LEVERAGING ORACLE DRM TO INTEGRATE ORACLE EBS CUSTOMER DATA WITH SALESFORCE

Cervello

INTRODUCTION

Hierarchy management for both reporting and operational environments can create technical, business and architectural challenges in any organization. Most organizations struggle with mastering customer data along with combining it with the business views and hierarchies required to effectively manage the business. Typically a group will initiate a siloed solution that lacks both the proper governance process and extensibility to be leveraged beyond the department.

This whitepaper explores how a leading medical supply and software manufacturer leveraged Oracle Data Relationship Manager (DRM) to integrate Oracle EBS customer data with Salesforce. To improve the sales team effectiveness and customer visibility, there was a need to create Customer and Regional hierarchies for bill-to and ship-to sites in Salesforce. The sales operations team needed to establish business logic and validation rules in Oracle DRM as customer master data flows from Oracle EBS to Oracle DRM before being loaded to Salesforce. This presentation highlights the business and technical challenges the team faced and how Oracle DRM created a robust solution to satisfy both the IT and business functions.

ABOUT THE CLIENT

The medical supply and software manufacturer provides devices, software and services. Their client-base spans globally ranging from collection centers to hospitals along with a broad set of product lines and solutions. Business environment changes such as rapidly improving technology, emerging markets and acquisition activities have created challenges to continue to sustain and grow revenue. This environment has created pressure on the sales organization to sustain the current customer base and expand into new customer areas. To effectively achieve this it is critical for sales teams to have a single view into the customer and properly manage selling activities of products and services.

From a technology and platform perspective the sales team leverages the Salesforce Sales Cloud for customer relationship management. This includes tracking of individual customer sites where product is billed and ultimately shipped. Activities include opportunity management; contact tracking and historical invoice

tracking. The enterprise manages three separate Salesforce organizations (instances) for North America, Europe and Asia-Pac. Each organization integrates with a single enterprise instance of Oracle E-Business Suite (EBS). Integration between Oracle EBS and Salesforce is handled utilizing Web Methods, which has a standard connector with Salesforce. In addition, Oracle Business Intelligence Applications (OBIA) is utilized for pre-built reporting solutions on top of Oracle EBS.

PRIMARY BUSINESS CHALLENGES

As stated, there is a set of business drivers that warranted a need to get a better view into a Customer. Previously, individual bill-to and ship-to sites for a customer were not organized within Salesforce. For example, a major collection customer will have hundreds of sites across the United States alone. From a Salesforce perspective, each of these sites are represented as Accounts. The challenge is that without creating a hierarchy of these Accounts there is no view into a Customer at an aggregate level. In addition, there is a need to create regional structures to group sites within a Customer.

Salesforce.com does provide a concept of creating a hierarchy for Accounts. A single Account can have a Parent Account relationship, which will create a hierarchy with as many levels as needed. Unfortunately, there are some inherent gaps with the standard implementation.

The first limitation is that there is no natural method to view related objects rolled up in the Customer hierarchy.   In Salesforce a related object is an Opportunity or Contact record that is associated with a specific Account. If a user wants to see all Opportunities for a Customer that has multiple Sites at an aggregate level, it is not possible in a standard implementation. It requires customization leveraging the Force.com, which is a development platform used to extend Salesforce.

The second limitation is related to easily viewing the full Customer hierarchy structure. Customer structures, although searchable, are not easily navigable or user-friendly to view. As an administrator or an end-user, it can create challenges to fully understand a Customer, Region and the related Site structures.

The third limitation is related to the governance associated with maintaining the Customer hierarchies from an administrative perspective. Salesforce as a CRM is considered an operational system and therefore is not inherently strong with tracking and maintaining history for hierarchy changes. The ability to create scenarios and provide version control for maintaining the Customer hierarchies is not the purpose of Salesforce. It is also not ideal for being a centralized hub for maintaining Customer hierarchies that can be leveraged in other dependent systems, such as a data warehouse or reporting platform.

Lastly, security and visibility is both a concern and a management nightmare. Rolling up Account Team access to individuals both up and down a Customer hierarchy is needed to provide the required access to Accounts and related Salesforce objects.

To address these limitations, the Salesforce implementation was extended to build out the Customer hierarchies and provide the single view into a Customer within Salesforce. Although the implementation was successful, there was still a gap with providing a centralized method to maintain the Customer hierarchies and integrate with Oracle EBS. There was also a set of business rules that needed to be established to automate new Customers and Sites being added in Oracle EBS without manual intervention from an administrator. In addition, there is a need to override a standard Customer hierarchy to group multiple Customers into a single hierarchy. This scenario occurs for customers that have multiple hospitals or military branches, which are technically individual Customers in EBS but from a management perspective need to be grouped into a single Customer in Salesforce.

Based on these post-Salesforce implementation needs, the decision was made to leverage Oracle Data Relationship Management (DRM) to manage the Customer hierarchies and become the integration hub between Oracle EBS and Salesforce.com.
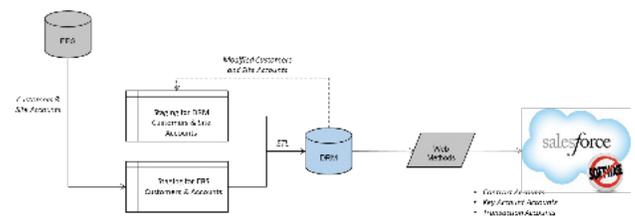
SOLUTION

Multiple technologies were used in order to extract Oracle EBS information and load the appropriate structure into Saleforce.com. At the start of the process there is custom extract, transform and load (ETL) process developed in PL/SQL that is responsible for extracting EBS data and applying the appropriate logic to transform the data before loading it into DRM.

As stated, the central aspect of the solution is Oracle's DRM tool which is responsible for maintaining the Account mapping structures for a Customer between Oracle EBS and Salesforce. Oracle DRM provides a portal for functional users to directly update the hierarchies that are used to define the Account structures in Salesforce.

Finally, to feed the Customer hierarchies into Salesforce, Web Methods is used to extract from a staging table and load delta changes into Salesforce.

The architecture and design of the final solution provides the following key functional aspects:

- Provide a single point for managing Customer Account structures and definitions
- Automated data feeds and extracts to update Salesforce.com
- Delivers audit functionality and version control for Customer hierarchy changes
- Provide appropriate end user access to Oracle DRM based on business unit and functional role
- Capability to create "point-in-time" snapshots of Customer hierarchies

A key design element to the end-to-end solution was handling Account identifiers in Salesforce. Since a key design requirement was to minimize the amount of transformation and logic when loading data to Salesforce, we couldn't rely on the system identifiers managed in Salesforce. As a result, we established custom identifier fields on the Account object in Salesforce that would be dependent on Oracle DRM to manage. Three new custom Account fields were added in Salesforce that would be managed and updated from Oracle DRM. These fields established the hierarchical relationships between a Parent and Child Account record to build the Customer hierarchy. This also decoupled the solution from Salesforce identifiers, which are dictated by Salesforce and cannot be changed.

The implementation of the Oracle DRM solution involved a few key functional aspects in the design. This included handling all use cases of a Customer hierarchy definition; validation and verification rules for Customer hierarchies and nodes; defined queries utilized by end users to find hierarchy changes and issues; automated creation of action scripts to update DRM; and export routines to integrate with staging tables and downstream dependent applications.

Another key design element was managing the delta changes from Oracle EBS and DRM before updating Salesforce. This involved three layers of comparison to identify changes and manage the deltas before updating Salesforce.

- Oracle EBS to DRM: handle net new Customers and Sites by comparing to existing DRM definitions
- Oracle EBS Master to EBS Delta: handle if a change to an existing Customer or Site has an impact on DRM and if the change should flow to Salesforce
- Oracle DRM Current to Oracle DRM Previous: handles comparisons in DRM by hierarchy level (Customer, Region and Site) to determine insert, update and deletion of nodes along with property changes

Finally, the data flow process and production process timing was a critical aspect to the solution. The current state prior to the DRM implementation was that new Sites created in Oracle EBS were loaded every hour to Salesforce so that invoices could also be loaded to Salesforce. If a Site Account weren't present the invoice load to Salesforce would fail. The challenge with this was that we now had a dependency on also creating the parent nodes (Customer and Region) as Accounts in Salesforce if they did not exist. To address this requirement and still provide hourly updates to Salesforce, the solution design was able to auto-create default Customer and Region accounts in DRM without administrator intervention if it was a new client. The design also handled attaching a new Site to a Region parent Account based on zip code ranges defined in DRM. This design guaranteed that orphan Site accounts will not exist and data can continue to flow to Salesforce.

LESSONS LEARNED

The final architecture and design and subsequent implementation provided a set of key lessons learned when integrating Oracle EBS, Oracle DRM and Salesforce.

The first key lesson learned is to identify all the use cases that need to be handled within Oracle DRM. Although close to 90% of the Customer hierarchies were considered of similar structure, the remaining cases drove the majority of the complexity of the integration. The majority of this complexity was as to how to allow the DRM administrator to be able to override default Customers and combine into a single "super" Customer structure.

The second key lesson learned is to minimize the amount of changes loaded to Salesforce as much as possible. One challenge when developing on the Force.com and Salesforce platform is managing to the governor limits of the multi-tenant environment. To handle this the Oracle DRM solution and implementation needed to only provide delta changes to be loaded to Salesforce. This was critical when considering the hourly loads to Salesforce. This also included minimizing the amount of transformation required as part of the Extract, Transform and Load process via Web Methods.

The third lesson learned was appropriately dealing with invalid character sets based on limitations in DRM. This is applicable for DRM node name and limitations may not be present in other source/target systems (so in this case, EBS is the source and did not have such limitations and neither did SFDC). Further, downstream systems that are fed from DRM also were impacted. This is a classic master data management issue.

Finally, testing end to end was a critical process. With the amount of integration required and number of moving parts it was critical that each aspect of the architecture was testing. This required a considerable amount of coordination across different groups and extensive documented test cases. This included the 10% unique use cases, which created the majority of the testing effort.

NEXT STEPS

After the initial implementation, there are a series of identified enhancements and extensions that are being implemented.

The first is extending the solution to the Europe instance of Salesforce. As mentioned earlier, there are multiple instances of Salesforce globally. Once North America was successfully deployed, both the Salesforce Customer hierarchy solution and Oracle DRM integration are being extended to the Europe Salesforce user base. Since there is a single instance of both Oracle EBS and DRM, the integration with another instance of Salesforce is primarily focused on extending both Web Methods and Salesforce.com.

The second extension is to provide the Salesforce and DRM administrator a method to do mass updates to the customer hierarchies. This is a fairly common activity when a Customer requires a major restructure of Regions and movement of sites. A Salesforce administrator will make mass updates using the Apex Data Loader, which is a tool that can use a comma-separated file to insert, update and delete records in Salesforce. To provide a similar experience with Oracle DRM, we are building an Excel-based tool using the DRM application-programming interface (API) that will be leveraged to make mass Customer hierarchy restructures.

In addition, there is currently a Sales Reporting initiative in Oracle Business Intelligence that is going to leverage the Customer hierarchies established in Oracle DRM. Output feeds will be created from DRM and integrated into the data warehouse environment and consumed within the Oracle Business Intelligence reporting solution.

Finally, the Oracle DRM platform is being considered to manage additional Salesforce related subject areas including product related data. This includes product codes, descriptions, product families, revenue and quantity schedules and product pricing.

CONCLUSION

Oracle Data Relationship Management is a powerful platform to manage hierarchy data. By combining this capability and centralized hub model with the Oracle E Business Suite customer master data you can create a best of breed solution. Once the hub architecture utilizing Oracle DRM has been established, integration with platforms such as Salesforce can be made. This approach avoids departmental and siloed interpretations of specific subject areas such as a single view of a customer. At the same time it provides the flexibility to create new methods via business specific hierarchies that can be leveraged at the enterprise level.

ABOUT CERVELLO

Cervello Inc., is a leading professional services and solutions provider focused on helping companies solve complex data challenges, improve business analytics and optimize business performance. We focus on transformative cloud-based technologies in enterprise performance management, data management and business intelligence and customer relationship management. Cervello works with some of the leading on-premise and cloud software providers such as Oracle, Host Analytics, Salesforce.com and Birst. Our core services include system implementation, advisory services, custom application development and managed services.

CONTACT US

For more information, visit us at www.mycervello.com or contact us at info@mycervello.com.