hadoop. On the outbound end of Hadoop, Oracle Loader for Hadoop, transforms datasets created by MapReduce-crunching into data block formats native to Oracle 11gR2 databases so they can be directly loaded.
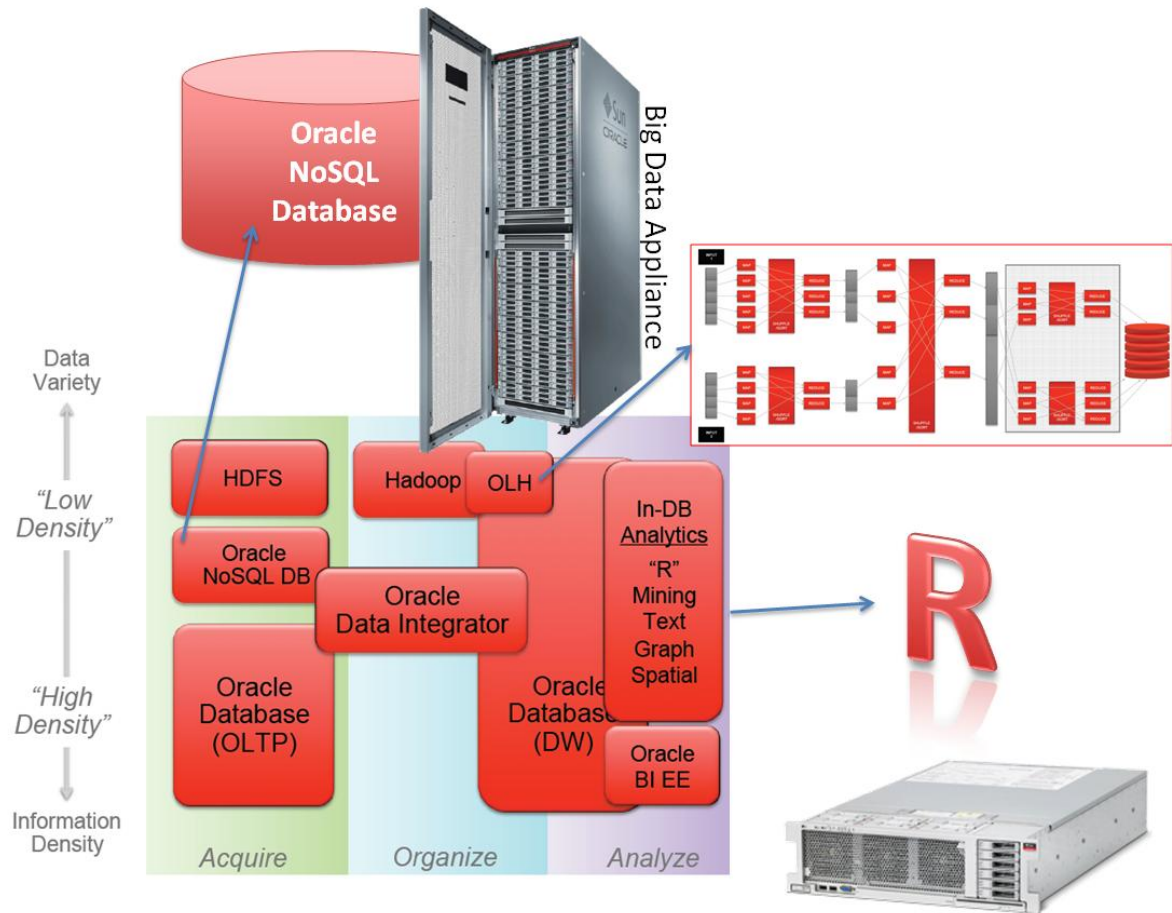
Oracle also announced the Big Data Appliance at OOW 2011, which integrates Hadoop, Oracle Enterprise Linux, the R programming language, and a NoSQL database with the Exadata hardware

In addition, there is the Exalytics machine, big on memory, with 1TB of data in main memory that can seem like 5 to 10TB thanks to some special compression techniques. The data in memory is actually the TimesTen relational OLTP and Essbase multidimensional OLAP databases, accessed through parallelized analytics algorithms across the 40 cores and 80 threads at about 200GB/sec. The machine can be connected to any Oracle database source with a Gigabit or 10 Gigabit Ethernet link, or to an Exadata through a 40Gb/sec InfiniBand link.

Note that the main idea for Analytics is that if the (intermediate results that lead to) answers to questions asked by users in analytical queries are preprocessed and preloaded in memory, those answers can be provided much faster.

The big R in the illustration above represents the open source statistical computing language called R (see http://www.r-project.org/). R provides a wide variety of statistical (linear and nonlinear modelling, classical statistical tests, time-series analysis, classification, clustering, …). R has been supported with Oracle Data Mining in the 11gR2 database since 2010. It will now be extended with an R API that R Applications can use to 'tap into data in the database'.

Forms 11gR2

And what about Forms 11g R2 you may wonder. That most important announcement of the entire conference. Well, the importance of this release is primarily in the fact that it is there. That Forms is still on the move. The most important aspect of this release is the integration - at last - with Oracle Access Manager which allows us to do authentication and authorization of Oracle Forms applications through OAM. This release also brings integration with Real User Experience Interaction- a component of Enterprise Manager for monitoring and tracking the actual experience for end users in terms of performance, availability and navigation. Forms 11gR2 is supported on Windows 64bit. The PL/SQL editor now supports line numbers when running on the Windows. Windows can now be fully maximized and iconic buttons and Read_Image_File can have images defined by dynamic URLs. This release also introduces a data type NCHAR allowing improved handling of Unicode character sets. Finally, this release allows the Forms menu bar and window decoration to be turned of, which allows Forms to be more seamlessly integrated into other technology pages such as HTML pages.

Again, no earth shattering new features, but certainly another step in the evolution of Forms, a platform that is still valid for some very distinct use cases.

Conclusions

Oracle Open World 2011 was - like previous edition of this huge gathering of Oracle aficionados - a whirlwind tour of announcements, demonstrations, unspoken statements and whispers. From sweeping declarations to subtle messages between the lines. It is a real challenge to sift through all the noise to extract the essential bits and pieces - given your focus areas and personal objectives.

Oracle is alive and on the move - nothing new there. Oracle is embracing trends - first of all cloud and no sql - and pursuing its own goals with appliances and Exa-machines. Fusion Applications is now generally available - and, very importantly from the point of view of Fusion Middleware: it provides a wonderful sources of inspiration, best practices and references for the use of Fusion Middleware products.

The Fusion Middleware stack itself was not subject of major announcements. We are experiencing a rapid evolution with BPM - huge steps that make it more usable. Most other components - ADF, WebCenter, OSB, SOA Suite, UCM - seem relatively stable, working to their next climax with the 12c release, apparently slated for the end of 2012. Each of these products is still seeing valuable progress and of course especially the ADF Mobile announcement is pretty substantial.

All in all it seems safe to state that Fusion Middleware has reached a level of maturity that will make it attractive to many organizations that may have been waiting for it to achieve that state.

Additional Resources

There are several excellent on line resources available to learn more about the Oracle Open World 2011 and JavaOne 2011 conferences. Take for example a look at these resources:

[Content Catalog with slides from most Oracle Open World presentations](https://oracleus.wingateweb.com/scheduler/eventcatalog/eventCatalog.do):
(https://oracleus.wingateweb.com/scheduler/eventcatalog/eventCatalog.do)

[Video streams for the most relevant OOW keynotes](http://medianetwork.oracle.com/media/channels/2011-openworld/948003235001)
(http://medianetwork.oracle.com/media/channels/2011-openworld/948003235001)

[Content Catalog with the slides for most JavaOne presentations]

[Video streams for JavaOne keynotes](http://medianetwork.oracle.com/media/channels/2011-JavaOne/374671059001) (http://medianetwork.oracle.com/media/channels/2011-JavaOne/374671059001)

[Parlays video-streams for dozens of JavaOne technical sessions](http://www.parleys.com/#st=4&id=102979):
http://www.parleys.com/#st=4&id=102979