

Oracle® Leads Management

Implementation and Administration Guide

Release 11*i*

Part No. B13534-01

June 2004

Oracle Leads Management Implementation and Administration Guide, Release 11i

Part No. B13534-01

Copyright © 2003, 2004, Oracle. All rights reserved.

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

If the Programs are delivered to the United States Government or anyone licensing or using the Programs on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software--Restricted Rights (June 1987). Oracle Corporation, 500 Oracle Parkway, Redwood City, CA 94065.

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and we disclaim liability for any damages caused by such use of the Programs.

The Programs may provide links to Web sites and access to content, products, and services from third parties. Oracle is not responsible for the availability of, or any content provided on, third-party Web sites. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Oracle is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Oracle is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

Contents

Send Us Your Comments	xi
Preface.....	xiii
How To Use This Guide	xiii
Documentation Accessibility	xiv
Other Information Sources	xiv
Training and Support.....	xix
Do Not Use Database Tools to Modify Oracle Applications Data	xx
About Oracle	xx
1 Introduction to Oracle Leads Management	
1.1 Oracle Leads Management Overview	1-2
1.1.1 Leads in the Marketing and Sales Cycles	1-2
1.1.2 Oracle Leads Management Features	1-3
1.2 New Features and Enhancements in this Release	1-3
1.2.1 New Features	1-3
1.2.1.1 Rule-based Deduplication	1-4
1.2.1.2 Linking Interactions to Lead Flow	1-4
1.2.2 Enhancements.....	1-4
1.2.2.1 Disqualifying Leads.....	1-4
1.2.2.2 Leads Rules Engine Usability.....	1-5
1.2.2.3 Leads Rules Performance Report	1-5
1.2.2.4 Supporting DUNS Number.....	1-5
1.2.2.5 Enhancing Performance	1-5
1.3 Oracle Leads Management Dependencies.....	1-6
1.4 Upgrading from Release 11.5.9	1-6
1.4.1 Single Product Catalog.....	1-6
1.4.2 Monitoring Engine	1-6

2 Business Flows

2.1	Business Flows in Oracle Leads Management	2-2
2.1.1	Capturing and Cleaning Leads	2-2
2.1.2	Processing Leads	2-4
2.2	Scenarios Using the Leads Processing Engine	2-6
2.2.1	Scenario For Qualifying Leads	2-6
2.2.1.1	Using the Qualification Engine to Qualify Leads.....	2-6
2.2.1.2	Creating the Qualification Rule Set	2-7
2.2.1.3	Using the Rating Engine.....	2-9
2.2.1.4	Creating the Rating Rule Set.....	2-10
2.2.1.5	Using the Channel Selection Engine	2-12
2.2.1.6	Creating the Channel Selection Rule Set.....	2-12
2.2.2	Scenario For Disqualifying Leads	2-14
2.2.2.1	Using the Qualification Engine to Disqualify Leads.....	2-14
2.2.2.2	Creating the Qualification Rule Set	2-15
2.2.2.3	Lead Rating Rule Set.....	2-17
2.2.2.4	Creating the Rating Rule Set.....	2-18
2.2.2.5	Lead Channel Selection Rule Set.....	2-20
2.2.2.6	Creating the Channel Selection Rule Set.....	2-21

3 Capturing and Cleaning Leads

3.1	Leads from Multiple Sources	3-2
3.2	The Lead Import Process	3-3
3.3	Importing Leads.....	3-4
3.3.1	Importing Leads from the HTML Interface.....	3-4
3.3.2	Evaluating Errors During Lead Import.....	3-6
3.3.3	Correcting Errors During Lead Import.....	3-7
3.3.4	Importing Leads From a Flat File.....	3-7
3.3.4.1	Importing Leads by Running the Concurrent Program.....	3-8
3.3.4.2	Importing Leads by Running SQL Loader Manually	3-9
3.3.4.3	Sample Flat File	3-10
3.3.4.4	Flexfields.....	3-10
3.3.5	Imported Leads and Oracle Sales Tables	3-11
3.4	The Import Sales Lead Concurrent Program.....	3-13
3.4.1	Import Sales Lead Concurrent Program Flow	3-13

3.4.2	Before Running the Concurrent Program.....	3-15
3.4.2.1	Setting Up DQM Match Rules.....	3-15
3.4.2.2	Setting Profiles Used by the Import Sales Lead Concurrent Program.....	3-18
3.4.2.3	Creating Valid Lookup Codes.....	3-19
3.4.2.4	Setting up Territories.....	3-20
3.4.2.5	Loading the Import Interface Tables.....	3-20
3.4.3	Running the Import Sales Lead Concurrent Program.....	3-21
3.4.3.1	Import Sales Lead Concurrent Program Parameters.....	3-21
3.4.4	Enhancing the Performance of the Import Sales Lead Concurrent Program.....	3-22
3.4.5	Limitation of the Import Sales Lead Concurrent Program	3-23
3.5	Data Quality	3-24
3.5.1	Customer Data Quality	3-25
3.5.1.1	Checking for Duplicate Original System Reference	3-25
3.5.1.2	Checking for Duplicate Customers	3-27
3.5.1.3	Checking for Duplicate Addresses.....	3-27
3.5.1.4	Checking for Contacts and Contact Points.....	3-28
3.5.2	Setting Up DQM Staging Schema.....	3-30
3.5.3	Designing Matching Rules to Detect Duplicate Customer or Person	3-30
3.5.3.1	Identify Duplicate Parties	3-31
3.5.3.2	Identify Duplicate Persons	3-35
3.5.3.3	Identify Duplicate Contacts.....	3-39
3.5.4	Leads Data Quality	3-43
3.5.4.1	The Deduplication Rule	3-44
3.5.4.2	Customizing the Deduplication Rule.....	3-45
3.5.4.3	Custom Deduplication Using User Hooks.....	3-47
3.6	Custom Codes with the Lead Import Program	3-50
3.6.1	Seeded Business Events.....	3-50
3.6.1.1	Error Handling	3-51
3.6.1.2	Execution Control	3-51
3.6.1.3	Sample Function Codes for the Business Event Subscriptions	3-52
3.6.2	Custom User Hook	3-54
3.6.2.1	Seeded Code for the User Hook	3-55
3.7	Purging Staged Lead Records	3-56

4 Linking Interactions to Leads

4.1	Interaction Matching Engine Overview	4-2
4.1.1	Interaction Types and Interaction Scores.....	4-2
4.1.2	Interaction Score Threshold	4-3
4.2	Functioning of the Interaction Matching Engine	4-4
4.2.1	The Mining Behavior of the Interaction Matching Engine.....	4-4
4.2.2	Linking an Interaction with Existing Leads	4-5
4.2.3	Linking if No Lead Exists for the Interaction.....	4-6
4.2.4	Running the Concurrent Program	4-7
4.3	Setting Up Interaction Types	4-7
4.4	Interaction Rule Sets.....	4-8
4.4.1	Creating an Interaction Rule Set	4-8

5 Processing Leads

5.1	Leads Processing Engine Overview.....	5-2
5.2	The Qualification Engine.....	5-4
5.2.1	How Does the Qualification Engine Work?	5-5
5.3	The Rating Engine	5-5
5.3.1	How Does the Rating Engine Work?.....	5-6
5.3.2	Setting up Ratings	5-6
5.4	The Channel Selection Engine	5-7
5.4.1	How Does the Channel Selection Engine Work?.....	5-7
5.4.2	Setting Up Channels	5-8
5.5	Best Practices	5-9
5.6	Working with Rule Sets	5-9
5.6.1	Creating a Rule Set.....	5-10
5.6.2	Searching for a Rule Set.....	5-12
5.7	The Rule Flows Report.....	5-12
5.8	Purging Unqualified Leads	5-14
5.9	Setting Up Lead Assignments	5-14
5.9.1	Setting Up Automatic Lead Assignment	5-15
5.9.2	Setting Up Immature Lead Assignment	5-16
5.9.3	Routing Leads Using a User Hook	5-18
5.10	Setting Up Lead Status.....	5-22
5.11	Using Custom Attributes.....	5-23

5.12	Setting Up Time Frames	5-24
5.12.1	Customizing Time Frames.....	5-24

6 Monitoring Leads

6.1	Monitoring Engine Overview.....	6-2
6.1.1	Scope of a Monitor	6-3
6.1.1.1	Upgrading from Release 11.5.9	6-4
6.1.2	Smart Time Frames and Expiration Dates.....	6-4
6.1.3	Timeouts in the Monitoring Engine	6-5
6.2	Running the Workflow Background Process Concurrent Program	6-6
6.3	Creating a Monitor Rule.....	6-7
6.3.1	Monitoring Engine Conditions	6-9
6.4	Viewing the Monitor Log.....	6-10
6.5	Searching for a Monitor	6-10

7 Operational Reports

7.1	Imports Records Manager Reports.....	7-2
7.2	Lead Processing History Reports.....	7-2
7.3	Rule Performance Reports	7-4
7.4	Rule Diagnostics Reports	7-5
7.5	Lead Linking Report.....	7-6

A System Profile Options

A.1	Setting System Profile Options.....	A-2
A.2	System Profile Options in Oracle Leads Management	A-3

B Seeded Data

B.1	Seeded Attributes	B-2
B.1.1	Seeded Lead Attributes	B-2
B.1.2	Seeded Interaction Matching Engine Attributes	B-6
B.2	Seeded Lookups.....	B-6
B.2.1	Time Frame	B-7
B.2.2	Lead Rank.....	B-7
B.2.3	Lead Status	B-7

C Concurrent Programs

C.1	Running Concurrent Programs	C-2
C.2	Concurrent Programs in Oracle Leads Management.....	C-3

D Oracle Leads Management API Reference

D.1	Oracle Leads Management Procedures.....	D-2
D.2	Parameter Specifications.....	D-3
D.2.1	Standard IN Parameters	D-3
D.2.2	Standard OUT Parameters	D-5
D.3	Type Declarations.....	D-6
D.4	Create Sales Lead.....	D-11
D.4.1	Procedure Specification	D-11
D.4.2	Parameter Descriptions	D-12
D.5	Update Sales Lead	D-13
D.5.1	Procedure Specification	D-13
D.5.2	Parameter Descriptions	D-14
D.6	Update Sales Lead Lines.....	D-14
D.6.1	Procedure Specification	D-14
D.6.2	Parameter Descriptions	D-15
D.7	Delete Sales Lead Lines.....	D-16
D.7.1	Procedure Specification	D-16
D.7.2	Parameter Descriptions	D-16
D.8	Lead Process After Create	D-17
D.8.1	Procedure Specification	D-17
D.8.2	Parameter Descriptions	D-18
D.9	Lead Process After Update.....	D-18
D.9.1	Procedure Specification	D-19
D.9.2	Parameter Descriptions	D-19
D.10	Run Lead Engines.....	D-20
D.10.1	Procedure Specification	D-20
D.10.2	Parameter Descriptions	D-20
D.11	Build Lead Sales Team	D-21
D.11.1	Procedure Specification	D-21
D.11.2	Parameter Descriptions	D-22
D.12	Rebuild Lead Sales Team.....	D-22

D.12.1	Procedure Specification.....	D-22
D.12.2	Parameter Descriptions	D-23
D.13	Update Sales Lead Contacts.....	D-23
D.13.1	Procedure Specification.....	D-24
D.13.2	Parameter Descriptions	D-24
D.14	Delete Sales Lead Contacts	D-25
D.14.1	Procedure Specification.....	D-26
D.14.2	Parameter Descriptions	D-26

Glossary

Index

Send Us Your Comments

Oracle Leads Management Implementation and Administration Guide, Release 11i

Part No. B13534-01

Oracle welcomes your comments and suggestions on the quality and usefulness of this document. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most?

If you find any errors or have any other suggestions for improvement, please indicate the document title and part number, and the chapter, section, and page number (if available). You can send comments to us in the following ways:

- Electronic mail: appsdoc_us@oracle.com
- FAX: (650) 506-7200 Attn: Oracle Applications Documentation Manager
- Postal service:
Oracle Corporation
Oracle Applications Documentation Manager
500 Oracle Parkway
Redwood Shores, CA 94065
USA

If you would like a reply, please give your name, address, telephone number, and (optionally) electronic mail address.

If you have problems with the software, please contact your local Oracle Support Services.

Preface

Welcome to the Oracle Leads Management Implementation and Administration Guide, Release 11*i*.

This guide assumes you have a working knowledge of the following:

- The principles and customary practices of your business area.
- Oracle Marketing and Oracle Leads Management.

If you have never used Oracle Leads Management, Oracle suggests you attend one or more of the Oracle Leads Management training classes available through Oracle University.

- Oracle Self-Service Web Applications.

To learn more about Oracle Self-Service Web Applications, read the *Oracle Self-Service Web Applications Implementation Manual*.

- The Oracle Applications graphical user interface.

To learn more about the Oracle Applications graphical user interface, read the *Oracle Applications User's Guide*.

See [Other Information Sources](#) for more information about Oracle Applications product information.

How To Use This Guide

The Oracle Leads Management Oracle Leads Management Implementation and Administration Guide contains the information you need to understand and use Oracle Leads Management. This guide contains seven chapters:

- [Chapter 1](#) provides an overview of Oracle Leads Management.

- [Chapter 2](#) provides business flows and scenarios that help you to understand Oracle Leads Management better.
- [Chapter 3](#) describes the setup tasks that you need to perform to process the captured records.
- [Chapter 4](#) describes the functioning of the Interaction Matching Engine.
- [Chapter 5](#) describes the functioning of the Leads Processing Engines.
- [Chapter 6](#) describes the functioning of the Monitoring Engine.
- [Chapter 7](#) describes the various operational reports in Oracle Leads Management.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For additional information, visit the Oracle Accessibility Program Web site at <http://www.oracle.com/accessibility/>

Accessibility of Code Examples in Documentation

JAWS, a Windows screen reader, may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, JAWS may not always read a line of text that consists solely of a bracket or brace.

Other Information Sources

You can choose from many sources of information, including documentation, training, and support services, to increase your knowledge and understanding of Oracle Leads Management.

If this guide refers you to other Oracle Applications documentation, use only the Release 11*i* versions of those guides.

Online Documentation

All Oracle Applications documentation is available online (HTML or PDF).

- **PDF Documentation**- See the Documentation CD provided with each release for current PDF documentation for your product. This Documentation CD is also available on *OracleMetaLink* and is updated frequently.
- **Online Help** - You can refer to Oracle iHelp for current HTML online help for your product. Oracle provides patchable online help, which you can apply to your system for updated implementation and end user documentation. No system downtime is required to apply online help.
- **11i Release Content Document** - Refer to the Release Content Document for new features listed release. The Release Content Document is available on *OracleMetaLink*.
- **About document** - Refer to the About document for patches that you have installed to learn about new documentation or documentation patches that you can download. The new About document is available on *OracleMetaLink*.

Related Guides

Oracle Leads Management shares business and setup information with other Oracle Applications products. Therefore, you may want to refer to other guides when you set up and use Oracle Leads Management.

You can read the guides online by choosing Library from the expandable menu on your HTML help window, by reading from the Oracle Applications Document Library CD included in your media pack, or by using a Web browser with a URL that your system administrator provides.

If you require printed guides, you can purchase them from the Oracle Store at <http://oraclestore.oracle.com>.

Guides Related to All Products

Oracle Applications User's Guide

This guide explains how to enter data, query, run reports, and navigate using the graphical user interface (GUI). This guide also includes information on setting user profiles, as well as running and reviewing reports and concurrent processes.

You can access this user's guide online by choosing "Getting Started with Oracle Applications" from any Oracle Applications help file.

Guides Related to This Product

Oracle Self–Service Web Applications Implementation Guide

This manual contains detailed information about the overview and architecture and setup of Oracle Self–Service Web Applications. It also contains an overview of and procedures for using the Web Applications Dictionary.

Oracle Trading Community Architecture Data Quality Management User Guide

This manual contains detailed information about Data Quality Management to manage duplicate parties in your TCA registry.

Installation and System Administration

Oracle Applications Concepts

This guide provides an introduction to the concepts, features, technology stack, architecture, and terminology for Oracle Applications Release 11*i*. It provides a useful first book to read before an installation of Oracle Applications. This guide also introduces the concepts behind Applications-wide features such as Business Intelligence (BIS), languages and character sets, and Self-Service Web Applications.

Installing Oracle Applications

This guide provides instructions for managing the installation of Oracle Applications products. In Release 11*i*, much of the installation process is handled using Oracle Rapid Install, which minimizes the time to install Oracle Applications and the Oracle technology stack by automating many of the required steps. This guide contains instructions for using Oracle Rapid Install and lists the tasks you need to perform to finish your installation. You should use this guide in conjunction with individual product user guides and implementation guides.

Oracle Applications Implementation Wizard User Guide

If you are implementing more than one Oracle product, you can use the Oracle Applications Implementation Wizard to coordinate your setup activities. This guide describes how to use the wizard.

Upgrading Oracle Applications

Refer to this guide if you are upgrading your Oracle Applications Release 10.7 or Release 11.0 products to Release 11*i*. This guide describes the upgrade process and lists database and product-specific upgrade tasks. You must be either at Release 10.7

(NCA, SmartClient, or character mode) or Release 11.0, to upgrade to Release 11*i*. You cannot upgrade to Release 11*i* directly from releases prior to 10.7.

“About” Document

For information about implementation and user documentation, instructions for applying patches, new and changed setup steps, and descriptions of software updates, refer to the “About” document for your product. “About” documents are available on Oracle *MetaLink* for most products starting with Release 11.5.8.

Maintaining Oracle Applications

Use this guide to help you run the various AD utilities, such as AutoUpgrade, AutoPatch, AD Administration, AD Controller, AD Relink, License Manager, and others. It contains how-to steps, screenshots, and other information that you need to run the AD utilities. This guide also provides information on maintaining the Oracle applications file system and database.

Oracle Applications System Administrator’s Guide

This guide provides planning and reference information for the Oracle Applications System Administrator. It contains information on how to define security, customize menus and online help, and manage concurrent processing.

Oracle Alert User’s Guide

This guide explains how to define periodic and event alerts to monitor the status of your Oracle Applications data.

Oracle Applications Developer’s Guide

This guide contains the coding standards followed by the Oracle Applications development staff and describes the Oracle Application Object Library components that are needed to implement the Oracle Applications user interface described in the *Oracle Applications User Interface Standards for Forms-Based Products*. This manual also provides information to help you build your custom Oracle Forms Developer forms so that the forms integrate with Oracle Applications.

Oracle Applications User Interface Standards for Forms-Based Products

This guide contains the user interface (UI) standards followed by the Oracle Applications development staff. It describes the UI for the Oracle Applications products and how to apply this UI to the design of an application built by using Oracle Forms.

Other Implementation Documentation

Oracle Applications Product Update Notes

Use this guide as a reference for upgrading an installation of Oracle Applications. It provides a history of the changes to individual Oracle Applications products between Release 11.0 and Release 11*i*. It includes new features, enhancements, and changes made to database objects, profile options, and seed data for this interval.

Oracle Workflow Administrator's Guide

This guide explains how to complete the setup steps necessary for any Oracle Applications product that includes workflow-enabled processes, as well as how to monitor the progress of runtime workflow processes.

Oracle Workflow Developer's Guide

This guide explains how to define new workflow business processes and customize existing Oracle Applications-embedded workflow processes. It also describes how to define and customize business events and event subscriptions.

Oracle Workflow User's Guide

This guide describes how Oracle Applications users can view and respond to workflow notifications and monitor the progress of their workflow processes.

Oracle Workflow API Reference

This guide describes the APIs provided for developers and administrators to access Oracle Workflow.

Oracle Applications Flexfields Guide

This guide provides flexfields planning, setup and reference information for the Oracle Leads Management implementation team, as well as for users responsible for the ongoing maintenance of Oracle Applications product data. This guide also provides information on creating custom reports on flexfields data.

Oracle eTechnical Reference Manuals

Each eTechnical Reference Manual (eTRM) contains database diagrams and a detailed description of database tables, forms, reports, and programs for a specific Oracle Applications product. This information helps you convert data from your existing applications, integrate Oracle Applications data with non-Oracle applications, and write custom reports for Oracle Applications products. Oracle eTRM is available on Oracle *Metalink*

Oracle Applications Message Manual

This manual describes all Oracle Applications messages. This manual is available in HTML format on the documentation CD-ROM for Release 11*i*.

Training and Support

Training

Oracle offers a complete set of training courses to help you and your staff master Oracle Leads Management and reach full productivity quickly. These courses are organized into functional learning paths, so you take only those courses appropriate to your job or area of responsibility.

You have a choice of educational environments. You can attend courses offered by Oracle University at any one of our many education centers, you can arrange for our trainers to teach at your facility, or you can use Oracle Learning Network (OLN), Oracle University's online education utility. In addition, Oracle training professionals can tailor standard courses or develop custom courses to meet your needs. For example, you may want to use your organization structure, terminology, and data as examples in a customized training session delivered at your own facility.

Support

From on-site support to central support, our team of experienced professionals provides the help and information you need to keep Oracle Leads Management working for you. This team includes your technical representative, account manager, and Oracle's large staff of consultants and support specialists with expertise in your business area, managing an Oracle server, and your hardware and software environment.

Oracle*MetaLink*

OracleMetaLink is your self-service support connection with web, telephone menu, and e-mail alternatives. Oracle supplies these technologies for your convenience, available 24 hours a day, 7 days a week. With *OracleMetaLink*, you can obtain information and advice from technical libraries and forums, download patches, download the latest documentation, look at bug details, and create or update TARs. To use *MetaLink*, register at (<http://metalink.oracle.com>).

Alerts: You should check *OracleMetaLink* alerts before you begin to install or upgrade any of your Oracle Applications. Navigate to the Alerts page as follows:

Technical Libraries/ERP Applications/Applications Installation and Upgrade/Alerts.

Self-Service Toolkit: You may also find information by navigating to the Self-Service Toolkit page as follows: Technical Libraries/ERP Applications/Applications Installation and Upgrade.

Do Not Use Database Tools to Modify Oracle Applications Data

*Oracle STRONGLY RECOMMENDS that you never use SQL*Plus, Oracle Data Browser, database triggers, or any other tool to modify Oracle Applications data unless otherwise instructed.*

Oracle provides powerful tools you can use to create, store, change, retrieve, and maintain information in an Oracle database. But if you use Oracle tools such as SQL*Plus to modify Oracle Applications data, you risk destroying the integrity of your data and you lose the ability to audit changes to your data.

Because Oracle Applications tables are interrelated, any change you make using Oracle Applications can update many tables at once. But when you modify Oracle Applications data using anything other than Oracle Applications, you may change a row in one table without making corresponding changes in related tables. If your tables get out of synchronization with each other, you risk retrieving erroneous information and you risk unpredictable results throughout Oracle Applications.

When you use Oracle Applications to modify your data, Oracle Applications automatically checks that your changes are valid. Oracle Applications also keeps track of who changes information. If you enter information into database tables using database tools, you may store invalid information. You also lose the ability to track who has changed your information because SQL*Plus and other database tools do not keep a record of changes.

About Oracle

Oracle develops and markets an integrated line of software products for database management, applications development, decision support, and office automation, as well as Oracle Applications, an integrated suite of more than 160 software modules for financial management, supply chain management, manufacturing, project systems, human resources and customer relationship management.

Oracle products are available for mainframes, minicomputers, personal computers, network computers and personal digital assistants, allowing organizations to integrate different computers, different operating systems, different networks, and

even different database management systems, into a single, unified computing and information resource.

Oracle is the world's leading supplier of software for information management, and the world's second largest software company. Oracle offers its database, tools, and applications products, along with related consulting, education, and support services, in over 145 countries around the world.

Introduction to Oracle Leads Management

This chapter introduces you to Oracle Leads Management and its features. The new features and enhancements for this release, obsolete features, dependencies are also briefly described.

Topics included are:

- [Section 1.1, "Oracle Leads Management Overview"](#)
- [Section 1.2, "New Features and Enhancements in this Release"](#)
- [Section 1.3, "Oracle Leads Management Dependencies"](#)
- [Section 1.4, "Upgrading from Release 11.5.9"](#)

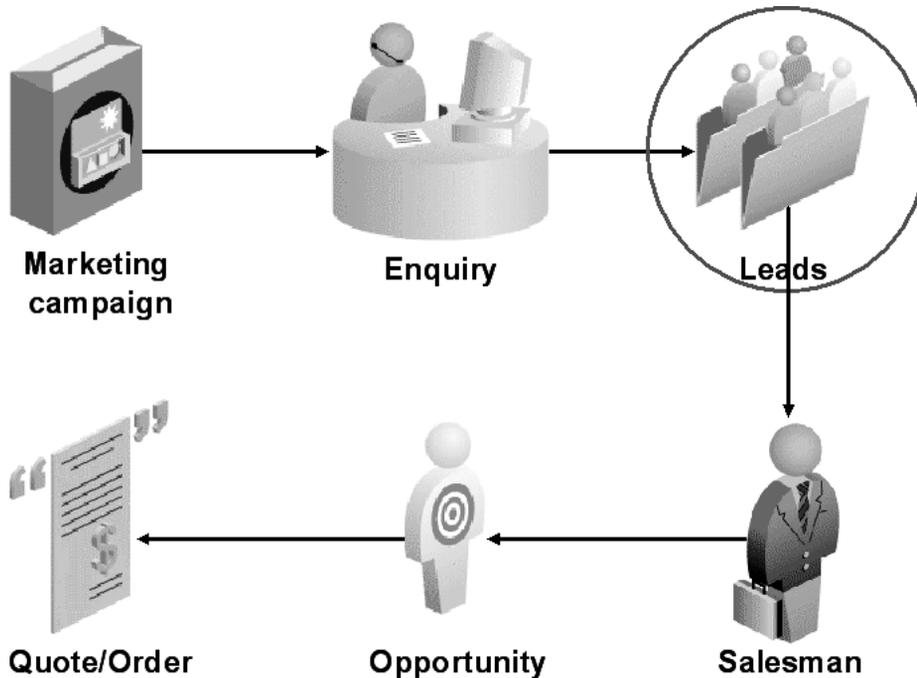
1.1 Oracle Leads Management Overview

Oracle Leads Management provides solutions to automate and optimize prospect-to-sales conversion across the enterprise. It provides a staging area for all prospective leads. In the staging area, the leads go through data quality processing, qualification, and prioritization before being assigned to sales teams for conversion.

1.1.1 Leads in the Marketing and Sales Cycles

Where does a lead appear in the Marketing and Sales cycles? [Figure 1-1](#) illustrates this.

Figure 1-1 *Leads in the Marketing and Sales Cycles*



1. **Marketing Campaign:** The marketing department kicks off a campaign by targeting prospective customers.
2. **Enquiry:** In response to the campaign, interested prospects make an enquiry. At this point, the details of the prospects are captured.

3. **Leads:** The details of prospects who respond to a campaign or who make an enquiry are stored as leads. Leads are prospective customers. After the leads are processed, they are assigned to appropriate sales teams.
4. **Opportunity:** The sales teams convert some leads to opportunities. An opportunity is still a prospect, but more mature. An opportunity is more likely to be converted to an order.
5. **Quote/Order:** Promising opportunities receive a quote from the sales team. When the quote is accepted, the opportunity become an order.

1.1.2 Oracle Leads Management Features

The Oracle Leads Management features include:

- **Capture of leads** - Leads are captured from various sources such as marketing campaigns, events, and referrals.
- **Customer and lead data quality** - Captured records are processed for customer and leads quality.
- **Interactions linked to leads** - Relevant interactions are attached to leads, and wherever appropriate, new leads are generated from interactions.
- **Real time flexible rules engine** - The Leads Processing Engines can be set up for lead filtering, evaluation and distribution based on business rules.
- **Integrated monitoring** - Sales-ready leads are monitored for prompt action by the sales team.
- **Lead utilization and effectiveness analysis** - Using the operational reports, the effectiveness of the engines and the status of the leads processed by them are tracked.

1.2 New Features and Enhancements in this Release

This section describes the Leads functionality to be delivered in the Oracle E-Business Suite 11.5.10 release. If you are implementing this product prior to the release, using product minipacks or family packs, some new functionality may be dependent on integration with other Oracle products. Please consult *MetaLink* for relevant product patches and documentation.

1.2.1 New Features

The new features for this release are described in the following sections.

- [Section 1.2.1.1, "Rule-based Deduplication"](#)
- [Section 1.2.1.2, "Linking Interactions to Lead Flow"](#)

1.2.1.1 Rule-based Deduplication

During the process of capturing leads, the data is checked for duplicates in two phases. Customer deduplication is performed first by Data Quality Management (DQM), followed by leads deduplication. Leads deduplication is done to prevent entry of duplicate lead records.

In release 11.5.10, leads deduplication is done by means of a rule. Based on the definition for the rule, the lead attributes are compared with existing lead attributes in the Sales Leads table. A predefined deduplication rule is provided based on the deduplication logic in release 11.5.9. You can customize this rule.

1.2.1.2 Linking Interactions to Lead Flow

In release 11.5.10, Oracle Leads Management provides rules for linking interactions to leads flow. Using the Interaction Matching Engine, you can create rules that evaluate marketing interactions and attach them to the relevant leads. For example, if a party has asked for more details about a product, this interaction is linked to the party's lead record. Based on rules set up, if a lead does not exist for an interaction, a new lead may also be generated.

1.2.2 Enhancements

The enhancements for this release are discussed in the following sections.

- [Section 1.2.2.1, "Disqualifying Leads"](#)
- [Section 1.2.2.2, "Leads Rules Engine Usability"](#)
- [Section 1.2.2.3, "Leads Rules Performance Report"](#)
- [Section 1.2.2.4, "Supporting DUNS Number"](#)
- [Section 1.2.2.5, "Enhancing Performance"](#)

1.2.2.1 Disqualifying Leads

The Qualification Engine has been enhanced to identify unqualified leads. Based on the rules setup, unqualified leads are graded NULL, and are routed to the Channel Selection Engine.

1.2.2.2 Leads Rules Engine Usability

In release 11.5.10, the Leads Processing Engine provides easier creation and deployment of rule sets. The following enhancements have been made:

- **Campaign-specific Lead Evaluation:** You can configure one or more rule sets in the Leads Rules Engine that are appropriate to different types of campaigns, or templates. This enables a marketing user to seamlessly leverage a pre-defined rules flow for leads processing and tracking.
- **Central Tracing Views:** The Leads Rules Engine provides a central tracing view to enable easier deployment and testing of rule setups. The central tracing view supports grouping many rule sets by common attributes such as Campaign Type, Campaign, Country, Product, and Industry. You can enter lead attributes and trace the flow of a lead to identify redundancy or conflict in the rule set configuration, and update rule sets from this view.
- **Derived Attributes:** Derived attributes are custom attributes set up for leads evaluation. The derived attributes user interface in Oracle Leads Management is enhanced to support only leads data, and to support testing and validation of functions that are used to evaluate leads.
- **Rule Sets User Interface:** The rule sets user interface has been enhanced to support less clicks while creating rule sets.

1.2.2.3 Leads Rules Performance Report

The Leads Rules Performance report, which provides utilization and effectiveness performance for leads evaluation, has been enhanced to support additional actual comparisons to leads rules output.

1.2.2.4 Supporting DUNS Number

Lead Import now supports importing of DUNS numbers with leading zeros. The DUNS number is stored in character format.

1.2.2.5 Enhancing Performance

The Import Sales Lead concurrent program now runs in parallel with a parent process and multiple child processes to improve the speed of processing many records in a batch. For more information, see [Section 3.4.4, "Enhancing the Performance of the Import Sales Lead Concurrent Program"](#).

1.3 Oracle Leads Management Dependencies

Oracle Leads Management is dependent upon Oracle Field Sales (OFS) and Oracle TeleSales (OTS) for its proper functioning. Ensure that both these applications are installed and implemented correctly. For more information on OFS and OTS, see the *Oracle Field Sales Implementation Guide* and the *Oracle TeleSales Implementation Guide*.

Additional documents related to the modules discussed in this guide are referred to in the preface.

1.4 Upgrading from Release 11.5.9

For the following features, you must perform specific tasks to upgrade to release 11.5.10 successfully. This is not applicable for new 11.5.10 users.

- [Section 1.4.1, "Single Product Catalog"](#)
- [Section 1.4.2, "Monitoring Engine"](#)

1.4.1 Single Product Catalog

Oracle Leads Management does not support the use of the Sales Interest Types and Sales Interest Codes from release 11.5.10. For more information and steps to upgrade, see the *Oracle Advanced Product Catalog User Guide*.

1.4.2 Monitoring Engine

In release 11.5.9, if both the Country and Rating attributes were defined for a monitor rule, a lead satisfying either of the attributes were picked for monitoring. However, from release 11.5.10 onwards, if both the attributes are defined for the monitor rule, a lead must satisfy both the attributes to be monitored. For more information, see [Section 6.1.1.1, "Upgrading from Release 11.5.9"](#).

Business Flows

This chapter provides business flows and scenarios that will help you understand Oracle Leads Management better.

Topics included are:

- [Section 2.1, "Business Flows in Oracle Leads Management"](#)
- [Section 2.2, "Scenarios Using the Leads Processing Engine"](#)

2.1 Business Flows in Oracle Leads Management

In Oracle Leads Management, leads are captured and processed before they are assigned to appropriate sales channels. The following sections give a high-level flow of a lead in Oracle Leads Management. Each of the processes are discussed at length in their respective chapters.

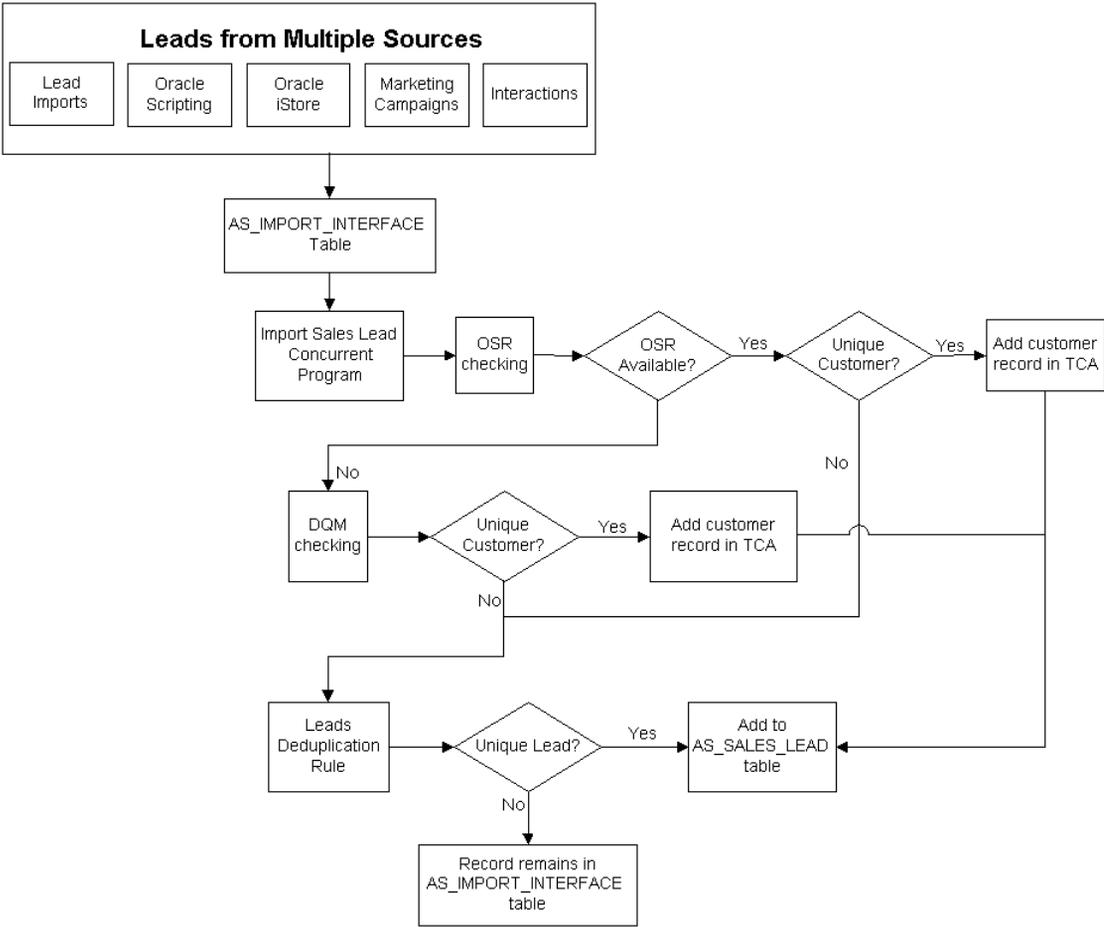
Topics in this section include:

- [Section 2.1.1, "Capturing and Cleaning Leads"](#)
- [Section 2.1.2, "Processing Leads"](#)

2.1.1 Capturing and Cleaning Leads

[Figure 2–1](#) gives the flow of leads after they are stored in the AS_IMPORT_INTERFACE table. For more information on the import process, see [Chapter 3, "Capturing and Cleaning Leads"](#).

Figure 2-1 Business Flow for Capturing and Cleaning Leads



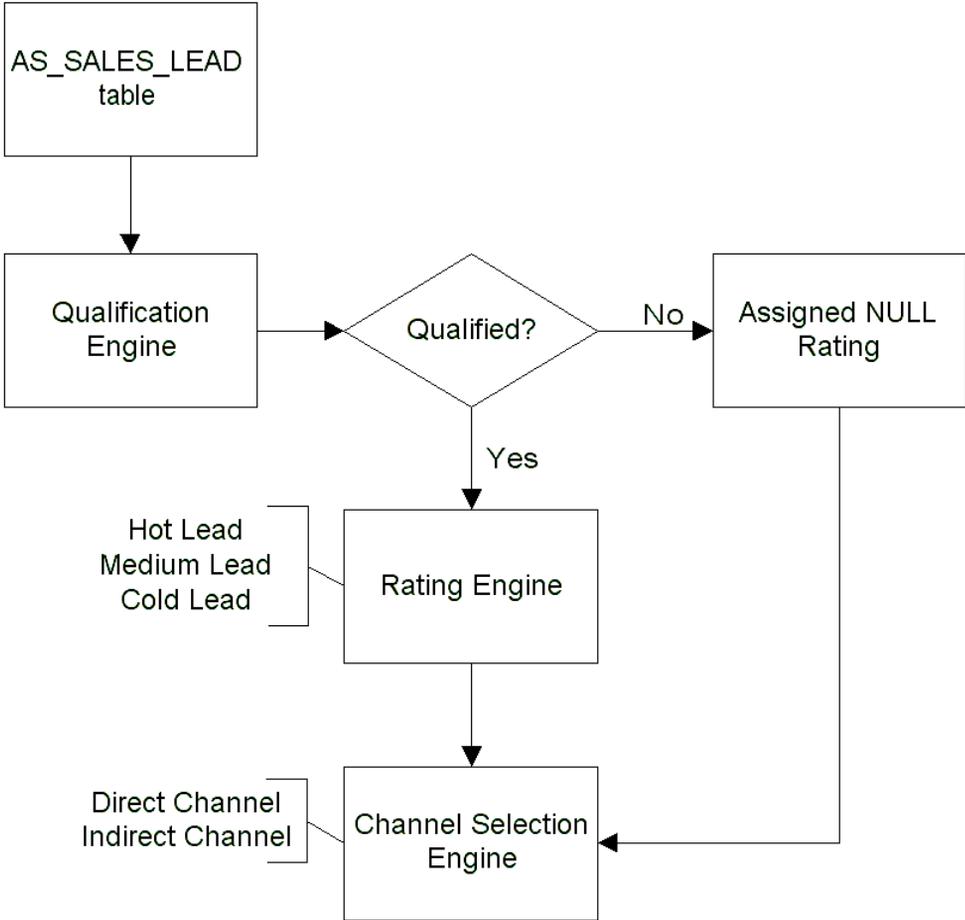
1. Leads enter Oracle Leads Management from multiple sources: Lead Imports, Oracle Scripting, Oracle iStore, Marketing Campaigns, and Interactions. The lead records are stored in the AS_IMPORT_INTERFACE table.
2. The Import Sales Lead concurrent program processes each lead record in the AS_IMPORT_INTERFACE table in the following order and manner:
 - a. The lead record is checked for the existence of Original System Reference.

- b. Next, Data Quality Management (DQM) checks if a customer record for the lead already exists in TCA. If a record does not exist, then a new record is created in TCA.
- c. After DQM, the record is run through the Leads Deduplication rule to check if the lead record exists in the AS_SALES_LEAD table. If it does not exist, then the record is added to the AS_SALES_LEAD table.

2.1.2 Processing Leads

[Figure 2-2](#) gives the flow of leads after they enter the AS_SALES_LEAD table. For more information on the processing of leads, see [Chapter 5, "Processing Leads"](#).

Figure 2-2 Business Flow for Processing Leads



1. The leads in the AS_SALES_LEAD table are processed by the Lead Processing Engine.
2. The first engine that processes leads is the Qualification Engine. If the lead is qualified, then it is routed to the Rating Engine. If not, it is graded NULL, and routed to the Channel Selection Engine.
3. The Rating Engine assigns a rating to all the qualified leads. Examples of the grades could be Hot, Medium, Cold, and so on.

4. Based on the rating assigned to the lead, the Channel Selection engine assigns a sales channel to the lead. Examples of channel are Direct, Indirect, and so on.

2.2 Scenarios Using the Leads Processing Engine

Two scenarios are presented in this section to understand the setting up of the Leads Processing Engine. The Qualification, Rating, and Channel Selection engines make up the Leads Processing Engine.

- [Section 2.2.1, "Scenario For Qualifying Leads"](#)
- [Section 2.2.2, "Scenario For Disqualifying Leads"](#)

2.2.1 Scenario For Qualifying Leads

In this scenario, the qualification engine is used to qualify leads.

Vision Enterprises is into the business of selling computers and computer accessories to organizations. Recently, they have run a road show called VisionHardware to showcase sleek monitors, lightweight laptops, and high-end computers. Participating organizations were asked to fill forms with details such as product interest, budget status, and so on. These details have been captured, and they must now be processed by Oracle Leads Management so that the leads are followed up appropriately.

The Qualification, Rating, and Channel Selection engines must be set up. The following section details the rule sets set up by Vision Enterprises to process the leads. Create the rule sets based on the following model.

- [Section 2.2.1.1, "Using the Qualification Engine to Qualify Leads"](#)
- [Section 2.2.1.2, "Creating the Qualification Rule Set"](#)
- [Section 2.2.1.3, "Using the Rating Engine"](#)
- [Section 2.2.1.4, "Creating the Rating Rule Set"](#)
- [Section 2.2.1.5, "Using the Channel Selection Engine"](#)
- [Section 2.2.1.6, "Creating the Channel Selection Rule Set"](#)

2.2.1.1 Using the Qualification Engine to Qualify Leads

You can use the Qualification Engine to either qualify or disqualify leads. We use it to qualify leads in this scenario. A lead can be qualified when its attributes suggest interest in your products or services.

In this scenario, because we are using the engine only to qualify leads, the leads that are not processed by the engine are routed to the Channel Selection Engine. Qualified leads are routed to the Rating Engine.

[Table 2-1](#) details the conditions for the qualification engine rule set.

Table 2-1 Qualification Rule Set Conditions

Rule Set Component	Conditions	Remarks
Guards	Country=US Product Category=Monitors or Laptops or Computers	Guards define the domain of the rule set. Only those leads from the US and with interest in monitors, laptops, or computers will be processed by this rule set.
Rules	Qualified Flag = YES	The Qualification Flag for all qualified leads is set to Yes. This is applicable to both Rule 1 and Rule 2.
Rule 1	Purchase Timeframe=Within 1 month, 1-3 months, 3-6 months Purchase Amount Greater Than or Equals 50,000	Rules determine the conditions and action to be performed on the lead. Leads satisfying this criteria are qualified.
Rule 2	Purchase Timeframe=More than 1 Year Purchase Amount Less Than 50,000	Leads satisfying this criteria are qualified.

2.2.1.2 Creating the Qualification Rule Set

Use the following procedure to create the qualification rule set based on the conditions detailed in [Table 2-1](#).

Prerequisites

None

Steps

1. Log in as an administrator, and navigate to Administration > Leads > Processing Rules > Qualification.

2. Click **Create**.
3. Enter the details for the rule set. Set the Precedence to 100.
4. Click **Create**.

You will receive a confirmation message that the rule set has been created. Now, you will add the Guards for the rule set.

Creating Guards

5. In the Rule Set: Guard region, click **Add Attributes**.
6. From the Profile Attribute Name's second drop-down list, select Country.
This is the default.
7. Leave the Condition as Equals.
8. From the Value list, select United States.
9. Click **Apply and Add Another**.
10. From the Profile Attribute Name's second drop-down list, select Product Category.
11. Leave the Condition as Equals.
12. From the Value list, select Monitors, Laptops, Computers.
13. Click **Apply**.

Next, you will go on to add Rules for this rule set.

Creating Rules

14. Click the Rules link.
15. Click **Add Rule**.
16. Enter the name for the rule.
17. Enter 1 in the Order of Evaluation field.
18. From the If the condition is met, 'Qualified Flag' is set to drop-down list, select Yes.
19. Click **Add Attributes**.
20. From the Profile Attribute Name's second drop-down list, select Purchase Timeframe.

21. Leave the Condition as Equals.
22. From the Value list, select Within 1 month, 1-3 months, and 3-6 months.
23. Click **Apply and Add Another**.
24. From the Profile Attribute Name's second drop-down list, select Purchase Amount - Product.
25. From the Condition drop-down list, select Greater Than or Equals.
26. Enter 50000 in the Value field.
27. Click **Apply**.

Next, create Rule 2 as per the specifications in [Table 2-1](#). For the steps to create the rule, see [Creating Rules](#).

Now the Guards and Rules for the rule set are created. Creating the Qualification Rule Set is complete.

2.2.1.3 Using the Rating Engine

The leads that are qualified are routed to the Rating Engine. Based on the attributes of the lead, Vision Enterprises uses the following ratings: A, B, C and D to assign a rating.

[Table 2-2](#) details the conditions for the rating engine rule set.

Table 2-2 Rating Engine Rule Set Conditions

Rule Set Component	Conditions	Remarks
Guards	Country=US Product Category=Monitors or Laptops or Computers	Guards define the domain of the rule set. Only those leads from the US and with interest in monitors, laptops, or computers will be processed by this rule set.
Rule 1	Purchase Timeframe=Within 1 month, and Budget Status=Approved: Grade A	Rules determine the conditions and action to be performed on the lead. Leads satisfying this criteria are rated Grade A.

Table 2–2 Rating Engine Rule Set Conditions

Rule Set Component	Conditions	Remarks
Rule 2	Purchase Timeframe=1-3 months, and Budget Status=Approved: Grade B	Leads satisfying this criteria are rated Grade B.
Rule 3	Purchase Timeframe=3-6 months, and Budget Status=Approved: Grade C	Leads satisfying this criteria are rated Grade C.
Rule 4	Purchase Timeframe=3-6 months, and Budget Status=Pending: Grade D	Leads satisfying this criteria are rated Grade D.

2.2.1.4 Creating the Rating Rule Set

Use the following procedure to create the rating rule set based on the conditions detailed in [Table 2–2](#).

Prerequisites

None

Steps

1. Log in as an administrator, and navigate to Administration > Leads > Processing Rules > Rating.
2. Click **Create**.
3. Enter the details for the rule. Set the Precedence to 100.
4. Click **Create**.

You will receive a confirmation message that the rule set has been created. Now, you will add the Guards for the rule set.

Creating Guards

5. In the Rule Set: Guard region, click **Add Attributes**.
6. From the Profile Attribute Name's second drop-down list, select Country.
This is the default.
7. Leave the Condition as Equals.
8. From the Value list, select US.

9. Click **Apply and Add Another**.
10. From the Profile Attribute Name's second drop-down list, select Product Category.
11. Leave the Condition as Equals.
12. From the Value list, select Monitors, Laptops, Computers.
13. Click **Apply**.

Next, you will go on to add Rules for this rule set.

Creating Rules

14. Click the Rules link.
15. Click **Add Rule**.
16. Enter a name for the rule set, and 1 for the Order of Evaluation.
17. From the If the condition is met, Lead Rank is set to drop-down list, select Grade A.
18. Click **Add Attributes**.
19. From the Profile Attribute Name's second drop-down list, select Purchase Timeframe.
20. Leave the Condition as Equals.
21. From the Value list, select Within 1 month.
22. Click **Apply and Add Another**.
23. From the Profile Attribute Name's second drop-down list, select Budget Status.
24. Leave the Condition as Equals.
25. From the Value list, select Approved.
26. Click **Apply**.

Next, create Rule 2, 3, and 4 as per the specifications in [Table 2-2](#). For the steps to create the rule, see [Creating Rules](#).

Now the Guards and Rules for the rule set are created. Creating the Rating Rule Set is complete.

2.2.1.5 Using the Channel Selection Engine

You can use the Channel Selection Engine to assign a sales channel to the leads. The Territory Assignment Program decides the sales teams to assign the leads to using the sales channel. Vision Enterprises uses the Direct and Indirect Channels.

Table 2–3 details the conditions for the channel selection engine rule set.

Table 2–3 Channel Selection Engine Rule Set Conditions

Rule Set Component	Conditions	Remarks
Guards	Country=US	Guards define the domain of the rule set. Only those leads from the US will be processed by this rule set.
Rule 1	Qualified Flag = Yes and Purchase Amount > 50,000, route to Direct Channel	Rules determine the conditions and action to be performed on the lead. Leads satisfying this criteria are routed to the Direct Channel.
Rule 2	Qualified Flag = Yes, Purchase Amount < 50,000, and Lead Rating Equals A, route to Indirect Channel A	Leads satisfying this criteria are routed to Indirect Channel A.
Rule 3	Qualified Flag = Yes, Purchase Amount < 50,000, and Lead Rating Not Equals A, route to Indirect Channel B	Leads satisfying this criteria are routed to Indirect Channel B.

2.2.1.6 Creating the Channel Selection Rule Set

Use the following procedure to create the channel selection rule set based on the conditions detailed in Table 2–3.

Prerequisites

None

Steps

1. Log in as an administrator, and navigate to Administration > Leads > Processing Rules > Channel Selection.

2. Click **Create**.
3. Enter the details for the rule. Set the Precedence to 100.
4. Click **Create**.

You will receive a confirmation message that the rule set has been created. Now, you will add a Guard for the channel selection rule set.

Creating a Guard

5. In the Rule Set: Guard region, click **Add Attributes**.
6. From the Profile Attribute Name's second drop-down list, select Country.
This is the default.
7. Leave the Condition as Equals.
8. From the Value list, select US.
9. Click **Apply**.

Next, you will go on to add Rules for this rule set.

Creating Rules

10. Click the Rules link.
11. Click **Add Rule**.
12. Enter a name for the rule set, and 1 for the Order of Evaluation.
13. Select Direct Channel.
14. From the If the condition is met, Sales Channel is set to drop-down list, select Direct Channel.
15. Click **Add Attributes**.
16. From the Profile Attribute Name's second drop-down list, select Purchase Amount - Product.
17. From the Condition drop-down list, select Greater Than or Equals.
18. In the Value column, enter 50,000.
19. Click **Apply**.

Next, create Rule 2, 3, and 4 as per the specifications in [Table 2–3](#). For the steps to create the rule, see [Creating Rules](#).

Now the Guards and Rules for the rule set are created. Creating the Channel Selection Rule Set is complete.

2.2.2 Scenario For Disqualifying Leads

In this scenario, the qualification engine is used to disqualify leads.

Vision Communications is a company in the communication tools business. The company is the market leader in audio equipment and is now expanding into other communication equipment channels.

Using its current install base of large-scale voice customers, Vision plans a web marketing campaign called VisionVideos to cross-sell its video communication equipment to increase sales growth. The plan is to process the respondents to this campaign through Oracle Leads Management and route them to appropriate sales channels. The following section details the rule sets set up by Vision Communications to process the leads.

To process the leads, you must set up the Qualification, Rating and Channel Selection Engines. Create the rule sets based on the following model.

- [Section 2.2.2.1, "Using the Qualification Engine to Disqualify Leads"](#)
- [Section 2.2.2.2, "Creating the Qualification Rule Set"](#)
- [Section 2.2.2.3, "Lead Rating Rule Set"](#)
- [Section 2.2.2.4, "Creating the Rating Rule Set"](#)
- [Section 2.2.2.5, "Lead Channel Selection Rule Set"](#)
- [Section 2.2.2.6, "Creating the Channel Selection Rule Set"](#)

2.2.2.1 Using the Qualification Engine to Disqualify Leads

You can use the Qualification Engine to either qualify leads or disqualify leads. We use it to disqualify leads in this scenario.

A lead may be disqualified in the following situations:

- The lead's attributes do not have a contact address or e-mail ID
- The lead's e-mail ID has a competitor name
- The lead is an employee of your company
- The lead has a false name such as Mickey Mouse or XYZ.

You do not want your sales representatives to spend time on such leads.

In this scenario, because we are using the qualification engine only to disqualify leads, the leads that are not processed by the engine are automatically qualified and routed to the Rating Engine. The disqualified leads are graded NULL and routed to the Channel Selection Engine.

[Table 2–4](#) details the conditions for the qualification engine rule set.

Table 2–4 Qualification Rule Set Conditions

Rule Set Component	Conditions	Remarks
Guards	Country 'Not Equals' US Product Category=Monitors or Laptops or Computers	Guards define the domain of the rule set. Only those leads not from the US, and with interest in monitors, laptops, or computers will be processed by this rule set.
Rules	Qualified Flag = NO	The Qualification Flag for all disqualified leads is set to No. This is applicable for both Rule1 and Rule 2.
Rule 1	E-mail Address Contains <i>your company.com</i> Name Contains Mickey Mouse or ABC - any name that maybe invalid.	Leads satisfying this criteria are disqualified.
Rule 2	E-mail Address Is Null Customer Address Is Null	Rules determine the conditions and action to be performed on the lead. Leads satisfying this criteria are disqualified.

2.2.2.2 Creating the Qualification Rule Set

Use the following procedure to create the qualification rule set based on the conditions detailed in [Table 2–4](#).

Prerequisites

None

Steps

1. Log in as an administrator, and navigate to Administration > Leads > Processing Rules > Qualification.
2. Click **Create**.
3. Enter the details for the rule set. Set the Precedence to 100.
4. Click **Create**.

You will receive a confirmation message that the rule set has been created. Now, you will add the Guards for the rule set.

Creating Guards

5. In the Rule Set: Guard region, click **Add Attributes**.
6. From the Profile Attribute Name's second drop-down list, select Country.
This is the default.
7. From the Condition drop-down list, select Not Equals.
8. From the Value list, select United States.
9. Click **Apply and Add Another**.
10. From the Profile Attribute Name's second drop-down list, select Product Category.
11. Leave the Condition as Equals.
12. From the Value list, select Monitors, Laptops, Computers.
13. Click **Apply**.

Next, you will go on to add Rules for this rule set.

Creating Rules

14. Click the Rules link.
15. Click **Add Rule**.
16. Enter the name for the rule.
17. Enter 1 in the Order of Evaluation field.
18. From the If the condition is met, 'Qualified Flag' is set to drop-down list, select No.
19. Click **Add Attributes**.

- 20. From the Profile Attribute Name's second drop-down list, select E-mail Address.
- 21. From the Condition drop-down list, select Contains.
- 22. Enter <your company>.com in the Value field.
People with your company's e-mail address cannot be leads.
- 23. Click **Apply and Add Another**.
- 24. From the Profile Attribute Name's second drop-down list, select Name.
- 25. From the Condition drop-down list, select Contains.
- 26. Enter Mickey Mouse in the Value field.
You can enter any name that seems invalid. For example, ABC or XYZ.
- 27. Click **Apply**.

Next, create Rule 2 as per the specifications in [Table 2-4](#). For the steps to create the rule, see [Creating Rules](#).

Now the Guards and Rules for the rule set are created. Creating the Qualification Rule Set is complete.

2.2.2.3 Lead Rating Rule Set

The leads that are not processed by the qualification engine are automatically qualified and routed to the Rating Engine. The leads will be assigned the following ratings based on their attributes - A+, A, B, C, and D.

[Table 2-5](#) details the conditions for the rating engine rule set.

Table 2-5 Rating Engine Rule Set Conditions

Rule Set Component	Conditions	Remarks
Guards	Country = US Campaign = VisionVideos	Guards define the domain of the rule set. Only those leads from the US and resulting from the VisionVideos campaign will be processed by this rule set.

Table 2–5 Rating Engine Rule Set Conditions

Rule Set Component	Conditions	Remarks
Rule 1	Purchase Timeframe=Within 1 month, Budget Status=Approved: Grade as A+	Rules determine the conditions and action to be performed on the lead. Leads satisfying this criteria are rated Grade A +.
Rule 2	Purchase Timeframe=1-3 months, and Budget Status=Approved: Grade A	Leads satisfying this criteria are rated Grade A.
Rule 3	Purchase Timeframe=3-6 months, and Budget Status=Approved: Grade B	Leads satisfying this criteria are rated Grade B.
Rule 4	Purchase Timeframe=6-12 months, and Budget Status=Approved: Grade C	Leads satisfying this criteria are rated Grade C.
Rule 5	Purchase Timeframe=6-12 months, and Budget Status = Not Approved: Grade D	Leads satisfying this criteria are rated Grade D.

2.2.2.4 Creating the Rating Rule Set

Use the following procedure to create the rating rule set based on the conditions detailed in [Table 2–5](#).

Prerequisites

None

Steps

1. Log in as an administrator, and navigate to Administration > Leads > Processing Rules > Rating.
2. Click **Create**.
3. Enter the details for the rule. Set the Precedence to 100.
4. Click **Create**.

You will receive a confirmation message that the rule set has been created. Now, you will add the Guards for the rule set.

Creating Guards

5. In the Rule Set: Guard region, click **Add Attributes**.
6. From the Profile Attribute Name's second drop-down list, select Country.
This is the default.
7. Leave the Condition as Equals.
8. From the Value list, select US.
9. Click **Apply and Add Another**.
10. From the Profile Attribute Name's second drop-down list, select Campaign.
11. Leave the Condition as Equals.
12. Click **Go** to select the VisionVideos campaign.
13. Click **Apply**.
Next, you will go on to add Rules for this rule set.

Creating Rules

14. Click the Rules link.
15. Click **Add Rule**.
16. Enter a name for the rule set, and 1 for the Order of Evaluation.
17. From the If the condition is met, Lead Rank is set to drop-down list, select Grade A+.
18. Click **Add Attributes**.
19. From the Profile Attribute Name's second drop-down list, select Purchase Timeframe.
20. Leave the Condition as Equals.
21. From the Value list, select Within 1 month.
22. Click **Apply and Add Another**.
23. From the Profile Attribute Name's second drop-down list, select Budget Status.
24. Leave the Condition as Equals.
25. From the Value list, select Approved.
26. Click **Apply**.

Next, create Rule 2, 3, 4, and 5 as per the specifications in [Table 2-5](#). For the steps to create the rule, see [Creating Rules](#).

Now the Guards and Rules for the rule set are created. Creating the Rating Rule Set is complete.

2.2.2.5 Lead Channel Selection Rule Set

All the leads, both qualified and unqualified, are processed by the Channel Selection Engine. Vision Enterprises uses the following channels - Direct, Indirect Channel A, Indirect Channel B, Call Center A, and Call Center B.

[Table 2-6](#) details the conditions for the channel selection engine rule set.

Table 2-6 Channel Selection Engine Rule Set Conditions

Rule Set Component	Conditions	Remarks
Rule Set 1: Guards	Country=US	Guards define the domain of the rule set. Only those leads from the US will be processed by this rule set.
Rule Set 1: Rules 1, 2, and 3	Qualified Flag = Yes, Purchase Amount > 50,000, route to Direct Channel Qualified Flag = Yes, Purchase Amount < 50,000, and Lead Rating Equals A, route to Indirect Channel A Qualified Flag = Yes, Purchase Amount < 50,000, and Lead Rating Not Equals A, route to Indirect Channel B	Rules determine the conditions and action to be performed on the lead. Leads satisfying this criteria are routed appropriately.
Rule Set 2: Guards	Country=US	Only those leads from the US will be processed by this rule set.
Rule Set 2: Rule 1	Qualified Flag = No, route to Call Center Channel A	Leads satisfying this criteria are routed to Call Center Channel A.
Rule Set 3: Guards	Country Not US	Only those leads not from the US will be processed by this rule set.

Table 2–6 Channel Selection Engine Rule Set Conditions

Rule Set Component	Conditions	Remarks
Rule Set 3: Rule 1	Qualified Flag = No, route to Call Center Channel B	Leads satisfying this criteria are routed to Call Center Channel B.

2.2.2.6 Creating the Channel Selection Rule Set

Use the following procedure to create the channel selection rule set based on the conditions detailed in [Table 2–6](#).

Prerequisites

None

Steps

1. Log in as an administrator, and navigate to Administration > Leads > Processing Rules > Channel Selection.
2. Click **Create**.
3. Enter the details for the rule. Set the Precedence to 100.
4. Click **Create**.

You will receive a confirmation message that the rule set has been created. Now, you will add Rules for the channel selection rule set.

Creating Guards

5. In the Rule Set: Guard region, click **Add Attributes**.
6. From the Profile Attribute Name's second drop-down list, select Country. This is the default.
7. Leave the Condition as Equals.
8. From the Value list, select US.
9. Click **Apply**.

Next, you will go on to add Rules for this rule set.

Creating Rules

- 10.** Click the Rules link.
- 11.** Click **Add Rule**.
- 12.** Enter a name for the rule set, and 1 for the Order of Evaluation.
- 13.** From the If the condition is met, Sales Channel is set to drop-down list, select Direct Channel.
- 14.** Click **Add Attributes**.
- 15.** From the Profile Attribute Name's second drop-down list, select Qualify Flag.
- 16.** From the Condition drop-down list, select Equals.
- 17.** In the Value column, select Yes.
- 18.** Click **Apply and Add Another**.
- 19.** From the Profile Attribute Name's second drop-down list, select Purchase Amount - Product.
- 20.** From the Condition drop-down list, select Greater Than.
- 21.** In the Value column, enter 50,000.
- 22.** Click **Apply**.

Next, create Rule 2, and 3 as per the specifications in [Table 2-6](#). For the steps to create the rule, see [Creating Rules](#). Similarly, create Rule Sets 2 and 3 as well.

Now, the Guards and Rules for the rule set are created. Creating the Channel Selection Rule Set is complete.

Capturing and Cleaning Leads

This chapter gives you information about capturing leads, and setting up Oracle Leads Management to cleanse the records captured.

Topics included are:

- [Section 3.1, "Leads from Multiple Sources"](#)
- [Section 3.2, "The Lead Import Process"](#)
- [Section 3.3, "Importing Leads"](#)
- [Section 3.4, "The Import Sales Lead Concurrent Program"](#)
- [Section 3.5, "Data Quality"](#)
- [Section 3.6, "Custom Codes with the Lead Import Program"](#)
- [Section 3.7, "Purging Staged Lead Records"](#)

3.1 Leads from Multiple Sources

Leads are captured into Oracle Leads Management from various sources. The primary sources are:

Oracle Scripting

To develop a personalized relationship with customers, marketing organizations use the branching functionality in Oracle Scripting. The branching functionality responds differently to the input of customers based on their profiles or the answers that they provide to questions. When a customer expresses interest in a product, a lead is created and managed by Oracle Leads Management.

Oracle Scripting contains seeded scripts focused on various marketing activities. These scripts can be used with minimum configuration and can be deployed as call-center scripts or web surveys. In addition to simplifying processes, scripts can help to ensure communication consistency. For more information on implementing seeded scripts, see the *Oracle Marketing Implementation Guide* and the *Oracle Scripting Implementation Guide*.

Oracle iStore

Customers who use Oracle iStore for purchases can be mined in as leads for a cross-sell or an up-sell. Also, when customers abandon a shopping cart before making the final purchase, leads are created from such records, and followed up.

Marketing Campaigns

The marketing department may run several campaigns, and capture leads. These leads may be imported into Oracle Leads Management from a .csv, .txt or a flat file. For more information, see [Section 3.3, "Importing Leads"](#). The marketing campaign generates leads based on the Installbase.

Interactions

An interaction is a record of communication between a potential customer and a company representative. An interaction is generally timed and has an outcome or result that can be tracked. These interactions are tracked and leads are created from them.

Partner Referrals

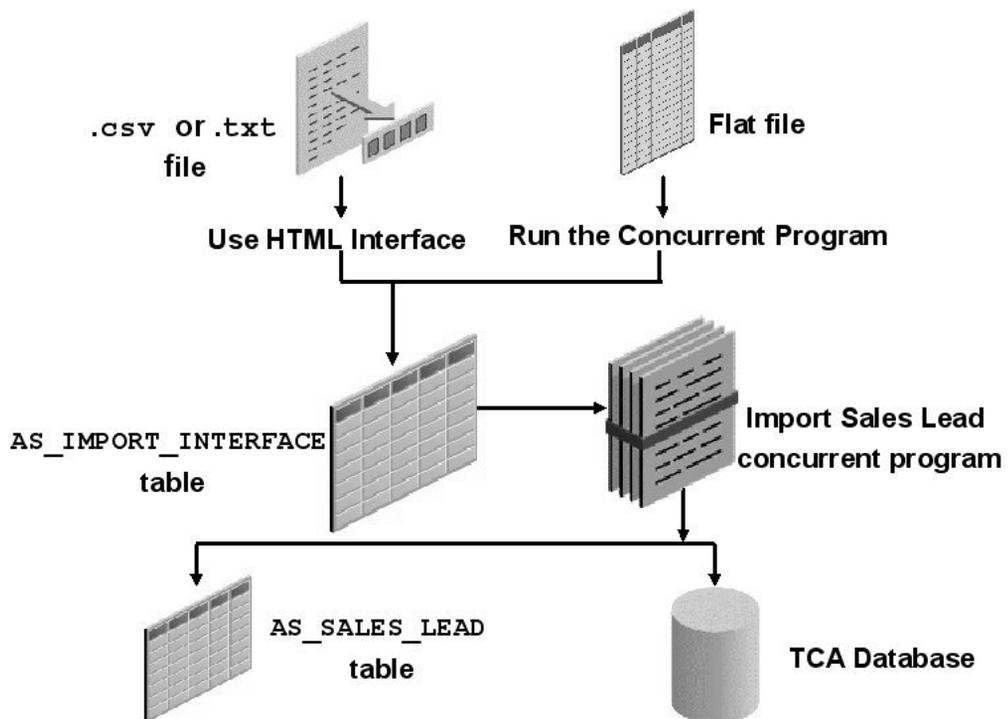
The Referral Management feature in Oracle Partner Management allows a partner to submit referrals to the vendor. After the vendor accepts the referrals, the referral

becomes a lead. For more information about Referral Management, see the *Oracle Partner Management Partner User Guide*.

3.2 The Lead Import Process

Importing leads is one of the sources to capture leads into Oracle Leads Management. [Figure 3-1, "The Lead Import Process"](#) depicts the methods of importing the leads and the processing that takes place after they are imported.

Figure 3-1 The Lead Import Process



You can import leads into Oracle Leads Management in two ways:

- Import data from a .csv or a .txt file, using the Lead Import utility in the HTML interface.

- Import data from a flat file by running the Lead Sales Table from Flat File concurrent program.

The imported records are stored in the `AS_IMPORT_INTERFACE` table. This table is an intermediary table that stages all lead records before they are refined and cleaned.

The Import Sales Lead concurrent program picks every lead from the `AS_IMPORT_INTERFACE` table, and runs it through Data Quality Management (DQM) to identify unique customer records, and through the Leads Deduplication rule to remove duplicate lead records.

All unique customer records identified by the DQM process are stored in the Trading Community Architecture (TCA) database. This database is a central repository that is accessible to all Oracle's E-Business Suite and ERP applications.

All unique lead records that are identified by the Leads Deduplication process are stored in the `AS_SALES_LEAD` table.

3.3 Importing Leads

Leads may be imported from a .csv, .txt, or from a flat file.

Topics in this section include:

- [Section 3.3.1, "Importing Leads from the HTML Interface"](#)
- [Section 3.3.2, "Evaluating Errors During Lead Import"](#)
- [Section 3.3.3, "Correcting Errors During Lead Import"](#)
- [Section 3.3.4, "Importing Leads From a Flat File"](#)
- [Section 3.3.5, "Imported Leads and Oracle Sales Tables"](#)

3.3.1 Importing Leads from the HTML Interface

You can import leads stored in a .csv or .txt file using the Import wizard from the HTML interface. Use the following procedure to import leads.

Prerequisites

A .csv or .txt file containing data for import is required.

Steps

1. Log in as an administrator, and navigate to Audience > Import.

2. In the Imports page, click **Create**.
The Import: Introduction page appears.
3. From the Data Type column, select Leads.
4. Click **Next**.
5. In the General region, enter the name and description of the import.
6. In the Source File section:
 - a. If the source file is at a client location, click **Go** next to the Client field to select the name and location of a source file from the local hard disk or network.
 - b. If the source file is at a server location, in the Server field, enter the URL for the source file.
 - c. If the source file is at a FTP location, click **Go** next to the FTP field, and enter the full path for the source file.

The file types supported are: a .zip file containing a .csv or a .txt file, a .csv file or a .txt file.
 - d. Use the Character Set drop-down list to select the character set that the source file uses.
 - e. Use the Column Delimiter list to select the delimiter used in the file to distinguish between two columns.

Choose tilde (~) unless you are using SQL Loader.
 - f. Use the Field Enclosed By list to select the character that encloses each field in the file.

This is required when the data in your file has special characters that must not be mistaken for the column delimiter.
 - g. Select the File Header Exists box, if the columns in the source file have a header.
7. Click **Next**.

The Source Fields are the columns in your import file. The Target Fields are the columns present in the table.
8. Select a Source field and a corresponding Target field.
9. Click ">".

The mapped fields appear in the Mapped Source Target fields section.

Ensure that all mandatory fields are mapped.

10. Click **Next**.

11. After reviewing the details, click **Import**.

Your import data is submitted for processing.

12. Click **Finish**.

You can track the status of the import from the Imports page.

Note: During the import process, the Assigned Date value (if present) for a lead is not imported. This date is assigned by the system.

3.3.2 Evaluating Errors During Lead Import

The Import Sales Lead concurrent program stores errors that occur during lead import in the AS_LEAD_IMPORT_ERRORS table.

Table 3–1 gives the status and descriptions for the lead import errors that you can see in the HTML interface.

Table 3–1 Status and Descriptions for Lead Import Errors in the HTML Interface

Status	Description
Complete	All records are complete/successful.
Incomplete - Errors Found	One or more records have errors.
Incomplete - Duplicates Found	One or more records are duplicates.
Incomplete - Duplicates and Errors Found	One or more duplicates AND one or more errors found.
Error	All records have errors.
Duplicate	All records are duplicates.

Note: Do not change the status of an imported lead from `Success` to `New`. A status of `Success` means that a lead has been successfully imported and cannot be imported again.

3.3.3 Correcting Errors During Lead Import

Use the following procedure to check for errors detected by the Import Sales Lead concurrent program during lead import, and correct them.

Prerequisites

You must have imported records into the AS_IMPORT_INTERFACE table.

Steps

1. Log in as an administrator, and navigate to Audience > Audience Workbench > Imports.
2. Click the name of the import which has an error.
3. From the Results region, click the Number of Error Records link.
The Import Error Detail page appears with details of errors for each record.
4. Click the Import Source Line ID link, and modify the column which has an error.
5. Click **Update**.
6. To reload the records, navigate back to the Import Details page, and click **Reload**.

Note: When you modify any of the details for a record, the original record is not updated. Instead, a new record is created and updated with the modifications.

3.3.4 Importing Leads From a Flat File

The flat file you are importing leads from must be a tilde-delimited file. The file name must have the extension .dat (for example, mynewleads.dat). Note the following points:

- To distinguish between imports, use different batch IDs.
- There are some mandatory columns in the AS_IMPORT_INTERFACE table. For such columns, a null or incorrect entry in the field results in a database error. You must provide valid values to these columns. Run SQL*Plus queries to obtain some of the values required by this table.

- If you have flexfields set up in your application, you must also populate the AS_IMP_SL_FLEX table. See [Section 3.3.4.4, "Flexfields"](#) for more information.

There are two ways to import leads into the AS_IMPORT_INTERFACE table from a flat file.

- [Section 3.3.4.1, "Importing Leads by Running the Concurrent Program"](#)
- [Section 3.3.4.2, "Importing Leads by Running SQL Loader Manually"](#)

3.3.4.1 Importing Leads by Running the Concurrent Program

Use the following details to run the Load Sales Lead Interface Table from Flat File concurrent program.

Note:

- You must have read and write permissions for the directory on the server.
 - The data in your import file must contain the required fields and the Load Status of each record must be NEW.
-
-

Prerequisite: FTP the tilde(~) delimited flat file with lead data to a directory on the server. The file must have the .dat extension.

Responsibility: Oracle Sales Administrator

Name of program: Load Sales Lead Interface Table from Flat File

Parameter:

P_DATAFILE - Name of the .dat file

Schedule - Once

For the steps to run the concurrent program, see [Section C.1, "Running Concurrent Programs"](#).

Note: The Load Sales Lead Interface Table from Flat File concurrent program supports only loading the AS_IMPORT_INTERFACE table. To take advantage of other interface tables, you must write your own program to populate them.

See Also

- For a sample data file, see [Section 3.3.4.3, "Sample Flat File"](#).
- For information on flexfields, see [Section 3.3.4.4, "Flexfields"](#).

3.3.4.2 Importing Leads by Running SQL Loader Manually

If you have access to Oracle SQL Loader, you can import details from the flat file without running the Load Sales Lead Interface Table from Flat File concurrent program. You must upload the flat file to a server, and create the SQL Loader file which will import the records from the flat file.

Use the following procedure to upload the file.

Prerequisites

- You must be familiar with Oracle SQL Loader as described in the *Oracle9i Database Utilities Guide*.
- You must be familiar with running SQL Plus database queries.

Steps

1. FTP the tilde(~) delimited flat file with lead data to a directory on the server. The file must have the .dat extension.

Note:

- The directory on the server must have read and write permissions.
 - The data in your import file must contain the required fields and the Load Status of each record must be NEW.
-
-

2. Create an Oracle SQL Loader parameter file. Here is what a sample SQL Loader parameter file will look like:

```
userid=<username>/<password>
control=ASTSLIMP.CTL
data=<path><lead import data file name>.dat
```

3. Upload the lead records using this parameter file as command line parameter to SQL Loader.

The records from the flat file are imported and processed by the Import Sales Lead concurrent program.

3.3.4.3 Sample Flat File

Below is a sample flat file for loading the AS_IMPORT_INTERFACE table. This example contains only one line of data.

```

~16-Sep-01~-1~16-Sep-01~-1~-1~LEAD_LOAD~16-Sep-01~NEW~ABC Corp~US~123
Xyzst.~Suite1008~~~RedwoodCity~94065~CA~~~7374~2000~CUSTOMER~MAR~15000~5000000~6
000000~DECISIONMAKER~1023472~1900~N~M~MR~~Hislast~Hisfirst~A~AccountsPayableSupe
rvisor~ARC~101~926~2667~GEN~650~123~926~2600~650~www.xyz.com~abc@xyz.com~Y~Y~N~N
~NEW~DECISION_
MAKER~DIRECT~10000~APPROVED~US~1~3MONTHS~~AAA~Lead1~EMAIL~159~424~425~357~204~EA
~100~50000~10588~10699~Leadcollectedon16
-SEP-01~~NEW~10001~OTN::990~10004~Y~Y~~~~~N~Y~Y~~~~10060~Y~1~Jan-00~~USERENTER
ED~XYZHQ~~~10~~~~~94065~1282~~~HQ~~~1008~~~~~XYZ~~~~~15-Aug-01~~~~~N~
1987SIC~~101~550000~~~N~MARKET~~10~~~Importantcontact~~Y~InformationTechnology~I
T~DECISION_MAKER~10588~N~Y~USER_ENTERED~1~1~~~MAILHTML~123~~~~~TSTENH

```

3.3.4.4 Flexfields

Use the AS_IMP_SL_FLEX table to store the flexfield values for all the following entities (tables). The entity names are seeded in AS_LOOKUPS, lookup_type = ENTITY_NAME.

- HZ_PARTIES
- HZ_LOCATIONS
- HZ_CONTACT_POINTS
- HZ_PARTY_SITES
- HZ_ORG_CONTACTS
- AS_SALES_LEADS
- AS_SALES_LEAD_LINES
- AS_SALES_LEAD_CONTACTS

The flexfields are imported along with the other data in the AS_IMPORT_INTERFACE table during the lead import process. To populate the data in the optional tables, use SQL*Loader or SQLPLUS.

The flexfields columns in HZ_ORG_CONTACT_ROLES, the global flexfields columns in HZ_PARTIES, HZ_LOCATIONS, HZ_CONTACT_POINTS, and HZ_ORG_CONTACTS

are obsoleted. Hence, the Import Sales Lead concurrent program does not support these columns.

Reference

For information about how to plan and set up flexfields, see the *Oracle Applications Flexfields Guide*.

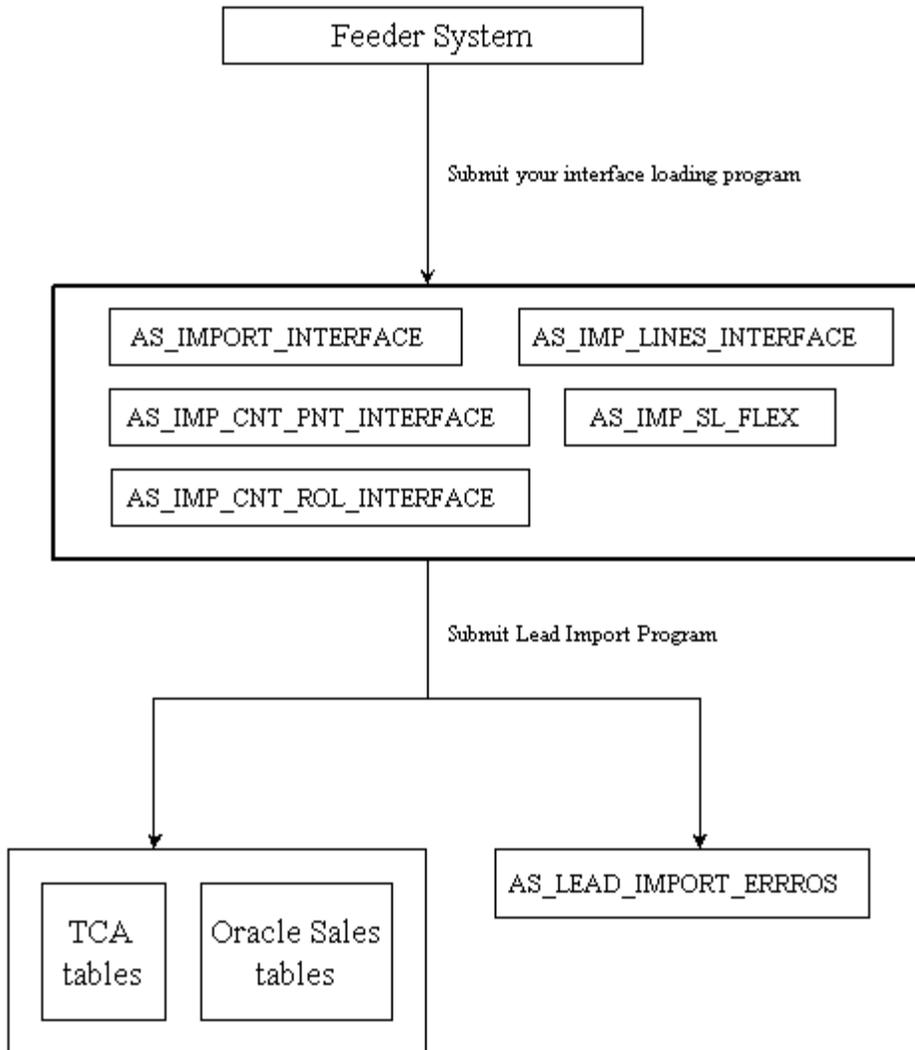
3.3.5 Imported Leads and Oracle Sales Tables

Figure 3–2 shows how lead information is imported into Oracle Sales tables.

From the feeder system, imported leads are stored in the interface tables. These are the AS_IMPORT_INTERFACE, AS_IMP_CNT_PNT_INTERFACE, AS_IMP_CNT_ROL_INTERFACE, AS_IMP_LINES_INTERFACE, and the AS_IMP_SL_FLEX tables.

After the Import Sales Lead concurrent program is run, appropriate records are created in the TCA database, Oracle Sales tables, and imported records that resulted in errors are stored in the AS_LEAD_IMPORT_ERRORS table.

Figure 3–2 Leads Data Imported into Oracle Sales Tables



3.4 The Import Sales Lead Concurrent Program

The Import Sales Lead concurrent program runs every record in the `AS_IMPORT_INTERFACE` table through the DQM and Leads deduplication processes. The TCA database is updated with any unique customer records. Unique lead records are stored in the `AS_SALES_LEADS` table. The concurrent program calls the leads processing engines to filter, qualify, rate and channel these leads to the sales teams.

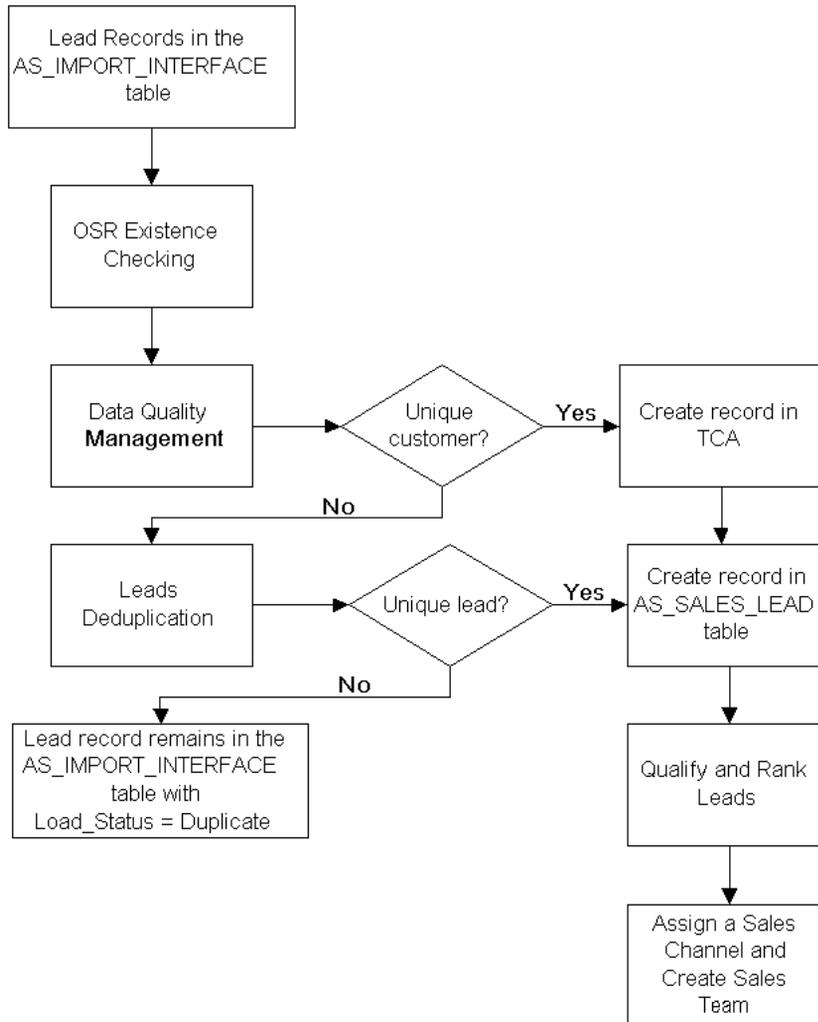
To set up, manage, and run the concurrent program, follow the procedures in these sections:

- [Section 3.4.1, "Import Sales Lead Concurrent Program Flow"](#)
- [Section 3.4.2, "Before Running the Concurrent Program"](#)
- [Section 3.4.3, "Running the Import Sales Lead Concurrent Program"](#)
- [Section 3.4.4, "Enhancing the Performance of the Import Sales Lead Concurrent Program"](#)
- [Section 3.4.5, "Limitation of the Import Sales Lead Concurrent Program"](#)

3.4.1 Import Sales Lead Concurrent Program Flow

[Figure 3–3](#) illustrates the manner and sequence in which the Import Sales Lead concurrent program processes records.

Figure 3–3 Import Sales Lead Concurrent Program Flow



The Import Sales Lead concurrent program does the following:

1. Checks for the existence of Original System Reference (OSR) using Leads Data Quality. See [Section 3.5.1.1, "Checking for Duplicate Original System Reference"](#).

2. Checks for the existence of customer, address, contact, and contact points using DQM Data Quality. See [Section 3.5.1, "Customer Data Quality"](#).
3. Creates a record in the TCA database, if the imported record is unique.
4. Checks for duplicate leads using Leads Data Quality. See [Section 3.5.4, "Leads Data Quality"](#).
5. Creates a lead in the `AS_SALES_LEAD` table, if the imported record is unique.
6. Qualifies and ranks the lead using the Leads Processing Engine. See [Chapter 5, "Processing Leads"](#).
7. Identifies the sales team, and assigns the lead to the owner of the sales team. See [Chapter 5, "Processing Leads"](#).
8. Creates a sales team to interact with the customer (the lead's organization), if required.

3.4.2 Before Running the Concurrent Program

The procedures in the following sections must be complete before running the Import Sales Lead concurrent program:

- [Section 3.4.2.1, "Setting Up DQM Match Rules"](#)
- [Section 3.4.2.2, "Setting Profiles Used by the Import Sales Lead Concurrent Program"](#)
- [Section 3.4.2.3, "Creating Valid Lookup Codes"](#)
- [Section 3.4.2.4, "Setting up Territories"](#)
- [Section 3.4.2.5, "Loading the Import Interface Tables"](#)

3.4.2.1 Setting Up DQM Match Rules

All customer records are stored in the TCA database. The purpose of the DQM program is to perform existence checking for a customer in the TCA database. If a customer in the import record already exists in TCA, then DQM returns the `party_id` for the same. If not, the Import Sales Lead concurrent program creates a new customer record.

DQM checks the following attributes to identify unique records - Customers, Addresses, Contacts, and Contact Points.

DQM uses the matching rules to decide if a customer record exists in TCA. You can create a rule based on the business requirements in your organization.

The DQM match rules that will be used are dependent on the following profiles:

- OS: Use DQM Rule code to match Party
- OS: Use DQM Rule code to match Person
- OS: Use DQM Rule code to match Contact

Use the following procedure to create a sample DQM rule to find duplicate contact records. The procedure is based on the sample rule explained in [Section 3.5.3.3, "Identify Duplicate Contacts"](#).

Prerequisites

None

Steps

1. Log in to Oracle Forms with the Trading Community Manager responsibility.
2. Navigate to Data Quality Management > Setup > Match Rules.
3. In the Rule Name and Description fields, enter the necessary details.
4. Select **Identify Duplicates** in the Purpose area.
5. In the Acquisition tab:
 - a. In the Attribute Name column, enter Name.
The Entity column displays Party.
 - b. In the Attribute Name column, enter Contact Name.
 - c. The Entity column displays Contacts, and the Type column displays Custom Attribute.
 - d. In the Attribute Name column, enter Phone Number Flexible Format.
 - e. The Entity column displays Contact_Points.
 - f. In the Attribute Match area, select **Match All**.
 - g. Click **Save**.
6. In the Transformation area, choose each attribute, and select a transformation.
 - a. For the Name attribute, select WR Names + Cleanse.
 - b. For the Contact Name attribute, select WR Person.
 - c. For the Phone Number Flexible Format attribute, select Exact.

7. In the Scoring tab
 - a. In the Match Threshold field, enter 110.
 - b. In the Attribute Name column, enter Contact Name.
 - c. The Entity column displays Contacts, and the Type column displays Custom Attribute.
 - d. In the Score column, enter 100.
 - e. In the Attribute Name column, enter an e-Mail Address.
 - f. The Entity column displays Contact_Points.
 - g. In the Score column, enter 10.
 - h. In the Attribute Name column, enter Phone Number Flexible Format.
 - i. The Entity column displays Contact_Points.
 - j. In the Score column, enter 10.
 - k. In the Attribute Name column, enter a URL.
 - l. The Entity column displays Contact_Points.
 - m. In the Score column, enter 10.
8. In the Transformation area, choose each attribute, and select a transformation.
 - a. For the Contact Name attribute:
 - Select Exact String, and enter 100 for Weight (%).
 - Select WR Person + Cleanse, and enter 90 for Weight (%).
 - b. For the e-Mail Address attribute, select Exact (E-mail), and enter 100 for Weight (%).
 - c. For the Phone Number Flexible Format attribute, select Exact, and enter 100 for Weight (%).
 - d. For the URL attribute, select Cleanse (URL), and enter 100 for Weight (%).
9. Click **Compile**.

The rule is compiled and is available for use.

See Also

For other sample matching rules, see [Section 3.5.3, "Designing Matching Rules to Detect Duplicate Customer or Person"](#).

3.4.2.2 Setting Profiles Used by the Import Sales Lead Concurrent Program

Because the Import Sales Lead concurrent program triggers other programs as part of its flow, the following profiles must be set before it is run.

- OS: Use DQM Rule code to match Party
- OS: Use DQM Rule code to match Person
- OS: Use DQM Rule code to match Contact
- OS: Default Resource ID Used for Sales Lead Assignment

Use the following procedure to set values for the profiles.

Prerequisites

Create DQM matching rules.

Steps

1. Log in to Oracle Forms with the System Administrator responsibility.
2. Navigate to Profile > System > Open.
3. In the Profile field, enter the full or partial name of the profile, and click **Find**.
Use the % wild card, if required.
4. At the Site level:
 - a. For the OS: Use DQM Rule code to match Contact profile, associate it with rules that find matching records based on the Contact in the imported record.
 - b. For the OS: Use DQM Rule code to match Party profile, associate it with rules that find matching records based on the Party ID in the imported record.
 - c. For the OS: Use DQM Rule code to match Person profile, associate it with rules that find matching records based on the Person in the imported record.
 - d. For the OS: Default Resource ID Used for Sales Lead Assignment, set it to the resource who will handle any leads that are not assigned to any current territory.

3.4.2.3 Creating Valid Lookup Codes

Lookup codes map to drop-down lists in the User Interface. The `SOURCE_SYSTEM` lookup type identifies the source of the leads. For example, lead sources could be from a marketing campaign or a partner referral.

The `SOURCE_SYSTEM` lookup type categorizes the leads in the system, and helps you to track them. Seeded values in the `SOURCE_SYSTEM` lookup type are Interaction, Marketing, New, Referral, Sales_Campaign, Store, and User.

Use the following procedure to create additional codes for the `SOURCE_SYSTEM` lookup type.

Prerequisites

None

Steps

1. Log in to Oracle Forms with the Oracle Sales Administrator responsibility.
2. Navigate to Oracle Sales Setup > Lookup Codes > Sales.
3. Select **View > Query By Example > Enter**.
4. In the Oracle Sales Lookups form, enter `SOURCE_SYSTEM` in the Type field.
The seeded values - Interaction, Marketing, New, Referral, Sales_Campaign, Store, and User - appear.
5. In the Code field, enter the new code value for the lookup.
6. In the Meaning field, enter a meaning for the lookup code.
The meaning is displayed as one of the values in the drop-down list. For example, the Meaning 'Yes' is displayed for Code Y. The code is stored in a hidden field.
7. In the Description field, give a description for the lookup code.
The description along with the meaning gives more information about your lookup code.
8. In the Tag field, enter a tag to describe your lookup code.
The tag can be used to categorize lookup values. This field is optional.
9. In the From and To fields, enter the dates between which this lookup code will be active.

If you do not enter a start date, the lookup code is valid immediately.

10. In the Enable field, indicate whether applications can use your lookup code.

11. Click **Save**.

The new code value for the `SOURCE_SYSTEM` lookup type is saved.

3.4.2.4 Setting up Territories

A territory refers to the geographical location of a lead and a sales team. Setting up a territory is important so that the lead is assigned to the right sales team in the correct geographical location.

Create territories in the Oracle Sales and TeleSales node on the territory setup form of Territory Manager. Territory Manager is part of the CRM Foundation module.

Generate Territory Packages Concurrent Program

Run the Generate Territory Packages concurrent program. This concurrent program builds the API that returns the winning territories which are defined in territory setup. Run the program at least once before you import leads and every time after the territory setup is modified. You need not run this program every time you import leads.

Reference

Oracle Territory Management Implementation Guide.

3.4.2.5 Loading the Import Interface Tables

Load the `AS_IMPORT_INTERFACE` table before running the Import Sales Lead concurrent program

AS_IMPORT_INTERFACE (mandatory): This interface table holds sales leads, customers, addresses, and contacts information to be imported. This table also holds space to import five lead lines in one record.

The following are auxiliary tables. Load data into these tables using a custom program.

- **AS_IMP_LINES_INTERFACE (optional):** This interface table can be used to hold lead lines information, in case you have more than five line items for a lead.
- **AS_IMP_CNT_ROL_INTERFACE (optional):** This interface table is used to hold contact roles information to be imported.

- **AS_IMP_CNT_PNT_INTERFACE (optional):** This interface table is to hold any extra contact points information to be imported apart from the AS_IMPORT_INTERFACE table.
- **AS_IMP_SL_FLEX (optional):** This interface table is to store the flexfields values.

3.4.3 Running the Import Sales Lead Concurrent Program

The Import Sales Lead concurrent program must be scheduled to run at particular intervals. As a result of the concurrent program, unique leads are stored in the AS_SALES_LEAD table, and if any of these lead records are unique to the TCA database, they are added to it.

Use the following details to run the Import Sales Lead concurrent program.

Prerequisite: Complete all tasks covered in [Section 3.4.2, "Before Running the Concurrent Program"](#).

Responsibility: Oracle Sales Administrator

Parameters:

- Lead Source System - NEW
- Show Debug Message - N
- Batch Id - Batch number if you have imported leads in batches

Schedule: Periodically

For the steps to run the concurrent program, see [Section C.1, "Running Concurrent Programs"](#).

3.4.3.1 Import Sales Lead Concurrent Program Parameters

[Table 3–2](#) lists the parameters for the Import Sales Lead concurrent program.

Table 3–2 Import Sales Lead Concurrent Program Parameters

Parameter	Req?	Lookup	Remarks
Lead Source System	Y	SOURCE_SYSTEM	Used to identify leads generated from different business entities. Only the records that match the parameter value are selected for processing. This is case-sensitive.

Table 3–2 Import Sales Lead Concurrent Program Parameters

Parameter	Req?	Lookup	Remarks
Debug message?	N	Y or N	Default is N. If set to Y, the debug messages can be seen by clicking View Log in the Concurrent Request screen.
BatchID	N	-	Used to process a small set of data. This is particularly useful when leads are imported in batches. The Batch ID may be used to process only a particular set of data in a batch.
Purge error message?	N	Y or N	Default is N. If set to Y, all records in the AS_LEAD_IMPORT_ERRORS table are deleted.

Note: The Import Sales Lead concurrent program validates currency codes from the FND_CURRENCIES table while on the HTML UI, the currency codes are picked from the AS_LOOKUP table (lookup type = REPORTING_CURRENCY). The currency codes in both the places must be synchronized to import a lead successfully. If the currency code is not found in the FND_CURRENCIES table, then the currency value set in the JTF_Profile_Default_Currency profile is used.

3.4.4 Enhancing the Performance of the Import Sales Lead Concurrent Program

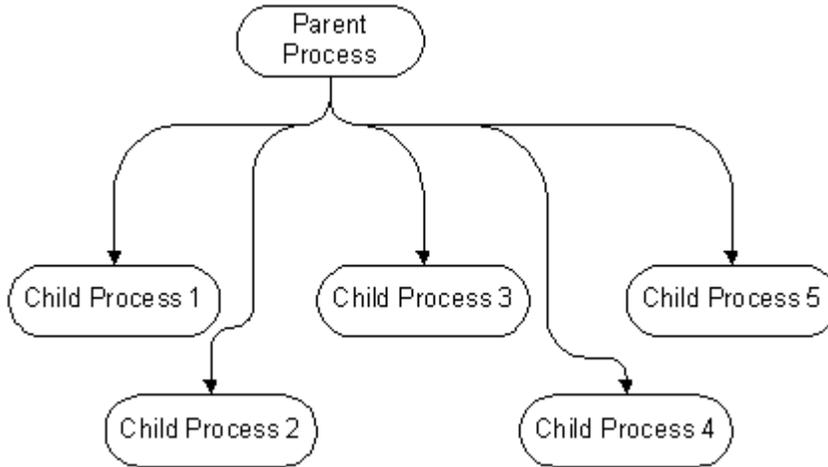
The Import Sales Lead concurrent program processes a number of records in a batch which may be time consuming. In release 11.5.10, the performance of the concurrent program has been improved by running multiple concurrent programs in parallel - each processing fewer number of records.

Parallel Lead Import

In order to improve the performance, the Import Sales Lead concurrent program, itself being the parent, spawns multiple child requests to process the imported records in the AS_IMPORT_INTERFACE table. Since the child requests run in parallel, significant performance improvement is achieved. The concurrent program raises the Lead Import - Pre event before spawning child processes and then raises Lead Import - Post event after all the child processes are complete. For more information on the Pre and Post events, see [Section 3.6.1, "Seeded Business Events"](#).

The Import Sales Lead concurrent program splits into a parent process and multiple child processes. [Figure 3–4](#) illustrates the relationship between the parent and child processes.

Figure 3–4 Parent and Child Processes



The OS:Minimum Number of Records for Parallel Processing in Lead Import profile governs the number of child processes that are spawned. For each n number of records in the AS_IMPORT_INTERFACE table, a new request is placed. The default value for this profile is 400.

3.4.5 Limitation of the Import Sales Lead Concurrent Program

The Import Sales Lead concurrent program checks the database for duplicates using the DQM logic of customer, address, contact, and contact point before creating new records. However, the DQM logic has one limitation.

While importing leads, if the concurrent program creates new records such as Party, Contact, Party Site and Contact Points, the new entries are not reflected in the DQM staging schema. Therefore, if the same set of leads is imported again without any changes, the DQM logic will fail causing the Lead deduplication program to fail as well. To overcome this, the DQM Synchronization concurrent program must be run after the first import and before the next import.

However, lead deduplication can still fail if there are duplicate leads in a single set of imported records. See [Section 3.6.2, "Custom User Hook"](#) to avoid this.

Running the DQM Synchronization Program

Use the following details to run the DQM Synchronization Program concurrent program.

Prerequisite: None

Responsibility: Trading Community Manager

Schedule: Once

For the steps to run the concurrent program, see [Section C.1, "Running Concurrent Programs"](#).

3.5 Data Quality

An information system is only as good as the data which resides within it. In Oracle Leads Management, any lead records that are imported go through rigorous screening and filtering. The records are checked for:

- Customer Data Quality - performed by Data Quality Management (DQM).
- Leads Data Quality - performed by the Leads Deduplication rule.

[Table 3–3](#) gives the sequence in which the Customer and Leads data quality checks are performed.

Table 3–3 *Checking for Data Quality in imported Lead Records*

Task Performed	Program
Checking for Duplicate Original System Reference	Import Sales Lead concurrent program
Checking for Duplicate Customers	Import Sales Lead concurrent program using DQM
Checking for Duplicate Addresses	Import Sales Lead concurrent program using DQM
Checking for Contacts and Contact Points	Import Sales Lead concurrent program using DQM
Leads Data Quality	Import Sales Lead concurrent program using the Lead Deduplication Rule

Topics in this section include:

- [Section 3.5.1, "Customer Data Quality"](#)
- [Section 3.5.2, "Setting Up DQM Staging Schema"](#)
- [Section 3.5.3, "Designing Matching Rules to Detect Duplicate Customer or Person"](#)
- [Section 3.5.4, "Leads Data Quality"](#)

3.5.1 Customer Data Quality

The Import Sales Lead concurrent program uses the rule-based DQM tool to identify existence of customer records in the TCA database. It uses customer entities like parties (both organization and person), party site, contacts and contact points information to match a record.

When a lead record is imported, it is important to find if a record for this customer already exists in your database. The DQM program matches the imported record with the records in the TCA database to find a matching customer record. If a match is not found, a customer record is created by the Import Sales Lead concurrent program in the TCA registry database.

The DQM program uses rules to identify a matching record. The rules that are used are dependent upon the profiles that are set. See [Section 3.4.2.1, "Setting Up DQM Match Rules"](#) and [Section 3.4.2.2, "Setting Profiles Used by the Import Sales Lead Concurrent Program"](#).

For more information on DQM, see the *Oracle Trading Community Architecture Data Quality Management User Guide*.

Sections in this topic include:

- [Section 3.5.1.1, "Checking for Duplicate Original System Reference"](#)
- [Section 3.5.1.2, "Checking for Duplicate Customers"](#)
- [Section 3.5.1.3, "Checking for Duplicate Addresses"](#)
- [Section 3.5.1.4, "Checking for Contacts and Contact Points"](#)

3.5.1.1 Checking for Duplicate Original System Reference

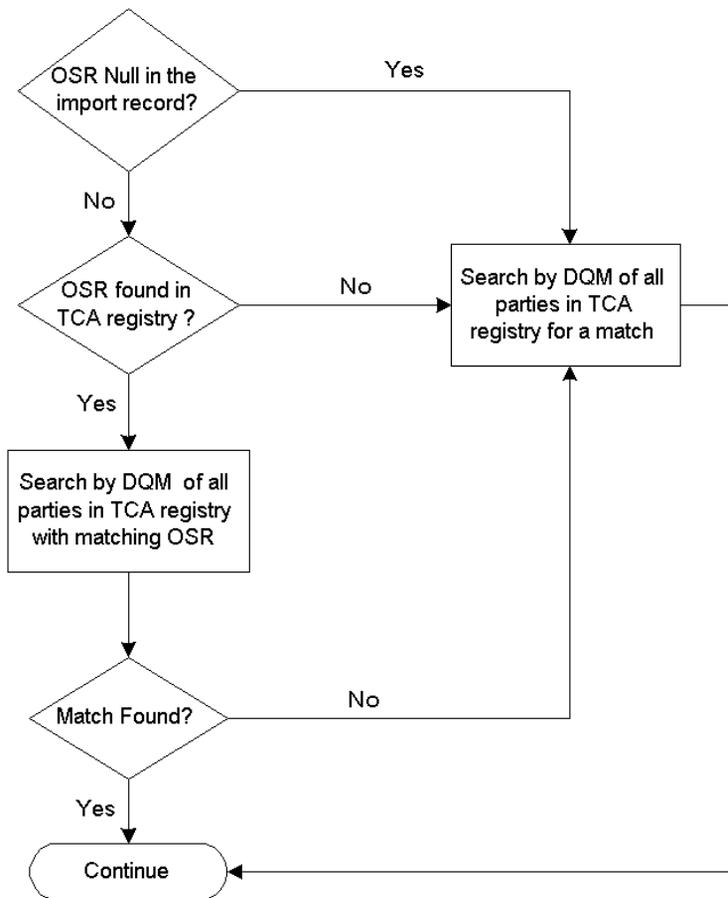
When leads are imported from a list generated by a third-party data source, each entry carries the ID of the party record in that third party database. This ID is referred to as Original System Reference (OSR).

The OSR is carried over to the party record in TCA thus maintaining a correlation between the TCA Party ID recorded on the lead and the party record in the external data source. If the existing party in TCA has a different OSR, the record is updated with the latest one.

OSR is not included in the list of attributes checked by DQM. Therefore, the OSR existence checking is done by the Import Sales Lead concurrent program itself.

[Figure 3-5](#) gives the logic followed to check for duplicate OSR before DQM starts matching party and address.

Figure 3-5 OSR Logic Used by Import Sales Lead Concurrent Program



If the OSR is available in the import record, then the Import Sales Lead concurrent program uses it to find a set of parties with the same OSR in the TCA database. If a matching set is found, the DQM matching rules are applied on this set alone to find a matching party.

If OSR is available and no matching party set is found or if the OSR itself is not available in the import record, then the DQM matching rules are applied to all the records in the TCA database to find a matching party.

If a matching party is found by using the DQM matching rules, then the matched party_id is reused. If a matching party is not found, the Import Sales Lead concurrent program creates a new party.

Note: It is recommended that you pass OSR in an import record, if known. This substantially improves the performance of the Import Sales Lead concurrent program.

3.5.1.2 Checking for Duplicate Customers

After the Import Sales Lead concurrent program checks for the existence of OSR, the DQM program starts checking for a matching customer in the TCA database. Depending on whether the import record is an organization or a person, the matching rule created to identify duplicate Party or Person is used.

The Import Sales Lead concurrent program calls the `HZ_PARTY_SEARCH.FIND_PARTIES` API to run the rules that find duplicate customers. The Organization name (Party) or first name and last name (Person) is passed in along with the address-related information in the party site record. If the lead is created for Organization, the contact information is also passed in to find a better match on the party. If the lead is created for Person, the contact information is not passed in.

The API call returns the context ID and the number of matches found. The parties are returned and populated in the `HZ_MATCHED_PARTIES_GT` table sorted on score. If the number of matches found is greater than zero, the context ID is used to get the match details. The highest score will have the best match. If multiple parties with the same high scores are found, the party that was created last is picked up.

3.5.1.3 Checking for Duplicate Addresses

When a matching party is identified, the addresses between the import record and matching party are compared to see if they match as well.

The `get_matching_party_sites` API is called to check if the matching address exists. The party sites are returned and populated in the `HZ_MATCHED_PARTY_SITES_GT` table. If a match is found, the existing location ID and party site ID are used. If no match is found, the location and party site in the imported record are used.

3.5.1.4 Checking for Contacts and Contact Points

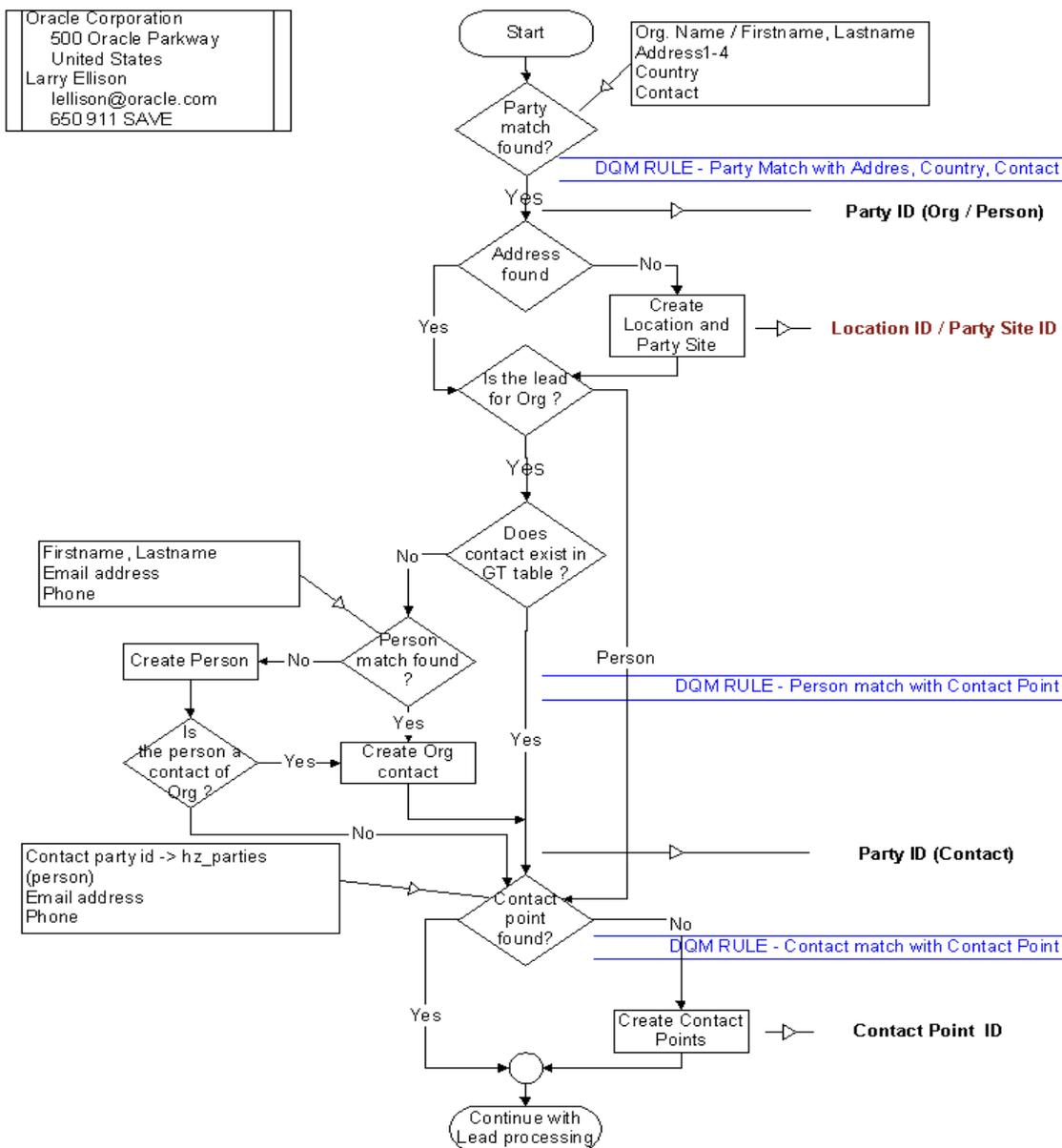
If the lead is created for Person and an existing party is found, the contact information is checked. The matching rule created to identify duplicate contacts is used. The party ID of the person along with the contact points are passed while calling the `get_matching_contact_points` API. The matched contact points are found in the `HZ_MATCHED_CPTS_GT` table.

If the lead is created for an Organization and there is no contact Person found for the Organization, a search is performed to find a matching contact Person existing in the database. In order to find a match, the contact person's first name, last name and contact points are passed in while calling the `HZ_PARTY_SEARCH.FIND_PARTIES` API.

To find a duplicate contact person, you must create a new rule which takes the details of the contact person name and the contact points.

[Figure 3–6](#) depicts how the DQM program checks for duplicate entities in the database.

Figure 3–6 How DQM Checks for Duplicate Entities



3.5.2 Setting Up DQM Staging Schema

Use the DQM Staging Program to create the staged schema and *interMedia* indexes. This program applies transformation functions to a portion of the data contained in the TCA registry and generates a separate schema with the transformed and standardized data. The time that the program takes to create the staged schema depends on the size of your database and the number of attributes and transformation functions that you defined.

Reference

Oracle Trading Community Architecture Data Quality Management User Guide

Use the following details to run the DQM Staging program concurrent program.

Prerequisites: Define attributes and transformation functions in DQM. For sample transformation functions, see [Section 3.5.3, "Designing Matching Rules to Detect Duplicate Customer or Person"](#).

Responsibility: Trading Community Manager

Parameters:

- Number of Parallel Staging Workers - 1
- Staging Command - STAGE_ALL_DATA
- Continue Previous Execution - No

For the steps to run the concurrent program, see [Section C.1, "Running Concurrent Programs"](#).

Note: Run the DQM Staging Program every time you add or modify the attributes or the transformation functions.

3.5.3 Designing Matching Rules to Detect Duplicate Customer or Person

You need three DQM match rules for customer or person existence checking during the lead import process. Matches are initially identified using the Acquisition Attributes, and a score is assigned to each match based on the scoring attributes. The party with the highest score is matched with the import lead record.

Use the following as samples while designing DQM matching rules for the Import Sales Lead concurrent program. Matching rules can also be tailored according to specific requirements using other seeded and custom attributes. For more detailed

information, see the *Oracle Trading Community Architecture Data Quality Management User Guide*.

Sections in this topic include:

- [Section 3.5.3.1, "Identify Duplicate Parties"](#)
- [Section 3.5.3.2, "Identify Duplicate Persons"](#)
- [Section 3.5.3.3, "Identify Duplicate Contacts"](#)

See Also

For steps to create the sample rules, see [Section 3.4.2.1, "Setting Up DQM Match Rules"](#).

3.5.3.1 Identify Duplicate Parties

The first match rule is used to identify the organization party and address. It is assigned to the profile OS:Use DQM Rule Code to Match Party.

Use the following sample rule to identify existence of party (Organization) and addresses for the same. Because this rule is also used to identify the existence of addresses, it must contain PARTY_SITES entity attributes as well as PARTY entity attributes.

[Figure 3–7](#) depicts a screen shot of the sample rule with the acquisition attributes for Party existence checking.

Figure 3–7 Acquisition Attributes for Party Existence Checking

The screenshot shows the Oracle Match Rules configuration window for a rule named 'LEAD_DUP_ORG'. The status is 'COMPILED'. The description is 'Finds duplicates based on Party Name & A'. The purpose is set to 'Identify Duplicates'. The 'Acquisition' tab is active, showing a table of attributes and a transformation.

Attribute Name	Entity	Type	Filter
Party Type	PARTY	Lookup	<input type="checkbox"/>
Name	PARTY		<input type="checkbox"/>
State	PARTY_SITES		<input type="checkbox"/>
Country	PARTY_SITES	Lookup	<input type="checkbox"/>
Contact Name	CONTACTS	Custom Attr	<input type="checkbox"/>

Attribute Match options:

- Match Any Attribute
- Match All Attributes

Transformations:

Transformation Name	Description
EXACT	Removes non-alphanumeric characters; fi

Figure 3–8 depicts a screen shot of the sample rule with the scoring attributes for Party existence checking.

Figure 3–8 Scoring Attributes for Party Existence Checking

Match Rules

Rule Name: **LEAD_DUP_ORG** Status: **COMPILED** [Compile]

Description: **Finds duplicates based on Party Name & A** [Unlock]

Purpose: Search Identify Duplicates [Copy]

Acquisition Scoring

Match Threshold: **120** Override Threshold: [] Automatic Merge Threshold: []

Attribute Name	Entity	Type	Score
Name	PARTY		100
Address	PARTY_SITES	Custom Attr	20
Postal Code	PARTY_SITES		20
State	PARTY_SITES		20
Contact Name	CONTACTS	Custom Attr	40

Transformations

Transformation Name	Description	Weight (%)	Exact	Similarity (%)
EXACT STRING	Captures exact string; removes non-alph	100	<input checked="" type="radio"/>	<input type="radio"/>
WR NAMES + CLEANSE	Person and org. name word replacemen	90	<input checked="" type="radio"/>	<input type="radio"/>

Match Rule Name: LEAD_DUP_ORG

Description: Finds identical organization parties based on Party Name and Address information.

Purpose: Identify duplicate organizations and addresses.

Table 3–4, Table 3–5, and Table 3–6 list the attributes for this rule.

Table 3–4 Existence Checking for Party: Acquisition Attributes

Acquisition Attributes	Entity	Transformation Function	Description	Type
Party Type	PARTY	EXACT	Catches format errors	Lookup
Party Name	PARTY	WR NAMES + CLEANSE	Captures the exact string, removes non-alphanumeric characters, forces upper case, removes vowels, and double letters	-

Table 3–4 Existence Checking for Party: Acquisition Attributes

Acquisition Attributes	Entity	Transformation Function	Description	Type
State	PARTY_SITES	WR STATE	Word replacement	-
Country	PARTY_SITES	EXACT	Captures the exact string, removes non-alphanumeric characters, forces upper case, and catches format errors.	Lookup
CONTACT NAME	CONTACTS	WR+CLEANSE+ REVERSE	Word replacement of Person and Organization names, removes vowels and double letters, reorders first word to the back.	Custom Attribute

Table 3–5 Existence Checking for Party: Matching Attributes

Attribute Match	Match Threshold	Override Threshold	Automatic Merge Threshold
Match Any	120	<null>	<null>

Table 3–6 Existence Checking for Party: Scoring Attributes

Scoring Attribute	Entity	Score	Transformation Function	Description	Type	Weight (%)
Party Name	PARTY	100	EXACT STRING	Captures the exact string, removes non-alphanumeric characters, and forces upper case.	-	100
-	-	-	WR CLEANSE & REVERSE	Word replacement of Person and Organization names, removes vowels and double letters, reorders first word to the back.	-	90

Table 3–6 Existence Checking for Party: Scoring Attributes

Scoring Attribute	Entity	Score	Transformation Function	Description	Type	Weight (%)
Address	PARTY_SITES	20	EXACT	Removes non-alphanumeric characters and forces upper case.	Custom Attribute	100
-	-	-	WR ADDRESS + CLEANSE	Address with word replacement, and removes vowels and double letters.	-	90
Postal Code	PARTY_SITES	20	EXACT	Removes non-alphanumeric characters and forces upper case.	-	100
State	PARTY_SITES	20	WR STATE	State word replacements		100
Contact Name	CONTACTS	40	EXACT STRING	Captures the exact string, removes non-alphanumeric characters, and forces upper case.	Custom Attribute	100
-	-	-	WR CLEANSE + REVERSE	Word replacement of Person and Organization names, removes vowels and double letters, reorders first word to the back.	-	90

Apart from the specified attributes in the above sample matching rule, more party (organization) related attributes like DUNS Number, Tax Reference, and SIC Code can be specified as per custom requirements.

3.5.3.2 Identify Duplicate Persons

The second match rule is used to identify the person and address. It is assigned to the profile OS:Use DQM Rule Code to Match Person.

Use the following sample rule to identify existence of party (Person) and addresses. Because Party Type is an Acquisition attribute, the above sample matching rule can also be used for Person existence checking. You may add more Person-related attributes to the matching rule as per custom requirements. The PARTY_SITES entity attributes must be specified because the same rule is used to identify existence of addresses for a specified person.

Figure 3–9 depicts a screen shot of the sample rule with the acquisition attributes for Person existence checking.

Figure 3–9 Acquisition Attributes for Person Existence Checking

The screenshot shows the 'Match Rules' configuration window for a rule named 'LEAD_DUP_PERSON'. The status is 'COMPILED'. The description is 'Finds duplicates based on Person Name &'. The purpose is set to 'Identify Duplicates'. The 'Acquisition' tab is active, showing a table of attributes and a matching rule configuration.

Attribute Name	Entity	Type	Filter
Name	PARTY		<input type="checkbox"/>
State	PARTY_SITES		<input type="checkbox"/>
Country	PARTY_SITES	Lookup	<input type="checkbox"/>
			<input type="checkbox"/>
			<input type="checkbox"/>

Attribute Match configuration:

- Match Any Attribute
- Match All Attributes

Transformations:

Transformation Name	Description
WR NAMES + CLEANSE	Person and org. name word replacement

Figure 3–10 depicts a screen shot of the sample rule with the scoring attributes for Person existence checking.

Figure 3–10 Scoring Attributes for Person Existence Checking

Match Rules

Rule Name: **LEAD_DUP_PERSON** Status: **COMPILED**

Description: **Finds duplicates based on Person Name &**

Purpose: Search Identify Duplicates

Acquisition **Scoring**

Match Threshold: **130** Override Threshold: Automatic Merge Threshold:

Attribute Name	Entity	Type	Score
Name	PARTY		100
Address	PARTY_SITES	Custom Attr	30
Postal Code	PARTY_SITES		10
State	PARTY_SITES		10

Transformations

Transformation Name	Description	Weight (%)	Similarity		Similarity (%)
			Exact		
EXACT STRING	Captures exact string; removes non-alph	100	<input checked="" type="radio"/>	<input type="radio"/>	<input type="text"/>
WR NAMES + CLEANSE	Person and org. name word replacemen	90	<input checked="" type="radio"/>	<input type="radio"/>	<input type="text"/>

Match Rule Name: LEAD_DUP_PERSON

Description: Finds duplicate persons based on Person Name & Address Information

Purpose: To identify duplicate persons and addresses

Table 3–7, Table 3–8, and Table 3–9 list the attributes for this rule.

Table 3–7 Existence Checking for Person: Acquisition Attributes

Acquisition Attributes	Entity	Transformation Function	Description	Type
Party Name	PARTY	WORD REPLACE + CLEANSE + REVERSE	Word replacement of Person and Organization names, removes vowels and double letters, reorders first word to the back.	-

Table 3–7 Existence Checking for Person: Acquisition Attributes

Acquisition Attributes	Entity	Transformation Function	Description	Type
State	PARTY_SITES	WORD REPLACE + EXACT	Word replacement of Person and Organization names, removes non-alphanumeric characters, and forces upper case.	-
Country	PARTY_SITES	EXACT	Removes non-alphanumeric characters, and forces upper case.	Lookup

Table 3–8 Existence Checking for Person: Matching Attributes

Attribute Match	Match Threshold	Override Threshold	Automatic Merge Threshold
Match All	130	<null>	<null>

Table 3–9 Existence Checking for Person: Scoring Attributes

Scoring Attribute	Entity	Score	Transformation Function	Description	Type	Weight (%)
Party Name	PARTY	100	EXACT_STRING	Captures the exact string, removes non-alphanumeric characters, and forces upper case.	-	100
-	-	-	WR CLEANSE & REVERSE	Word replacement of Person and Organization names, removes vowels and double letters, reorders first word to the back.	-	90
Address	PARTY_SITES	30	EXACT	Removes non-alphanumeric characters, and forces upper case.	Custom Attribute	100

Table 3–9 Existence Checking for Person: Scoring Attributes

Scoring Attribute	Entity	Score	Transformation Function	Description	Type	Weight (%)
-	-	-	WR ADDRESS + CLEANSE	Word replacement of Person and Organization names, removes vowels and double letters.	-	90
Postal Code	PARTY_SITES	10	EXACT	Removes non-alphanumeric characters, and forces upper case.	-	100
State	PARTY_SITES	10	WORD REPLACE + CLEANSE	Word replacement of State name, removes vowels and double letters.	-	100

3.5.3.3 Identify Duplicate Contacts

The third match rule is used to identify a contact of the organization with relationship type of *Contact of* only. No other relationships types (such as Employee Of, Consumer Of) are considered. This rule is assigned to the profile OS:Use DQM Rule Code To Match Contact.

Use the following sample matching rule to identify duplicate Contacts and Contact Points like e-mail, phone number, and URL. Because the same rule is used to identify Contact Points, specify the CONTACT_POINTS entity attributes while designing the matching rule for the identification of contacts.

Figure 3–11 depicts a screen shot of the sample rule with the acquisition attributes for Contact existence checking.

Figure 3–11 Acquisition Attributes for Contact Existence Checking

The screenshot shows the 'Match Rules' configuration window for a rule named 'LEAD_DUP_CONTACT'. The status is 'COMPILED'. The description is 'Finds duplicates based on Contact Name &'. The purpose is set to 'Identify Duplicates'. The 'Acquisition' tab is active, showing a table of attributes and a list of transformations.

Attribute Name	Entity	Type	Filter
Name	PARTY		<input type="checkbox"/>
Contact Name	CONTACTS	Custom Attr	<input type="checkbox"/>
Phone Number Flexible Forma	CONTACT_POINTS	Custom Attr	<input type="checkbox"/>
			<input type="checkbox"/>
			<input type="checkbox"/>

Attribute Match

Match Any Attribute

Match All Attributes

Transformations

Transformation Name	Description
WR NAMES + CLEANSE	Person and org. name word replacement

Figure 3–12 depicts a screen shot of the sample rule with the scoring attributes for Contact existence checking.

Figure 3–12 Scoring Attributes for Contact Existence Checking

Match Rules

Rule Name: **LEAD_DUP_CONTACT** Status: **COMPILED**

Description: **Finds duplicates based on Contact Name &**

Purpose: Search Identify Duplicates

Acquisition **Scoring**

Match Threshold: **110** Override Threshold: Automatic Merge Threshold:

Attribute Name	Entity	Type	Score
Contact Name	CONTACTS	Custom Attr	100
e-mail Address	CONTACT_POINTS		10
Phone Number Flexible Format	CONTACT_POINTS	Custom Attr	10
URL	CONTACT_POINTS		10

— Transformations —

Transformation Name	Description	Weight (%)	Exact	Similarity (%)
EXACT STRING	Captures exact string; removes non-alpha	100	<input checked="" type="radio"/>	<input type="text"/>
WR PERSON + CLEANSE	Person name word replacement + Remo	90	<input checked="" type="radio"/>	<input type="text"/>

Match Rule Name: LEAD_DUP_CONTACT

Description: Finds identical contacts based on Contact Name and Contact Points.

Purpose: To identify duplicate contact and contact points

Table 3–10, Table 3–11, and Table 3–12 list the attributes for this rule.

Table 3–10 Existence Checking for Contact: Acquisition Attributes

Acquisition Attributes	Entity	Transformation Function	Description	Type
Party Name	PARTY	WORD REPLACE + CLEANSE + REVERSE	Word replacement of Person and Organization names, removes vowels and double letters, reorders first word to the back.	-

Table 3–10 Existence Checking for Contact: Acquisition Attributes

Acquisition Attributes	Entity	Transformation Function	Description	Type
Contact Name	CONTACTS	WORD REPLACE + CLEANSE + REVERSE	Word replacement of Person and Organization names, removes vowels and double letters, reorders first word to the back.	Custom Attribute
Phone Number Flexible Format	CONTACT_POINTS	EXACT	Removes non-alphanumeric characters and white spaces.	Custom Attribute

Table 3–11 Existence Checking for Contact: Matching Attributes

Attribute Match	Match Threshold	Override Threshold	Automatic Merge Threshold
Match All	110	<null>	<null>

Table 3–12 Existence Checking for Contact: Scoring Attributes

Scoring Attribute	Entity	Score	Transformation Function	Description	Type	Weight (%)
Contact Name	CONTACTS	100	EXACT	Captures the exact string, removes non-alphanumeric characters, and forces upper case	Custom Attribute	100
-	-	-	WORD REPLACE + CLEANSE + REVERSE	Word replacement of Person and Organization names, removes vowels and double letters, reorders first word to the back.	-	90

Table 3–12 Existence Checking for Contact: Scoring Attributes

Scoring Attribute	Entity	Score	Transformation Function	Description	Type	Weight (%)
E-mail Address	CONTACT_POINTS	10	EXACT (E-mail)	Forces uppercase.	-	100
-	-	-	CLEANSE (E-mail)	Removes vowels and double letters.		90
Phone Number Flexible Format	CONTACT_POINTS	10	EXACT	Removes non-alphanumeric characters and white spaces.	-	100
URL	CONTACT_POINTS	10	CLEANSE (URL)	Removes non-alphanumeric characters, white spaces, vowels and double letters.	-	100

Note: For existence checking of Party (Organization & Person), Addresses, Contact and Contact Points, the Import Sales Lead concurrent program solely depends on the results returned by the DQM matching rules. The more effective the matching rule, the more precise the result.

3.5.4 Leads Data Quality

When lead records are imported, they are temporarily stored in the AS_IMPORT_INTERFACE table. After the records are processed for lead data quality, unique records are transferred to the AS_SALES_LEAD table.

The Import Sales Lead concurrent program uses the Deduplication Rule to identify duplicate lead records between the AS_IMPORT_INTERFACE table and the existing leads in the AS_SALES_LEAD table. The lead is identified as duplicate based on a set of attributes. If a lead is identified as a duplicate, the status of the lead is marked as Duplicate, and the record is not transferred to the AS_SALES_LEAD table. The PV: Run Lead Deduplication Rule profile must be set to Y for the concurrent program to run the deduplication rule.

While creating the deduplication rule, you can check for duplicate leads using the following lead attributes:

Mandatory Attributes

Customer

Optional Attributes Part of Seeded Rule

- Primary Contact
- Customer Address
- Campaign
- Response Channel
- Lead Note/Type

Other Available Optional Attributes

- Product Category
- Project
- Total Budget
- Total Purchase Amount - Product
- Budget Status
- Purchase Timeframe

Topics in this section include:

- [Section 3.5.4.1, "The Deduplication Rule"](#)
- [Section 3.5.4.2, "Customizing the Deduplication Rule"](#)
- [Section 3.5.4.3, "Custom Deduplication Using User Hooks"](#)

3.5.4.1 The Deduplication Rule

The deduplication rule offers you the flexibility to decide the attributes that will identify a duplicate lead for the requirements in your organization. You can also specify the number of days within which the lead should have been created. By default, all leads created in the last 7 days are checked for duplicates.

There is a seeded rule available, which you can customize. This is the algorithm that the seeded rule follows:

1. Look for duplicate customer (last name, first name). If duplicate, continue checking. Else unique lead.
2. Look at contact (name, address, and country). If duplicate, continue checking. Else unique lead.
3. Look at Campaign. If the matching lead(s) are created as a result of the same campaign, continue checking. Else unique lead.
4. Look at all other fields (vehicle response, first lead note). If all are duplicate, lead is duplicate. Else unique lead.

3.5.4.2 Customizing the Deduplication Rule

Use the following procedure to customize the deduplication rule to your requirements.

Prerequisites

None

Steps

1. Log in as an administrator and navigate to Administration > Leads > Processing > Deduplication.
2. Select the number of days from the Filter Attributes region.
The default value is 7. Leads created within the last *n* number of days are checked for duplicates.
3. To add additional attributes for the rule, select lead attributes from the drop-down lists in the Matching Attributes region.
These attributes are used to check for duplicates.
4. Click **Update**.
Your changes to the deduplication rule are saved.
5. Optionally, to remove an attribute, select it from the Remove column, and click **Update**.
The Customer attribute is mandatory, and cannot be removed.

The Deduplication Rule Flow

The flow of the deduplication rule is explained in [Table 3–13](#) using the settings in the seeded deduplication rule. The Result column in the table gives the outcome for each instance.

Table 3–13 Seeded Deduplication Rule Flow

Customer	Vehicle Response	Campaign	Customer Address	Primary Contact Last Name	Primary Contact First Name	Lead Note/Type	Result
Digital Harvester	E-mail	Laptop Cross Sell	500 Oracle Pkwy	Lorna	Bennie	Note1	Master Lead
Digital Harvester	E-mail	Laptop Cross Sell	500 Oracle Pkwy	Lorna	Bennie	Note1	Exact Duplicate Lead
Digital Harvester	E-mail	Laptop Cross Sell	500 Oracle Pkwy	Irvin	Bennie	Note1	Unique Lead (fails first duplicate check)
Digital Harvester	Phone	Laptop Cross Sell	500 Oracle Pkwy	Lorna	Bennie	Note1	Unique Lead (fails Vehicle Response duplicate check)
Digital Harvester	E-mail	Printer Cross Sell	500 Oracle Pkwy	Lorna	Bennie	Note1	Unique Lead (fails Campaign duplicate check)
Digital Harvester	E-mail	Laptop Cross Sell	500 Oracle Pkwy	Lorna	Bennie	New Note1	Unique Lead (fails first note check)
Digital Harvester	E-mail	Laptop Cross Sell	500 Oracle Pkwy	Lorna	Bennie	Note1 New Note2	Duplicate Lead (second note not checked)

Variants in any other fields do not affect this check (such as Role, Source System, SIC code, and so on).

Contact Restrictions

The Import Sales Lead concurrent program allows you to set restrictions for the `do_not_phone_flag`, `do_not_fax_flag`, `do_not_email_flag`, and `do_not_mail_flag` flags for the contact (`relationship_party_id`), and the `do_not_mail_flag` for the address (`party_site`). If you want to set the restrictions, set these flag values to `Y`.

Contact Points

The Import Sales Lead concurrent program creates the Phone, E-mail, Web, and Fax contact points.

3.5.4.3 Custom Deduplication Using User Hooks

User hooks permit you to bypass Oracle code and implement custom functions instead. Use the following user hook to implement a custom function and check for duplicate leads.

The custom function is executed only if the lead is identified as unique by the Import Sales Lead concurrent program.

Hook Name: IS_DUPLICATE_LEAD

Package Name: AS_IMPORT_SL_CUHK

Purpose

While importing leads, the Import Sales Lead concurrent program does not check for duplicate leads that may be stored in other third-party applications.

To implement custom lead duplicate checking, write a package according to the following specifications. The Import Sales Lead concurrent program creates either a new lead or skips a lead import record based on the value returned by your program. If the record is skipped, then no lead is created and the `load_status` of that lead import record is set to `DUPLICATE`.

Do not commit in the package body. After the transaction is completed, Oracle Application code issues a commit.

This user hook is called by the Import Sales Lead concurrent program.

Calling Package

AS_IMPORT_SL_PVT.Is_Duplicate_Lead

API Name

Is_Duplicate_Lead_Pre

Procedure Specification

```
CREATE or REPLACE PACKAGE as_import_sl_cuhk IS
  PROCEDURE Is_Duplicate_Lead_Pre(
    p_api_version_number    IN NUMBER,
    p_init_msg_list         IN VARCHAR2 := FND_API.G_FALSE,
```

```

        p_validation_level      IN  NUMBER          := FND_API.G_VALID_LEVEL_
FULL,
        p_commit                IN  VARCHAR2        := FND_API.G_FALSE,
        p_import_interface_id   IN  NUMBER,
        x_duplicate_flag        OUT NOCOPY VARCHAR2,
        x_return_status         OUT NOCOPY VARCHAR2,
        x_msg_count             OUT NOCOPY NUMBER,
        x_msg_data              OUT NOCOPY VARCHAR2
    );
END as_import_sl_cuhk;
/

```

Procedure Body

```

CREATE or REPLACE PACKAGE BODY as_import_sl_cuhk AS

    PROCEDURE Is_Duplicate_Lead_Pre(
        p_api_version_number    IN  NUMBER,
        p_init_msg_list         IN  VARCHAR2        := FND_API.G_FALSE,
        p_validation_level      IN  NUMBER          := FND_API.G_VALID_LEVEL_
FULL,
        p_commit                IN  VARCHAR2        := FND_API.G_FALSE,
        p_import_interface_id   IN  NUMBER,
        x_duplicate_flag        OUT  VARCHAR2,
        x_return_status         OUT  VARCHAR2,
        x_msg_count             OUT  NUMBER,
        x_msg_data              OUT  VARCHAR2
    ) IS
BEGIN

    /*
    Custom code goes here for lead de-duplication
    Assign:
        x_duplicate_flag = 'Y' if lead is to be marked 'DUPLICATE'
        else
        x_duplicate_flag = 'N'
    */

    null;

END;

END as_import_sl_cuhk;
/

```

In Parameters

The four parameters below are standard inputs:

Parameter	Description
p_api_version_number	For 11 <i>i</i> Oracle Sales application, this is 2.0.
p_init_msg_list	Initialize message stack or not. This is set to FND_API.G_FALSE by default.
p_validation_level	Validation level for pass-in values. This is set to FND_API.G_VALID_LEVEL_FULL by default.
p_commit	To commit the whole API at the end of API, set to FND_API.G_FALSE. This is the default value.

The following parameter does not have a standard input:

Parameter	Description
p_import_interface_id	The import interface identifier. Pass the import_interface_id of the lead import record for which you want to perform the lead existence checking.

Out Parameters

The following three parameters are standard output parameters.

Parameter	Description
x_return_status	The return status. If your code completes successfully, then FND_API.G_RET_STS_SUCCESS must be returned. If you get an expected error, then return FND_API.G_RET_STS_ERROR, otherwise return FND_API.G_RET_STS_UNEXP_ERROR.
x_msg_count	The message count. Call FND_MSG_PUB.Count_And_Get to get the message count and messages.
x_msg_data	The messages. Call FND_MSG_PUB.Count_And_Get to get the message count and messages.

The following parameter does not have a standard output:

Parameter	Description
x_duplicate_flag	Indicates the status of the lead. Y indicates the lead import record is a duplicate and was not imported. N indicates the lead import record is not a duplicate and was imported.

3.6 Custom Codes with the Lead Import Program

In addition to the lead import process, you can subscribe to business events that will execute prior to and after the Import Sales Lead concurrent program is run. You can also customize the user hook provided to check for a party in TCA.

- [Section 3.6.1, "Seeded Business Events"](#)
- [Section 3.6.2, "Custom User Hook"](#)

3.6.1 Seeded Business Events

Subscribe to business events when you want additional processing on lead records, or when you want a specific result at the end of the lead import process. You can specify the function codes to run by associating them with a business event, and subscribing to the event.

What is a Business Event?

A business event is an occurrence of any logical event in the application. Examples of business events are Creating a lead, Importing a lead, or Converting a lead to opportunity.

What is Event Subscription?

Event subscription is a pointer to a function code or a workflow. With every business event, you can register one or more subscriptions. When a business event occurs, the subscription(s) associated with the event are called, and the associated function code is executed. If there are more than one subscriptions associated with an event, then the order of execution of these subscriptions is decided from the phase number associated with each subscription.

A Pre function code is executed prior to running the Oracle code and a Post function code is called after the Oracle code is executed. In Oracle Leads

Management, the Lead Import Event - Pre and Lead Import Event - Post functions are seeded for the Importing a Lead event.

Before starting to import a batch of records, the Import Sales Lead concurrent program raises the Lead Import Event - Pre function using the `WF_EVENT.Raise()` call. After the import process for a lead is completed, the Lead Import Event - Post function is executed. The same parameters that are passed to the Import Sales Lead concurrent program are passed to these functions as well. These parameters are `batch_id`, `source_system`, `debug_flag`, and `purge_error_flag`. For information about these parameters, see [Section 3.4.3.1, "Import Sales Lead Concurrent Program Parameters"](#).

If there is any exception raised from the function's code, the lead import process for the particular record is terminated, and the `load_status` for the record is marked as `ERROR`.

In general, there can be more than one function subscribed to an event. If there are no functions subscribed, then the control is returned to the calling program without raising an error or exception.

Alternatively, a workflow process can also be associated with an event.

References

- For sample subscription codes, see [Section 3.6.1.3, "Sample Function Codes for the Business Event Subscriptions"](#).
- To subscribe to a business event, see the *Oracle Workflow Developer's Guide*.

3.6.1.1 Error Handling

If there are any exceptions generated from the functions, the Import Sales Lead concurrent program terminates abnormally and any further execution is marked as `Error`. The functions must be able to handle these exceptions themselves. If the Business Event system itself raises any errors while calling the `WF_EVENT.Raise()` method, then the error is placed on the `WF_ERROR` queue and a notification is sent to the System Administrator.

3.6.1.2 Execution Control

If multiple subscriptions are defined for the same event, you can control the order in which the Event Manager executes the subscriptions by specifying a phase number for each subscription. Subscriptions are executed in ascending phase order. For example, you can enter 10 for the subscription that you want to execute first when an event occurs, 20 for the subscription that you want to execute second, and

so on. You can use phases to ensure that different types of actions are performed in the appropriate order, such as executing subscriptions that perform validation before subscriptions that perform other types of processing. If you enter the same phase number for more than one subscription, the Event Manager may execute them in any order, relative to each other. However, the Event Manager will execute that group of subscriptions in their specified place in the phase order, relative to subscriptions with other phase numbers.

You can also use the phase number to control whether the subscription is executed immediately or is deferred. The Event Manager treats subscriptions with a phase number of 100 or higher as deferred subscriptions (asynchronous). Subscriptions with a phase number from 1 to 99 are executed immediately (synchronously). The phase number for a lead import subscription must always be between 1 and 99. The phase number 0 (zero) is reserved for Oracle Workflow seeded subscriptions and should not be used.

3.6.1.3 Sample Function Codes for the Business Event Subscriptions

Use the following code as a sample to create your pre and post business event functions.

Sample Package Specification

```
CREATE OR REPLACE PACKAGE aml_import_event IS

    FUNCTION Pre(p_subscription_guid in raw,
                p_event in out wf_event_t) return varchar2;

    FUNCTION Post(p_subscription_guid in raw,
                 p_event in out wf_event_t) return varchar2;

END;
```

Sample Package Body

```
CREATE OR REPLACE PACKAGE BODY aml_import_event IS
```

Sample Function Code - Pre

```
FUNCTION Pre(p_subscription_guid in raw,
            p_event in out wf_event_t) return varchar2
    IS
    l_parameter_list wf_parameter_list_t;
    l_batch_id VARCHAR2(15);
    l_source_system VARCHAR2(30);
    l_debug_msg_flag VARCHAR2(1);
```

```

l_PURGE_ERROR_FLAG VARCHAR2(1);
l_string VARCHAR2(50);

BEGIN
--Get parameters:
l_parameter_list := WF_PARAMETER_LIST_T();
l_parameter_list := p_event.GetParameterList;

l_source_system := wf_event.GetValueForParameter('P_SOURCE_SYSTEM',l_parameter_
list);
l_batch_id := wf_event.GetValueForParameter('P_BATCH_ID',l_parameter_list);
l_debug_msg_flag := wf_event.GetValueForParameter('P_DEBUG_MSG_FLAG',l_
parameter_list);
l_PURGE_ERROR_FLAG := wf_event.GetValueForParameter('P_PURGE_ERROR_FLAG',l_
parameter_list);

/*
<-- CUSTOM GOES HERE -->

l_string := substr('Pre- '||l_batch_id||'- '||l_source_system||'- '||l_debug_msg_
flag
                ||'- '||l_PURGE_ERROR_FLAG,1,50);

commit;
<-- END CUSTOM CODE -->
*/

return 'SUCCESS';
END Pre;

```

Sample Function Code - Post

```

FUNCTION Post(p_subscription_guid in raw,
             p_event in out wf_event_t) return varchar2
IS
l_parameter_list wf_parameter_list_t;
l_batch_id VARCHAR2(15);
l_source_system VARCHAR2(30);
l_debug_msg_flag VARCHAR2(1);
l_PURGE_ERROR_FLAG VARCHAR2(1);
l_string VARCHAR2(50);

BEGIN
--Get parameters:

```

```

l_parameter_list := WF_PARAMETER_LIST_T();
l_parameter_list := p_event.GetParameterList;

l_source_system := wf_event.GetValueForParameter('P_SOURCE_SYSTEM',l_parameter_
list);
l_batch_id := wf_event.GetValueForParameter('P_BATCH_ID',l_parameter_list);
l_debug_msg_flag := wf_event.GetValueForParameter('P_DEBUG_MSG_FLAG',l_
parameter_list);
l_PURGE_ERROR_FLAG := wf_event.GetValueForParameter('P_PURGE_ERROR_FLAG',l_
parameter_list);

/*
<-- CUSTOM GOES HERE -->

l_string := substr('Post- '||l_batch_id||'- '||l_source_system||'- '||l_debug_
msg_flag
                ||'- '||l_PURGE_ERROR_FLAG,1,50);

commit;

<-- END CUSTOM CODE -->
*/

return 'SUCCESS';
END Post;
END aml_import_event;
/

```

3.6.2 Custom User Hook

You can execute a custom user hook that will be called from the Import Sales Lead concurrent program before DQM processes each record. This user hook tries to find a `party_id` with an exact customer name match in the `HZ_PARTIES` table in TCA. Calling the user hook from the Import Sales Lead concurrent program is controlled by the `OS:Execute Custom Code` from Lead Import profile. If the value of the profile is `Yes`, then the program is invoked.

Purpose of a Custom User Hook

Consider the following scenario where the custom user hook is used. A batch has two records with the same customer name, and the customer is not recorded in the `HZ_PARTIES` table in TCA. The custom user hook processes the first record in the batch and tries to find an exact string match with `party_name` in the `HZ_PARTIES` table. Because this is a new party, the query will not return any records. Next, DQM

is invoked to find a party match. This would also fail because it is a new party. Finally, the Import Sales Lead concurrent program creates a new customer in TCA, and creates a lead against it.

While processing the second record, before calling DQM, the user hook returns the `party_id` of the newly created customer. Hence, DQM processing for the record will be bypassed.

If the user hook is not available, DQM will fail because the synchronization is not done between the DQM Staging Schema and the TCA before the second row is processed. See [Section 3.4.5, "Limitation of the Import Sales Lead Concurrent Program"](#).

On the other hand, if the user hook returns more than one record for the exact customer match, it would mean that the customer was created before this import batch was loaded, and should have been present in the DQM staging area. Therefore, the program should call DQM to find the one right match among the multiple matches.

Note:

- The user hook party match program must be properly maintained for it to be effective.
 - The user hook party match program can find a matching party only if the customer names in the batch are exactly the same.
-
-

Performance Impact

Running the user hook may affect performance because the program queries the `HZ_PARTIES` table for a `party_id` with the exact `party_name`.

3.6.2.1 Seeded Code for the User Hook

The following is the seeded PL/SQL package that is seeded for the user hook. You can customize it to suit your setup.

Procedure Specification

```
PACKAGE aml_find_party_match_pvt

    PROCEDURE main (imp           IN OUT NOCOPY as_imp_int%ROWTYPE,
                   X_return_status OUT NOCOPY varchar2)
    l_party_id  number;
```

```
Begin

    X_return_status := FND_API.G_RET_STS_SUCCESS;

    SELECT party_id
    INTO l_party_id
    FROM    hz_parties hzp
    WHERE   hzp.party_name = imp.customer_name;

    Imp.party_id := l_party_id;

    UPDATE as_import_interface
        SET party_id = l_party_id
        WHERE import_interface_id = imp.import_interface_id;

Exception
    When NO_DATA_FOUND Then
        l_party_id := NULL;
    When TOO_MANY_ROWS Then
        l_party_id := NULL;

End main;
```

3.7 Purging Staged Lead Records

The `AS_IMPORT_INTERFACE` table is a temporary location where the imported records are stored before unique records are moved to the `AS_SALES_LEAD` table. After the records are checked for duplicates, the remaining records in the `AS_IMPORT_INTERFACE` may not be required. If they are retained in the `AS_IMPORT_INTERFACE` table, the number of records may affect the performance of the Import Sales Lead concurrent program.

You can delete the records from this table by running the Purge Leads Import Interface Table concurrent program.

Use the following details to run the concurrent program.

Prerequisites: None

Responsibility: Oracle Sales Administrator

Parameters:

- **From and To** - This date range indicates the dates when the lead was created. When the program is run, all leads that were created in the specified range are deleted. For example, if the From and To Dates are 3-Nov-2003 and

15-Nov-2003, all leads created between the 3rd and 15th of November, 2003 are deleted from the table.

- Load status of records to be purged - Status of the leads that you want to purge
- Debug - Enter *Yes* if you want to see debug messages.
- SQL Trace - Enter *Yes* if you want to trace SQL.

Schedule: Once

For the steps to run the concurrent program, see [Section C.1, "Running Concurrent Programs"](#).

Linking Interactions to Leads

This chapter describes the functioning of the Interaction Matching Engine.

Topics included are:

- [Section 4.1, "Interaction Matching Engine Overview"](#)
- [Section 4.2, "Functioning of the Interaction Matching Engine"](#)
- [Section 4.3, "Setting Up Interaction Types"](#)
- [Section 4.4, "Interaction Rule Sets"](#)

4.1 Interaction Matching Engine Overview

An interaction is a single contact event between a customer or customer system and a resource of the business. An example of an Interaction with a potential customer would occur when marketing campaigns target lists of e-mail addresses for mass mailings or when a person calls a call center to get product information.

The Interaction Matching Engine provides the mechanism to mine and evaluate these customer interactions and responses for sales follow up. Interaction Matching rules can be set up to determine the marketing interactions that must be evaluated to generate sales leads or to enhance the quality of existing leads.

To understand the Interaction Matching Engine and associated concepts, see the following sections:

- [Section 4.1.1, "Interaction Types and Interaction Scores"](#)
- [Section 4.1.2, "Interaction Score Threshold"](#)

4.1.1 Interaction Types and Interaction Scores

Customer interactions are of various types. Examples of interaction type are Event Registration, Survey Completion, Web Visit, and so on.

Each interaction type is associated with a default interaction score that indicates the significance of the interaction. For example, an interaction where the customer tells the call center agent to have a sales representative call back is a very significant type of interaction. This event has an immediate sale potential, and merits a high score. Whereas, an interaction type such as Event Registration may indicate that the customer has some interest in the product associated with the event. However, it does not merit immediate sales attention, and hence is worthy of a low score.

[Table 4–1](#) lists the seeded interaction types and the default interaction scores considered by the Interaction Matching Engine. The Seeded Interaction Rule uses these interaction scores. To set up additional interaction types, see [Section 4.3, "Setting Up Interaction Types"](#).

Table 4–1 Seeded Interaction Types and Scores

Interaction Type	Description	Default Interaction Score
Survey Callback	Customer has requested a call back from the company.	10
Event Registration	Customer has registered for an event organized by the company.	3

Table 4–1 Seeded Interaction Types and Scores

Interaction Type	Description	Default Interaction Score
Survey Completion	Customer has completed taking a survey posted by the company.	3
Web Collaboration	Customer has interacted with the company through the Web.	2
Inquired about an item from Customer	Customer has enquired about an item of the company.	3
Web Advertisement Visits	Customer has clicked on the company's advertisement on the web for details.	1
Web Offer Visits	Customer has clicked on the company's offer on the web for details.	1
E-mail Clickthroughs	Customer has clicked on a URL in an e-mail sent by the company.	1

Note: The scores for the interaction types can be changed when you are setting up rule sets in the Interaction Matching Engine. For more information, see [Section 4.4, "Interaction Rule Sets"](#)

4.1.2 Interaction Score Threshold

The Interaction Score Threshold (IST) is a set value against which the sum of all the interaction scores for a customer are matched. When the sum is equal to or greater than the IST, a lead is generated from the set of interactions. Each time an interaction is linked to a customer, the sum of interaction scores are evaluated against the IST.

For example, assume that the IST is set to 20. A customer interaction with a score of 5 is chosen by the Interaction Matching Engine. However, no lead is available for the corresponding customer. When another interaction for the same customer with a score of 10 is recorded, both the interaction scores are added and compared with the IST. In this case, the score does not exceed the IST yet. A lead is created from these interactions when the sum of all the interaction scores equal or exceed the IST.

The IST value is set in the OS: Interaction Score Threshold profile. The default value for this profile is 20.

4.2 Functioning of the Interaction Matching Engine

The Interaction Matching Engine functions in two key ways:

- If no lead exists for a customer, new marketing interactions for the customer are mined and used to generate a lead for sales follow up.
- After a lead is generated, or if a lead already exists for a customer, new marketing interactions for the customer are matched with the lead to improve its quality.

To understand the working of the Interaction Matching Engine, see the following sections:

- [Section 4.2.1, "The Mining Behavior of the Interaction Matching Engine"](#)
- [Section 4.2.2, "Linking an Interaction with Existing Leads"](#)
- [Section 4.2.3, "Linking if No Lead Exists for the Interaction"](#)
- [Section 4.2.4, "Running the Concurrent Program"](#)

4.2.1 The Mining Behavior of the Interaction Matching Engine

The Interaction Matching Engine is driven by a rule set that has an activation date range. A concurrent program (Run Interaction Matching Engine to Match or Create Leads) runs the Interaction Matching Engine at scheduled times. Based on the rule set, interactions are evaluated for their sales potential.

The Guards in the rule set support the Country and Campaign (Source Code) attributes. The attributes are ANDed during evaluation. You can also opt to capture all interactions without any attributes. The Rules in the rule set support the Interaction Type and the Interaction Score.

If the Interaction Matching Engine rules sets are modified, they are applicable only for the new interactions captured since the last time the engine ran. They are not used to match interactions that were processed before the rule sets changed.

All the matched interactions are tracked and displayed with the timestamp on the lead. You can also browse through all the responses that contributed to the lead. Review and periodically archive interactions that are not matched with leads or other interactions. You can view the interactions attached to a lead from the history details for the lead. For more information, see the *Oracle Field Sales User Guide*.

Note: Interactions with a Guest User's Party ID are not used for matching.

The mining behavior can be customized with business-specific logic that renders certain interactions in specific contexts valuable in determining sales value.

Mining by Source Code

The significance of an interaction may depend on the specific promotion associated with it. For example, an Event Registration for a strategic CEO level event is more significant than an Event Registration for a technical web seminar. Hence, the score of the interaction also depends on the promotion code associated with the interaction. For this type of mining, you can create specific rules that will leverage the source code as a guard.

In addition to the source code attribute, the country attribute can also be used as a guard to define interaction rules.

4.2.2 Linking an Interaction with Existing Leads

When an interaction is captured for linking, the Interaction Matching Engine first checks all the existing sales leads to match the interaction with a lead. The interaction is matched with leads based on the following criteria:

- **Creation Date Range** - This is the date range when the lead is created. For example, the interaction is compared with all leads created in the last 60 days.
- **Lead Status** - The status of the lead must be Open.
- **B2B and B2C** - The Interaction Matching Engine supports matching of Business to Business (B2B) as well as Business to Customer (B2C) leads. For B2B situations, the Customer (Organization) ID recorded in the interaction must match the Customer ID recorded in the lead. Optionally, the contact recorded in the interaction (if any) must match one of the contacts recorded on the lead. The profile OS: Interaction Matching Engine Matches Contact for B2B lead determines whether the primary contact should also be matched for B2B leads.

If more than one lead matches a given interaction, then a lead is chosen by its progress in the sales pipeline.

- Based on the rating of the lead, the lead with the better rating is chosen.
- Based on the creation date of the lead, the lead created most recently is chosen.

Lead Reevaluation

When an interaction is linked to a lead, the lead is reevaluated only if both these conditions are satisfied:

- One of the attributes in the Qualifying, Rating, or Channel Selection engine rule sets is Interaction Score.
- The OS: Auto Run Lead Engines While Update profile is set to `Yes`.

The Interaction Score can be leveraged in the Rating Engine to upgrade the lead value, along with any other relevant attributes. When new (additional) interactions are linked, the score of the new interaction will augment the rating of the lead.

When an unqualified lead is appended with an interaction, and the lead's Interaction Score goes up, the lead is processed by the qualification, rating, and channel selection engines again.

4.2.3 Linking if No Lead Exists for the Interaction

If a matching lead is not found for an interaction, the interaction is evaluated, and if the interaction score is equal to or greater than the IST score, then a lead is generated.

If the interaction score does not meet the IST, the interaction is linked with other unique interactions for the same party. All matching interactions for the prospect within the allowed time frame are scored and evaluated as an aggregate score. This aggregate score is constantly reevaluated against the IST, and a lead is created when appropriate. All interactions used to generate the lead are appended to the lead.

Lead Generation

When a lead is generated from a set of interactions, the lead is assigned to a sales channel with the following details:

- The name of the lead. This is taken from the interactions from which the lead was generated. The name could be the Contact first name or last name; or the name of the Organization.
- The customer and contacts are populated with the parties in the interactions. The party details can also be linked for contact point information.
- The source code is either the source code of the most recent interaction, or the source code of the interaction with the highest score. All the other source codes are recorded as lead lines along with the associated products.

- If present, all products and related offers associated with the interaction or source code are recorded as lead lines.

4.2.4 Running the Concurrent Program

The Run Interaction Matching Engine to Match or Create Leads concurrent program runs the Interaction Matching Engine.

Use the following details to run the Run Interaction Matching Engine to Match or Create Leads concurrent program.

Prerequisite: Interaction Matching Rules must be set up.

Responsibility: Oracle Sales Administrator

Parameters:

- Debug - Enter Yes if you want to see debug messages.
- SQL Trace - Enter Yes if you want to trace SQL.

Schedule - Once

For the steps to run the concurrent program, see [Section C.1, "Running Concurrent Programs"](#).

4.3 Setting Up Interaction Types

All customer interactions can be associated with a specific type. Examples of interaction types are Event Registration, Survey Completion, and Web Visit. You can set up interaction types that you want associated with the interaction rules.

Use the following procedure to set up interaction types.

Prerequisites

None

Steps

1. Log in as an administrator and navigate to Administration > Leads > Setup > Interaction Type.

The Interaction Type Setup page appears. The first few interaction types are seeded, and cannot be selected. You can enable or disable a seeded interaction type, and also change the description and meaning, if required.

2. To set up new interaction types, click the Next hyperlink.
3. Select an Action, and a corresponding Action Item for the interaction type.
4. Select the Enabled checkbox.

An interaction type that is not enabled is not visible while creating rules for the Interaction Matching Engine rule sets.

5. Enter the Meaning and Description for the interaction type.

The term entered in the Meaning column will appear as the Interaction Type when you create an interaction rule set.

6. Click **Update**.

4.4 Interaction Rule Sets

The interaction rules sets identify the interactions that are appropriate and relevant to be mined for follow up purposes.

The Linking rules can be customized to:

- Find the particular business interactions that merit linking such as completed surveys, event registration, web collaboration, web visits, or inbound calls.
- Identify worthy business interactions that merit lead generation using promotion source codes.

If more than one rule set with equal precedence win, the rule set with the highest score is selected.

You can view the interactions attached to a lead from the history details for the lead. For more information, see the *Oracle Field Sales User Guide*.

You can track the effectiveness of the interaction rule sets by viewing the Lead Linking Report. For more details, see [Section 7.5, "Lead Linking Report"](#).

4.4.1 Creating an Interaction Rule Set

An interaction rule set contains a guard and rules. The Country and Campaign attributes make up the guard options. Rules can be created using Interaction Types and Scores. You can view the details of the seeded interaction rule set called Seeded Interaction Rule.

When an interaction matches with the rule set conditions, a score is assigned to the interaction based on the type of the interaction. When the score reaches a threshold, a lead is created from the interaction(s).

While creating a rule set, it is in the Draft status by default. Use the following procedure to create an interaction rule set.

Prerequisites

None

Steps

1. Log in as an administrator and navigate to Administration > Leads > Processing Rules > Interaction Selection.
2. Click **Create**.
3. Enter the Name and Description for the rule set.
4. Select the Start and End Dates.
These are the dates between which the rule set is valid.
5. Enter a value for the precedence of the rule set.
Each rule set can have a different precedence, to define the order of importance for evaluation (where 100 is higher than 1).
6. Select values for the Currency and Owner fields.
7. Click **Create**.
8. If you do not want to filter any interactions, select Match all Interactions.
9. To filter interactions based on their attributes, select Match Interactions based on Guard.
Guards define the domain to which the rule set applies such as campaign-specific or country-specific.
 - a. Click **Add Attributes**.
 - b. Select Campaign or Country from the Profile Attribute Name drop-down list.
The page reloads to display the values for the profile that you have chosen.
 - c. From the Condition drop-down list, select a condition for the profile.
 - d. Select a value(s) for the profile.

- e. Click **Apply**.

The attribute is added.

- 10. Optionally, click **Apply And Add Another** to add another attribute.
- 11. Optionally, to modify an attribute, click the hyperlinked Profile Attribute Name.
- 12. Optionally, to remove an attribute, select it from the Remove column in the Rule Set: Guard region, and click **Update**.

You must now add rules to the rule set.

- 13. Click the Rule link.
- 14. From the Interaction Type drop-down list, select a type.
- 15. Enter a value for the score.
- 16. Click **Update**.

The interaction rule set is saved.

Processing Leads

This chapter explains the processing and assigning of leads by the Leads Processing Engine.

Topics included are:

- [Section 5.1, "Leads Processing Engine Overview"](#)
- [Section 5.2, "The Qualification Engine"](#)
- [Section 5.3, "The Rating Engine"](#)
- [Section 5.4, "The Channel Selection Engine"](#)
- [Section 5.5, "Best Practices"](#)
- [Section 5.6, "Working with Rule Sets"](#)
- [Section 5.7, "The Rule Flows Report"](#)
- [Section 5.8, "Purging Unqualified Leads"](#)
- [Section 5.9, "Setting Up Lead Assignments"](#)
- [Section 5.10, "Setting Up Lead Status"](#)
- [Section 5.11, "Using Custom Attributes"](#)
- [Section 5.12, "Setting Up Time Frames"](#)

5.1 Leads Processing Engine Overview

The Leads Processing engine comprises the qualification engine, the rating engine, and the channel selection engine.

- The Qualification Engine confirms that there is sufficient interest for a selling interface to engage the prospect through a touch point. See [Section 5.2, "The Qualification Engine"](#).
- The Rating Engine grades leads using business-specific logic appropriate to different campaign strategies, regions, or products. Rating enables the business to prioritize follow-up and response handling activities for effective cost and resource management. See [Section 5.3, "The Rating Engine"](#).
- The Channel Selection Engine channels the leads to a set of resources for assignment - direct or indirect, depending on its quality or domain. A sales channel is direct when routed to the sales force within the company and indirect when routed to partners or value added distributors. See [Section 5.4, "The Channel Selection Engine"](#).

The engines are based on a generic rules model, which consists of Guards, Rules, and Precedence.

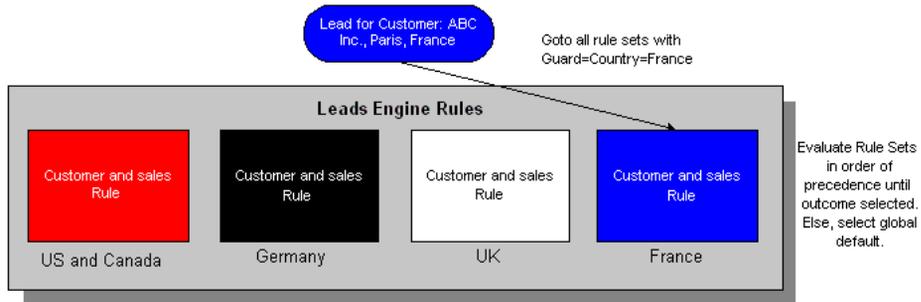
Guards

Guards are used to group rule sets into domain-specific buckets. They parse rule sets into groups based on business-specific practices. Each rule set defines the set of leads to which it applies such as product-specific, campaign-specific, and country-specific lead processing for each stage of lead evaluation.

Guards can have multiple conditions. There is an implicit AND across conditions and an implicit OR within conditions. For example, if the Guard is defined as Country = France, Germany, UK; Product Category = Printers, Desktops, then this is interpreted as evaluate all leads that originate from countries France or Germany or UK for product lines Printer or Desktops.

For example, see [Figure 5–1](#) for country-specific guards.

Figure 5–1 Country-specific Guard



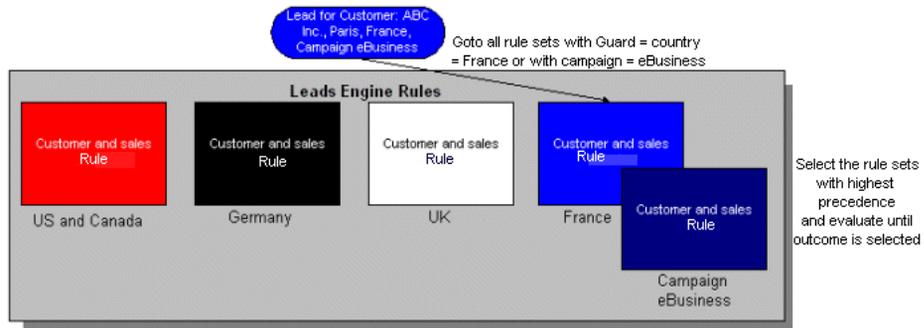
Precedence

After the rule sets are bucketed into different groups, the Precedence of each rule set is used to determine the order of evaluation. For example, if the attributes of a lead are matched with a Country-specific and a Campaign-specific rule set, by assigning the Country-specific rule set a higher precedence, this rule set is evaluated before the Campaign-specific rule set.

For precedence, 100 is higher than 1.

See [Figure 5–2](#) for an example on precedence in rule sets.

Figure 5–2 Precedence in Rule Sets



Rules

Using the guards, after the correct rule sets are selected, the rules of each rule set determine the conditions and action to be performed on the lead. For example, if certain conditions are true at the time of evaluation, the lead is set to qualified, or rated A, or channelized to Direct Sales.

Rules are evaluated in precedences from 1-n, where 1 is evaluated first. On evaluation, the winning rule set with the highest precedence is used to select the rule set result. If more than one winning rule set has equal precedence, the best or the highest ranked result, wins.

Reports

The Leads Processing History and Rule Performance Reports help you analyze the effectiveness of the rule sets in an engine. To troubleshoot rule set issues, see the Rule Diagnostics Report. See [Chapter 7, "Operational Reports"](#) for more information about reports.

For real life examples to use the Leads Processing Engine, see [Section 2.2, "Scenarios Using the Leads Processing Engine"](#).

5.2 The Qualification Engine

When a lead is run through the Leads Processing Engine, it is processed by the Qualification Engine (QE) first. In this release, the QE has two primary functions:

- **Qualifying a Lead:** A lead can be qualified when the attributes of the lead indicate interest in the purchase of a product. Some of these attributes can be: the lead has attended a product event, or the budget has been approved, or the purchase time frame is less than a year. Such details suggest that it is worthwhile for a sales team to follow up with the lead, and possibly arrive at a sale.
- **Disqualifying a Lead:** A lead can be disqualified when the attributes indicate that the lead is not genuinely interested in the purchase of a product. If the lead belongs to the Student category, or has an incomplete or incorrect e-mail address, or belongs to a competitor company, such leads can be filtered. They need not be pursued to a sales closure.

Note: At any point in time, it is recommended that you do not have rule sets to both qualify and disqualify leads.

At the end of the qualification process, a lead may be qualified or disqualified. All qualified leads are routed to the Rating engine. All disqualified leads are routed to the Channel Selection engine.

5.2.1 How Does the Qualification Engine Work?

Each rule set is a grouping of rules. The rule set is defined by its Guard. The rules with a guard define the criteria and outcome.

The QE identifies the rule sets that can evaluate the lead by comparing the guard values in the rule set with the attributes of the lead being processed. For example, if the lead has Campaign A as an attribute, the QE searches for a rule set with Campaign A as a guard value.

After the matching qualification rule sets are identified, the engine starts evaluating the rules of each rule set, starting with the rule set of the highest precedence.

When a rule set wins, i.e., all the qualification rules of the rule set are met for the lead, the engine stops evaluation. Depending on the outcome, the lead is then qualified or disqualified, and the winning rule set is logged into a history table for analysis.

If no rule sets win, the lead is set to the value specified in the OS: Default Qualified Flag for Lead Qualification Engine profile. The default value for this profile is No.

In the case where two rule sets win, and one rule set qualifies the lead, and the other disqualifies, the lead is qualified by the QE.

See Also

[Section 5.6, "Working with Rule Sets"](#)

5.3 The Rating Engine

After a lead is qualified by the Qualification Engine, it is processed by the Rating Engine (RE). The RE prioritizes the leads based on their attributes, and assigns them ratings. The rating helps the sales representative decide the importance of a lead, and accordingly follow up with the lead.

Topics in this section include:

- [Section 5.3.1, "How Does the Rating Engine Work?"](#)
- [Section 5.3.2, "Setting up Ratings"](#)

5.3.1 How Does the Rating Engine Work?

When a lead is run through the RE, the engine first identifies the correct Rating rule set to evaluate. This process finds all matching rule sets by applying the lead attribute values against each rule set's guard values. For example, if the lead has Campaign A as an attribute, the RE looks for rule sets with Campaign A as a guard value.

After a matching Rating rule set is identified, the engine starts evaluating the rules for each rule set, starting with the rule set of the highest precedence. The RE evaluates the rules in the order of evaluation assigned and stops when it finds a rule that matches the lead.

When a rule wins, that is, all the criteria are met for the lead, the RE stops evaluation. The lead is then assigned a rating, and the winning rule set is logged into a history table for analysis.

If more than one rule set with equal precedence win, the highest rating is selected.

If no rule sets win, the rating set in the OS: Default Rating for Lead Rating Engine profile is used. The default value for this profile is Cold Lead. This profile value must not be set to blank.

5.3.2 Setting up Ratings

Ratings must be set up before rating rule sets are defined. Use this procedure to set up ratings.

Prerequisites

None

Steps

1. Log in as an administrator, and navigate to Administration > Leads > Setup > Rating.
2. In the Grade field, enter a grade for the lead.
The lead is rated with this grade.
3. In the Description field, enter a brief description of the grade.
4. In the Precedence field, enter a number to set the precedence for the grade.

The precedence is used when the attributes of the lead match with more than one grade. In this case, the lead is assigned the grade with a higher precedence. For precedence, 100 is higher than 1.

5. Select the Enabled check box to include this grade to rate leads.
6. Click **Update**.
7. Optionally, click **Restore** to clear a grade that you are setting up.
The grade is cleared.
8. Optionally, select the check box in the Remove column to remove a grade.
9. Click **Update**.
The grade is removed.

Note: When you create a rule set, the grades that you just set up may not appear in the drop-down lists. You must restart the apache to reload data from the database.

See Also

[Section 5.6, "Working with Rule Sets"](#)

5.4 The Channel Selection Engine

The Channel Selection Engine (CSE) is responsible for distributing leads to the appropriate teams for further follow up and action. Based on channel selection rules and the lead attributes, a lead is assigned to a channel.

Examples of channels are Inside Sales, Direct Sales, Indirect Sales, and Partner.

Topics included in this section are:

- [Section 5.4.1, "How Does the Channel Selection Engine Work?"](#)
- [Section 5.4.2, "Setting Up Channels"](#)

5.4.1 How Does the Channel Selection Engine Work?

The CSE is similar to the Qualification and Rating engines. When a lead is run through the CSE, the engine first identifies the correct Channel Selection rule set to evaluate. This process finds all matching rule sets by applying the lead attribute values against each rule set's guard values. For example, if the lead has United

States as its Country attribute, the CSE looks for rule sets with United States as a guard value for Country.

After the matching channel selection rule sets are identified, the engine starts evaluating the rules, starting with the rule with the highest precedence. Each rule has an order of evaluation associated with it. The Channel Selection Engine evaluates the rules in that order and stops when it finds a rule that matches the lead.

When a rule wins, i.e., all the criteria are met for the lead, the engine stops evaluating channel rule sets. The lead is then assigned to the selected channel and the winning rule set is logged into a history table for analysis.

If more than one rule set with equal precedence win, the highest ranked channel is selected.

5.4.2 Setting Up Channels

Channels must be set up before channel selection rule sets are defined. Use this procedure to set up channels.

Prerequisites

None

Steps

1. Log in as an administrator, and navigate to Administration > Leads > Setup > Channel.

The Channel Setup page appears.

2. Select a channel from the Channel drop-down list.
3. Select the Indirect check box if this is an indirect channel.
4. Enter a number in the Precedence field, to set the precedence for this channel.

The precedence is used when the attributes of a lead match with more than one channel. In this case, the lead is assigned the channel with a higher precedence. For precedence, 100 is higher than 1.

5. Click **Update**.
6. Optionally, click **Restore** to clear a channel that you are setting up.

The channel is cleared.

7. Optionally, select the check box in the Remove column to remove a channel.
8. Click **Update**.

The channel is removed.

Note: When you create a rule set, the channels that you just set up may not appear in the drop-down lists. You must restart the apache to reload data from the database.

See Also

[Section 5.6, "Working with Rule Sets"](#)

5.5 Best Practices

Some best practices that you can use when you are working with the Leads Processing Engines are listed below:

- The Qualification Engine must not be used to both qualify and disqualify leads.
- If you use the Qualification Engine only to disqualify leads, it is recommended that you migrate your existing qualification rule sets to the Rating Engine. This ensures that the disqualified leads do not appear in sales representative queues.
- Create your channel selection rule sets in such a manner that all the disqualified leads are isolated and treated differently. A sales representative may work on them and if found suitable, change the Qualified flag to Yes.
- Create rule sets using the Campaign attribute. By doing so, leads generated from a specific campaign are processed by the specific rule sets. This helps you identify leads generated for a campaign, and you can track their progress in the sales pipeline.

For example, you want to track leads for campaign VisionVideos. When you create your rule sets, use Campaign as a guard and select this campaign. Hence, all leads that result from the VisionVideos campaign are processed by this rule set.

5.6 Working with Rule Sets

Rule Sets are mediums through which the Qualification, Rating, and Channel Selection engines process leads. The rule sets contain guards and rules which are

matched against the lead attributes. This section guides you to create rule sets, guards and rules for all the engines in the Leads Processing Engine.

Topics included in this section are:

- [Section 5.6.1, "Creating a Rule Set"](#)
- [Section 5.6.2, "Searching for a Rule Set"](#)

5.6.1 Creating a Rule Set

Use the following procedure to create a rule set for any of the engines- Qualification, Rating, or Channel Selection, and add guards and rules to it.

Prerequisites

None

Steps

1. Log in as an administrator, and navigate to Administration > Leads > Processing Rules > Qualification/Rating/Channel Selection.
2. Click **Create**.
3. Enter the Name and Description for the rule set.
4. In the Start Date and End Date fields, enter dates.
These are the dates between which the rule set is valid.
5. By default, the Status is Draft.
The status can be changed after the rule set is created.
6. Enter a number in the Precedence field.
Each rule set can have a different precedence, to define the order of importance for evaluation (where 100 is higher than 1).
7. Select the currency to be associated with this rule set from the Currency drop-down list.
8. In the Owner field, enter a name.
 - a. Click **Go**.
 - b. Select the owner to own this rule set.
 - c. Click **Create**.

Creating a Guard

9. In the Rule Set: Guard region, click **Add Attributes**.

Specifying Attributes

10. Select a profile from the Profile Attribute Name drop-down list.

The page reloads to display the values for the profile that you have chosen.

11. From the Condition drop-down list, select a condition for the profile.
12. Select a value(s) for the profile.
13. Click **Apply**.

The attribute is added.

14. Optionally, click **Apply And Add Another** to add another attribute.
15. Optionally, to modify an attribute, click the hyperlinked Profile Attribute Name.
16. Optionally, to remove an attribute, select it from the Remove column in the Rule Set: Guard region, and click **Update**.

You must now add rules to the rule set.

17. Click the Rules link.
18. Click **Add Rule**.

Creating a Rule

19. Enter a name for the rule, and specify its order of evaluation.
20. If this is a qualification rule set, select the Qualified Flag.
Select **Yes** if this rule set is to qualify leads, and **No** if it is to disqualify leads. The Qualified Flag attribute in the lead record will get this value.
21. If this is a rating rule set, select a rating from the drop-down list.
This rating is assigned to the lead if its attributes match the rule.
22. If this is a channel selection rule set, select a sales channel from the drop-down list.
This channel is assigned to the lead if its attributes match the rule.
23. Click **Add Attributes**.

To specify attributes for the rule, follow the procedure in the [Specifying Attributes](#) section.

24. To add another rule, click the Rules link, and follow the procedure in the [Creating a Rule](#) section.
25. Optionally, to modify an attribute, click the hyperlinked Profile Attribute Name.
26. Optionally, to remove an attribute, select it from the Remove column in the Rule Conditions region, and click **Update**.
27. Optionally, to make a copy of the rule set, click **Copy** in the Rule Set Details page.
Copy a rule set when you want another similar rule set with minor differences.

5.6.2 Searching for a Rule Set

You can search for any rule sets using the Search feature. You can search based on particular attributes used in the rule set.

Prerequisites

None

Steps

1. Log in as an administrator, and navigate to Administration > Leads > Processing Rules > Qualification/Rating/Channel Selection.
2. Enter the details of the rule set that you are looking for.
3. Click **Search**.
The results are displayed based on your search criteria.
4. Optionally, click **Clear** to clear the values in all the Search fields.

5.7 The Rule Flows Report

This Rule Flows report allows you to query across the Qualification, Rating, and Channel Selection engines for rule sets based on certain guard values.

This report supports multiple rule set groupings in the Rules Engine setup to track rule sets across processing flows. The rule sets are grouped based on the engine

type. For example, Qualification rule sets are displayed first, then Rating, followed by the Channel Selection rule sets.

Search By

- Date Range
- Guard values - Country, Campaign, Custom Setup, and Product Category
- Processing Stage
- Rule Set Status

Prerequisites

Rule sets must exist.

Steps

1. Log in as an administrator, and navigate to Administration > Leads > Processing Rules > Rule Flows.
2. Select a Country, Campaign, Custom Setup attribute, and a Product Category, as required.

These are set as guard values in the rule sets.

3. Select a date range between which the rule sets are valid.

This date range represents the start and end dates of the rule sets.

4. From the Processing Stage drop-down list, select the processing stage of the leads.

Processing Stage includes Qualification, Rating, and Channel Selection. For example, if Qualification is selected, only rule sets in the Qualification Engine are included in the report.

5. From the Status drop-down list, select the status of the engine that should be included in the report.

6. Click **Run Report**.

The report appears below displaying the guard values of each rule set.

7. Optionally, click the hyperlink in the Rule Set column to modify a rule set.

5.8 Purging Unqualified Leads

You can remove unqualified leads from the `AS_SALES_LEADS` table by running a concurrent program. Based on the following conditions, leads are deleted:

- The lead is unqualified, and it is not converted to opportunity
- The status of the lead is the same as the value in the OS: Default Status for Leads profile.

Unqualified leads that have been converted to opportunity will not be deleted by this program.

Note: After leads are purged from the system, any Trend reports set up in Oracle Daily Business Intelligence (DBI) or a custom application will be affected.

Use the following details to run the Purge Unqualified Sales Leads concurrent program.

Prerequisite: None

Responsibility: Oracle Sales Administrator

Parameters:

- From and To Date - Enter a date range indicating the leads created during this period.
- Debug - Enter *Yes* if you want to see debug messages.
- SQL Trace - Enter *Yes* if you want to trace SQL

Schedule: Once

See Section [Section C.1, "Running Concurrent Programs"](#) for the steps to run the concurrent program.

5.9 Setting Up Lead Assignments

Various system profiles can be set up to assign resources to leads based on the Leads Processing Engine results. Depending on the requirements in your organization, you can also define custom functions to route leads to appropriate resources.

Topics in this section include:

- [Section 5.9.1, "Setting Up Automatic Lead Assignment"](#)
- [Section 5.9.2, "Setting Up Immature Lead Assignment"](#)
- [Section 5.9.3, "Routing Leads Using a User Hook"](#)

5.9.1 Setting Up Automatic Lead Assignment

You can set up the application to automatically assign resources to a lead whenever an agent or salesperson creates or updates the lead. This is achieved by assigning values to selected profiles.

- **OS: Assign New Lead**

To enable automatic lead assignment, set the value of this profile to **N**. This is the default value. When this profile is set to **N**, a call to the Territory Manager API automatically assigns resources to the lead using the territories defined in Territory Manager. The first person the program assigns becomes the lead owner. The rest of the resources in the territory become sales team members on the lead.

If the lead creator is a valid sales agent or salesperson, the lead creator is added to the lead sales team when the lead is created.

- **OS: Auto Convert Lead to Opportunity**

If this profile is set to **Y**, an opportunity is created for all qualified indirect leads, and the partner matching workflow is launched.

- **OS: Lead Incubation Channel**

All immature leads are assigned to a particular channel. The channel is decided by the value in this profile. The lead owner of an immature lead is determined by immature lead assignment. See [Section 5.9.2, "Setting Up Immature Lead Assignment"](#) for more information.

- **OS: Default Resource ID Used for Sales Lead Assignment**

Set this profile to a resource who will handle any lead that is not matched with any territory. If this profile is not set, the lead is assigned to the agent or salesperson who created or updated the lead.

Note: If both the resource in OS: Default Resource ID Used for Sales Lead Assignment and the user who created or updated the lead do not have a valid sales role assigned to them, then the leads you import will not be accessible from either Oracle Sales Online or Oracle TeleSales.

- **OS: Calendar Setting Required for Assignment**

Set this profile to *Yes* if the territories in your organization use agent availability as one of the criteria for assigning agents. This enables the automatic assignment of lead owners based on availability. This profile is set to *No* by default.

Apart from setting this profile, you must also make sure that each resource has a calendar set up for them.

For more details, see the *Oracle CRM Application Foundation Implementation Guide*.

See [Section A.1, "Setting System Profile Options"](#) to set values to these profiles.

5.9.2 Setting Up Immature Lead Assignment

An immature lead is a lead that is not yet ready for a sales representative to spend time on. It is a low quality and low grade lead that needs to be matured by the marketing team before it is assigned to a sales team.

The Channel Selection engine assigns all immature leads to a specific channel, such as an Immature channel. This channel is decided by the value in the OS: Incubation Channel profile. When a lead is assigned to the immature channel, the owner is decided by the value in the OS: Default Lead Marketing Owner profile.

If the OS: Default Lead Marketing Owner profile is not set, the Territory Assignment program assigns all immature leads to the resources identified to act on immature leads.

The Maturation Assignment page in the Administration > Leads tab can be used to provide information about the territory assignment setup.

Note: The Maturation Assignment page will be obsolete from the next release. Use the Sales Channel to group the immature leads, and the Territory Assignment program will assign them to an appropriate resource based on the Channel Qualifier.

Use this procedure to add one or more resources to manage immature leads.

Prerequisites

None

Steps

1. Log in as an administrator, and navigate to Administration > Leads > Processing Rules > Maturation Assignment.

The Immature Lead Assignment page appears.

2. In the Add a resource field, enter the name of the resource.

You can use the % wildcard, if required.

3. Click **Go**.

The Find a Sales Person page appears listing sales persons matching your search criteria.

4. Select one or more sales persons.

5. Click **Select**.

The Immature Lead Assignment page appears, and the sales person(s) is added.

6. Select the Owner check box to indicate that a sales person must be the owner of the lead assigned.

7. From the Country drop-down list, select the country for this sales person.

8. In the Postal Code From and Postal Code To fields, enter the postal from and to codes.

9. Optionally, to remove a sales person, select the Remove check box.

10. Click **Update**.

5.9.3 Routing Leads Using a User Hook

You can implement custom rules for lead assignment by implementing the Lead Routing Engine user hook.

Hook Name: AS_LEAD_ROUTING_WF

Package Name: AS_LEAD_ROUTING_WF_CUHK

Purpose

If you are implementing custom lead routing rules, then create a package body according to these specifications.

Note: Do not commit in this package body. After the transaction is complete, Oracle application code will issue a commit.

This user hook will be called when an agent or salesperson is creating and updating a lead in the Lead tab, and from the Import Sales Lead concurrent program whenever the routing engine is called.

The calling package is AS_LEAD_ROUTING_WF.GetOwner.

API name

Get_Owner_Pre

Procedure Specification

```
PROCEDURE Get_Owner_Pre(
    p_api_version_number    IN NUMBER,
    p_init_msg_list         IN VARCHAR2 := FND_API.G_FALSE,
    p_validation_level      IN VARCHAR2 := FND_API.G_VALID_LEVEL_FULL,
    p_commit                IN VARCHAR2 := FND_API.G_FALSE,
    p_resource_id_tbl       IN AS_LEAD_ROUTING_WF.NUMBER_TABLE,
    p_group_id_tbl          IN AS_LEAD_ROUTING_WF.NUMBER_TABLE,
    p_person_id_tbl         IN AS_LEAD_ROUTING_WF.NUMBER_TABLE,
    p_resource_flag_tbl     IN AS_LEAD_ROUTING_WF.FLAG_TABLE,
    p_sales_lead_rec        IN AS_SALES_LEADS_PUB.SALES_LEAD_Rec_Type,
    x_resource_id           OUT NUMBER,
    x_group_id              OUT NUMBER,
    x_person_id             OUT NUMBER,
    x_return_status         OUT VARCHAR2,
    x_msg_count             OUT NUMBER,
    x_msg_data              OUT VARCHAR2
```

```
)
IS

    l_resource_count          NUMBER;
BEGIN
    -- Standard Start of API savepoint
    SAVEPOINT GET_OWNER_PRE_PVT;

    -- Standard call to check for call compatibility.
    IF NOT FND_API.Compatible_API_Call ( l_api_version_number,
                                         p_api_version_number,
                                         l_api_name,
                                         G_PKG_NAME)
    THEN
        RAISE FND_API.G_EXC_UNEXPECTED_ERROR;
    END IF;

    -- Initialize message list IF p_init_msg_list is set to TRUE.
    IF FND_API.to_Boolean( p_init_msg_list )
    THEN
        FND_MSG_PUB.initialize;
    END IF;

    -- Initialize API return status to SUCCESS
    x_return_status := FND_API.G_RET_STS_SUCCESS;

    -- Api body

    l_resource_count := p_resource_id_tbl.count;
    IF l_resource_count > 0
    THEN
        x_resource_id := p_resource_id_tbl(1);
        x_group_id := p_group_id_tbl(1);
        x_person_id := p_person_id_tbl(1);
    ELSE
        x_resource_id := NULL;
    END IF;

    -- END of API body

    -- Standard check for p_commit
    IF FND_API.to_Boolean( p_commit )
    THEN
        COMMIT WORK;
    END IF;
```

```

-- Standard call to get message count and IF count is 1,get message info
FND_MSG_PUB.Count_And_Get
( p_count          => x_msg_count,
  p_data           => x_msg_data );

END Get_Owner_Pre;

END AS_LEAD_ROUTING_WF_CUHK;

```

In Parameters

[Table 5–1](#) lists the standard input parameters.

Table 5–1 Standard Input parameters

Parameter	Description
p_api_version_number	For 11i Oracle Sales applications, this is set to 2.0.
p_init_msg_list	Should the message stack be initialized? By default, this is set to FND_API.G_FALSE.
p_validation_level	The validation level of the pass-in value. By default, this is set to FND_API.G_VALID_LEVEL_FULL.
p_commit	Should a commit be issued for the whole API at the end? By default, this is set to FND_API.G_FALSE.

The following three parameters store the available resources for this customized package to decide the owner of the sales lead. Their data type is TABLE of NUMBERS.

- p_resource_id_tbl
- p_group_id_tbl
- p_person_id_tbl

[Table 5–2](#) lists other input parameters.

Table 5–2 Other Input Parameters

Parameter	Description
p_resource_flag_tbl	<p>This parameter specifies the source of the resource:</p> <ul style="list-style-type: none"> ■ D: Default resource from the OS: Default Resource ID used for Sales Lead Assignment profile. ■ L: Login user. ■ T: Territory definition. <p>If the sales lead matches any territory, the above parameters will include all the resources returned from the territory engine and the p_resource_flag_tbl will be all T.</p> <ol style="list-style-type: none"> 1. If the sales lead does not match any territory, and the OS: Default Resource ID used for Sales Lead Assignment profile is set: <ul style="list-style-type: none"> ■ p_resource_id_tbl(1), p_group_id_tbl(1), p_person_id_tbl(1) is the default resource defined in this profile. ■ p_resource_flag_tbl(1)=D ■ p_resource_id_tbl(2), p_group_id_tbl(2), p_person_id_tbl(2)=L. ■ p_resource_flag_tbl(2)=L 2. If the sales lead does not match any territory, and the OS: Default Resource ID used for Sales Lead Assignment profile is not set: <ul style="list-style-type: none"> ■ p_resource_id_tbl(1), p_group_id_tbl(1) ■ p_person_id_tbl(1)=L ■ p_resource_flag_tbl(1)=L
p_sales_lead_rec	This provides the whole definition of a sales lead. This record is provided to help decide the sales lead owner.

Out Parameters

The following three parameters store the result of this user hook:

- x_resource_id
- x_group_id
- x_person_id

Together, they set the sales lead owner.

If x_resource_id is NULL, the owner is decided based upon Oracle's logic.

For example, `x_resource_id=1001`, `x_group_id=10`, `x_person_id=100`. The resource with the resource ID 1001, group ID 10, and person ID 100 will be assigned as the owner of the sales lead.

Table 5–3 lists the standard output parameters.

Table 5–3 Standard Output Parameters

Parameter	Definition
<code>x_return_status</code>	The return status. If your code completes successfully, then <code>FND_API.G_RET_STS_SUCCESS</code> must be returned. If you get an expected error, then return <code>FND_API.G_RET_STS_ERROR</code> , otherwise return <code>FND_API.G_RET_STS_UNEXP_ERROR</code> .
<code>x_msg_count</code>	The message count. Call <code>FND_MSG_PUB.Count_And_Get</code> to get the message count and messages.
<code>x_msg_data</code>	The messages. Call <code>FND_MSG_PUB.Count_And_Get</code> to get the message count and messages.

5.10 Setting Up Lead Status

Some lead statuses are seeded in the application. They are New, In Progress, Converted to Opportunity, Dead Lead, and Loss.

Use the following procedure to define alternate statuses.

Prerequisites

None

Steps

1. Log in as an administrator, and navigate to Administration > Sales > Opportunity > Status Code.

The Status Code page appears.

2. Click **Create**.
3. In the Status Code field, enter the status code name.
You cannot see this name. It is for internal use only.

4. In the Meaning field, enter the meaning.
Users see this in the drop-down list when they select a lead status.
5. Add an optional description for the status.
6. Select the **Enabled** check box to enable this status.

Note: Do not select either Open or Include in Forecast check boxes. These are reserved for setting up opportunity statuses.

7. In the Used for region, select the Sales Lead box.
8. Select Neither in the Win Loss Indicator region.
The other options are reserved for opportunity statuses.
9. Click **Create**.

5.11 Using Custom Attributes

Apart from the seeded attributes, you can add custom attributes to meet your specific requirements. For example, you may want to set up rule sets based on a complex business logic, and the seeded attributes do not meet the requirements completely. For seeded attributes in Oracle Leads Management, see [Section B.1, "Seeded Attributes"](#).

Use the following procedure to create custom attributes.

Prerequisites

None

Steps

1. Log in as an administrator, and navigate to Administration > Leads > Setup > Custom Attributes.
2. Click **Create**.
3. Select a value type for the attribute from the Value Type drop-down list.
4. Enter a name and description for the new attribute.
5. Click the Active option to activate the attribute.
6. Click **Create**.

Use these attributes while creating a rule set.

5.12 Setting Up Time Frames

Time frames determine the expiration date of a lead. The expiration date of a lead is the maximum length of time frame relative to the creation date. You can assign number of days to a time frame.

Some seeded examples:

Within 1 week : 7 days

1-3 months : 90 days

For all seeded values, see [Appendix B, "Seeded Data"](#).

5.12.1 Customizing Time Frames

The time frame periods can be customized to suit your requirements. However, the time frame itself cannot be modified, and new ones cannot be created. You must enable the time frames that you want to use in your organization.

Prerequisites

None

Steps

1. Log in as an administrator, and navigate to Administration > Leads > Setup > Timeframe.
2. From the Time Frame column, select the time frame that you want to customize.
3. In the Days column, enter the new period.

4. Click **Update**.

The time frame is added with the new period. There may be multiple periods defined for the same time frame.

5. Select the check box in the Enable column to enable this time frame.

Note: Similar time frames cannot be enabled at the same time. Deselect any, if enabled.

6. Select a time frame from the Remove column to remove it.

Note: Seeded time frames cannot be removed.

7. Click **Update**.
8. Click **Restore** to clear a time frame that you are customizing.
The time frame is cleared.

6

Monitoring Leads

This chapter gives you information about the Monitoring Engine.

Topics included are:

- [Section 6.1, "Monitoring Engine Overview"](#)
- [Section 6.2, "Running the Workflow Background Process Concurrent Program"](#)
- [Section 6.3, "Creating a Monitor Rule"](#)
- [Section 6.4, "Viewing the Monitor Log"](#)
- [Section 6.5, "Searching for a Monitor"](#)

6.1 Monitoring Engine Overview

The Monitoring Engine enables the administrator to track the state of a lead and ensure that each lead is acted upon in a timely manner. When a lead has not been acted upon, the monitoring engine can be set up to send reminders to the owner, or reroute the lead.

The lead monitor is set to trigger on two conditions:

- When a lead is created (See Creation Date Monitors)
- When a lead is assigned to a sales team (See Assignment State Monitors)

When a lead satisfies one of the above conditions, a monitor is selected for the lead based on its scope. Based on the conditions specified, notification(s) or reminder(s) are sent to the lead owner and/or the owner's manager. If the lead is still not acted upon, it may be timed-out and rerouted.

A monitor is active on a lead till the lead reaches the To condition set for that monitor. When the monitors are evaluated again after changes in some lead attributes, the monitors are either stopped or continue to run the course. They are stopped if the lead's attributes no longer match the monitor's scope. If the new state of the lead satisfies another monitor, it is reassigned.

Creation Date Monitors

Every time a new lead is created, the monitoring engine is called. If the lead matches a Creation Date (Creation From State) monitor, the lead is attached to this monitor. For this lead, the monitors defined for Assignment From State are never used. The creation date monitors are picked up only once for a single lead.

The Creation Date monitors are also called Absolute Monitors.

Assignment State Monitors

If the lead does not match any of the Creation Date monitors, then a matched Assignment State (Assignment From State) monitor is chosen for the lead. Every time this lead is reassigned, the monitoring engine is called. At this time, any old Assignment State monitors that may be active on the lead are terminated, and the workflow for the new monitor is launched. If a new Assignment State monitor is not available for the new state, then no action is taken.

The Assignment State monitors are also called Relative Monitors because they are evaluated every time some lead attributes such as Lead Owner and Lead Rating change.

Behavior of the Monitoring Engine

Table 6–1 elucidates the behavior of the monitoring engine depending on the change in the lead attributes.

Table 6–1 Monitoring Engine Behavior

Event	Creation Date	Assignment State
Change in Lead Rank	Monitors are reevaluated. If the current monitor is no longer valid, it is stopped. No new Creation Date monitors are assigned. This lead is never picked up by an Assignment State monitor.	Monitors are reevaluated. If the current monitor is no longer valid, it is stopped. A new Assignment State monitor that satisfies the lead attributes is assigned.
Change in Lead Owner	Creation Date monitors are not evaluated.	Monitors are reevaluated. If the current monitor is no longer valid, it is stopped. A new Assignment State monitor that satisfies the lead attributes is assigned.

Topics in this section include:

- [Section 6.1.1, "Scope of a Monitor"](#)
- [Section 6.1.2, "Smart Time Frames and Expiration Dates"](#)
- [Section 6.1.3, "Timeouts in the Monitoring Engine"](#)

6.1.1 Scope of a Monitor

The scope of a monitor is defined by the Country and Rating attributes. A monitor can have either one or both these attributes defined for it. A lead satisfying the attributes is processed by the appropriate monitor. For example, if the scope of a monitor is Country=*US* and Rating=*A*, all leads that belong to the *US* and have an *A* rating are processed by this monitor.

When no monitors satisfy the lead attributes, the lead is not monitored.

Note: There is an implicit AND between the Country and Rating attributes, if both are selected.

6.1.1.1 Upgrading from Release 11.5.9

In release 11.5.9, if both the Country and Rating attributes were defined for a monitor rule, a lead satisfying either of the attributes were picked for monitoring. However, from 11.5.10 onwards, if both the attributes are defined for the monitor rule, a lead must satisfy both to be monitored.

For example, a lead named Allen has the rating A. Only one monitoring rule set has been defined as follows:

Rule 1

Country = US

Rating = A

In release 11.5.9, lead Allen was monitored by Rule 1. In 11.5.10, lead Allen will not be monitored at all.

Migration Steps

If you want the above sample rule to work the same way as it worked in release 11.5.9, you must define additional rules as follows:

Rule 2

Country = US

Rule 3

Rating = A

In the example, lead Allen will now be monitored by Rule 3. Rule 1 can remain as is serving the And condition. If a lead belongs to the US and is rated A, then Rule 1 (which matches the lead best) is used to monitor the lead.

However, it should be remembered that Creation Date monitors always take precedence over the Assignment State monitors.

6.1.2 Smart Time Frames and Expiration Dates

Smart time frames determine the expiration date of a lead. For more information about smart time frames, see [Section 5.12, "Setting Up Time Frames"](#).

The Monitoring Engine uses smart time frames through the Relative to Expiration Date check box. If the check box is selected, and the expiration date is set for the lead, the owner, or owner's manager, or both, are notified *n* days before the expiration date of the lead. If the check box is selected, and the expiration date is not set for the lead, the monitor condition will never be satisfied by the lead.

6.1.3 Timeouts in the Monitoring Engine

The concept of a timeout in the monitoring engine is the mechanism to make sure that leads are followed up within a certain period of time. If a timeout in days is mentioned in the monitor rule set, then the lead is reassigned to a new owner from the sales team after the elapse of the timeout days unless the rule set condition is reached.

Consider the following scenarios to understand timeouts in the monitoring engine.

Monitor 1

Monitor 1 has been defined in the application with the following conditions:

Monitor 1 Rule Conditions

Condition	Value Set
From	Creation
To	Accepted
Timelag	1 day
Relative to Expiration	N
Reminders	To Lead Owner
Timeout	2 days

A new lead NewTech Inc. is created that satisfies Monitor 1. The monitor is started as and when the lead is created. The lead owner has one day to accept the lead. After one day, the first notification goes to the lead owner stating that the lead must be accepted. Two days after the monitor was started i.e. one day after the first notification was sent, the monitor times out and the lead gets reassigned to the new owner.

Monitor 2

Monitor 2 has been defined in the application with the following conditions:

Monitor 2 Rule Conditions

Condition	Value Set
From	Creation
To	Accepted

Monitor 2 Rule Conditions

Condition	Value Set
Timelag	1 day
Relative to Expiration	Y
Reminders	None
Timeout	2 days

Note: In this example, the Relative to Expiration condition is set to Y. When a lead is created, it is assigned an expiration date based on the Timeframe attribute.

A new lead HiBiz Corp. is created that satisfies Monitor 2. The monitor is started as and when the lead is created. The lead owner has one day to accept the lead. In this case, technically the first notification will be sent only one day prior to the expiration date of the lead. However, the timeout will still happen two days after the lead is created. This means that if the lead were due to expire in 2 weeks time, the first notification will be sent one day before the two weeks are over. But that would never happen because the monitor would always timeout in two days flat.

Timeouts are valid only for very important leads where an opportunity might be lost if not followed up within a certain time period. For the HiBiz Corp. lead, the timeout is not needed because the objective is to follow up on the lead before it expires.

6.2 Running the Workflow Background Process Concurrent Program

The Workflow Background Process concurrent program sends notifications and reminders from triggered monitors. You must schedule the concurrent program to run every day, or twice a day, if required.

Use the following details to run the Workflow Background Process concurrent program.

Prerequisite: None

Responsibility: System Administrator

Parameters:

- Item Type - Monitoring Engine Workflow

- Process Deferred - Y
- Process Timeout - Y
- Process Stuck - Y

Schedule: Once

For the steps to run the concurrent program, see [Section C.1, "Running Concurrent Programs"](#).

6.3 Creating a Monitor Rule

Use the following procedure to set up one or more monitoring rules for lead monitoring.

Prerequisites

Make sure the system profile `OS : Max_Lead_Reroutes` is set. The number assigned to this profile defines the maximum number of reroutes allowed to a lead that is timed out by the monitoring engine. If the lead does not have a owner after the maximum reroutes, it is routed to the default resource defined by the `AS_DEFAULT_RESOURCE_ID` profile.

Steps

1. Log in as an administrator, and navigate to Administration > Leads > Processing Rules > Monitoring Rules.
2. Click **Create**.
3. Enter the details in the Monitoring Rule Name and Description fields.
4. In the Valid From and To Date fields, enter dates.
These are the dates between which the rule is valid.
5. From the Status drop-down list, select a status.
The rule will be effective only in the Active status.
6. In the Owner field, enter a name.
 - a. Click **Go**.
 - b. Select the owner to own this rule.
7. In the Monitor Conditions region, enter the conditions for the monitor.

See [Section 6.3.1, "Monitoring Engine Conditions"](#).

8. In the Notification Recipients region, select the Notify Owner and/or the Notify Owner Manager check box.
9. In the Reminders region:
 - a. Select the No Reminder option.
OR
 - b. Select the Send number of Reminders option.
 - In the Number of Reminders field, enter a number.
This indicates the number of reminders to send.
 - In the Frequency...Days field, enter a number.
This indicates the frequency in which to send the reminders.
10. In the Timeout region:
 - a. Select the No Timeout option.
OR
 - b. Select the Timeout after the following period option.
 - In the Days field, enter the number of days.
This indicates the number of days after which the lead is timed out.
11. Click **Create**.

The monitor rule is created. Next, you must define the scope for this monitor.
12. In the Monitor Scope region, click **Add Attributes**.
13. From the Profile Attribute Name drop-down list, select a profile.

The page reloads to display the values for the profile that you have chosen.
14. From the Condition drop-down list, select a condition for the profile.
15. Select a value(s) for the profile.
16. Click **Apply**.

The attribute is added.
17. Optionally, click **Apply And Add Another** to add another attribute.

18. To modify an attribute, click the hyperlinked Profile Attribute Name in the Add Guard/Rule Profile Attribute Name page.
19. To remove an attribute, select it from the Remove column in the Add Guard/Rule Profile Attribute Name page, and click **Update**.

The attribute is removed.

Note: For a complete list of supported conditions, see [Section 6.3.1, "Monitoring Engine Conditions"](#).

6.3.1 Monitoring Engine Conditions

Table 6–2 lists the monitor conditions with their descriptions.

Table 6–2 *Monitor Engine Conditions*

Condition	Description
From State	The state of the lead (whether it was created or assigned) when it first qualifies for a monitor condition. The values supported for the From State are Creation and Assignment.
To State	The state of the lead till when the monitor conditions are applicable. The states supported are Accepted, In Progress, Closed, and Updated.
Time Lag Limit	Number of days when the first notification must be sent after the From State changes.
Relative to Expiration Date check box	Three possibilities exist: <ul style="list-style-type: none"> ■ If selected, and the expiration date is set for the lead, a notification is sent n days before the expiration date of the lead, where $n = \text{Time Lag Limit}$. ■ If selected, and the expiration date is not set for the lead, the monitor condition will never be satisfied by the lead. ■ If not selected, the first notification is sent n days after the lead creation date or the lead assignment date, where $n = \text{Time Lag Limit}$.
Total Reminders	Total number of reminders to be sent after the first notification.
Frequency	The number of days between two reminders, and between the notification and the first reminder.

Table 6–2 Monitor Engine Conditions

Condition	Description
Timeout Number	The number of days after the first notification when the lead must be timed out if it still satisfies the monitor condition.

6.4 Viewing the Monitor Log

The monitor log keeps track of all the leads that were processed by a monitor. For each monitor, you can view details such as the monitor triggered date, the notification and reminder recipient(s), and the status of a lead that triggered the monitor.

You can search for a specific log by specifying the name of a monitor, a valid date range, the status of a monitor, and the name and ID details of a lead.

Prerequisites

None

Steps

1. Log in as an administrator, and navigate to Administration > Leads > Setup > Monitoring Rules.
2. From the Reports column, click the Reports icon for the monitor whose log you wish to view.

The Monitor Log page appears.

3. Specify the details of the monitor log to view.

The Valid Date range indicates the dates between which the monitor should have been triggered.

4. Click **Run Report**.

The monitor log details that satisfy your search criteria appear in the Results region.

6.5 Searching for a Monitor

You can search for a monitor using the following attributes - name, owner, status, rule set valid date, customer category, and country.

Prerequisites

None

Steps

1. Log in as an administrator, and navigate to Administration > Leads > Processing Rules > Monitoring.
2. Enter the details in the Name and Owner fields.
3. From the Status drop-down list, select the status of the monitor.
4. In the Start Date and End Date fields, select the dates between which the monitor is valid.
5. From the Lead Rating drop-down list, select the rating of the lead for the monitor that you are searching for.
6. From the Country drop-down list, select the country.
7. Click **Search**.

The results are displayed based on your search criteria.

Operational Reports

This chapter explains the various operational reports in Oracle Leads Management.

Topics included are:

- [Section 7.1, "Imports Records Manager Reports"](#)
- [Section 7.2, "Lead Processing History Reports"](#)
- [Section 7.3, "Rule Performance Reports"](#)
- [Section 7.4, "Rule Diagnostics Reports"](#)
- [Section 7.5, "Lead Linking Report"](#)

7.1 Imports Records Manager Reports

The Import Records Manager report enables you to search all records imported as leads, to find import exceptions and recover from errors, and to track the number of leads imported by different sources.

To resolve errors resulting from an import operation, see [Section 3.3.3, "Correcting Errors During Lead Import"](#).

Prerequisites

None

Steps

1. Log in as an administrator, and navigate to Administration > Leads > Operational Reports > Import Records Manager.
2. From the Source System drop-down list, and select the source application from which the leads were generated.
3. In the Promotion ID field, enter the Source Code of the marketing campaign or campaign schedule.
4. In the Import Interface ID field, enter the import interface ID that was assigned to the lead during import.
5. In the Batch ID field, enter the batch ID of the imported leads.

You may have assigned a batch number to identify similar leads during lead import.

6. Enter other details, as required.
7. Click **Search**.

The report appears below.

7.2 Lead Processing History Reports

The Lead Processing History report provides the ability to view details of each execution of the Rules Engine per lead, and the state of the lead at different executions. Details such as the date and time, winning rule, and the outcome of each engine run for each lead is displayed.

Search By

Processing Stage (including all), Lead Name, Lead ID, Rule Set Name, Guard fields, Creation and Evaluation dates, and Result.

Group By

Unique Lead ID and Lead Name. All sorting is within a lead.

Note: If the value in the OS: Default Rating for Lead Rating Engine profile is set to BLANK, lead records that get this as their default rating will not be displayed by this report.

Use the following procedure to run the report.

Prerequisites

None

Steps

1. Log in as an administrator, and navigate to Administration > Leads > Operational Reports > Lead Processing History.
2. From the Processing Stage drop-down list, select the processing stage of the leads - qualification, rating, or channel selection.
3. In the Lead Number field, enter the number assigned to the lead to see its processing history.
Leave this field blank if you do not wish to view the report for a specific lead.
4. In the Rule Set field, enter the name of the rule set if you wish to view the report based on a specific rule set.
5. In the Lead Name field, enter the name of the lead to see its processing history.
Leave this field blank if you do not wish to view the report for a specific lead.
6. From the Rule Result drop-down list, select the rule result.
The report displays processing history only for the selected rule result.
7. From the Lead Status drop-down list, select the status of the lead.
The report displays processing history only for the selected status.

8. Enter the lead creation date range for the period when the details must be displayed.
9. From the Product Category drop-down list, select a product category for which the leads were processed.
10. From the Country drop-down list, select the customer's country for whom the leads were processed.
11. In the Source Name field, enter the name of the campaign from which the leads were generated.
12. Click **Run Report**.

The report appears below.

7.3 Rule Performance Reports

The Rule Performance report provides the ability to view utilization and effectiveness of rule sets (for optimization of rule configuration). This stores the last engine run for each lead, the date and time, the winning rule, and the outcome. Order and revenue tracking enables you to correlate the ranking predictability of leads in relation to actual performance.

Search By

Processing stage (including all), Lead Name and Lead ID, Rule Set Name, Guard fields, Creation and Evaluation dates, and Result.

Group By

Stage, Rule Set, and Result, in that sequence.

Summary Reports

- Utilization - Total number of leads processed (matching criteria)
- Total Number of Orders
- Order amount
- Revenue in \$
- Effectiveness
 - Total number and % of total of Status
 - Total number and % total Upgraded or Downgraded

- Total number and % of total Accepted.

For failed leads which use the default resource, the report displays the lead outcome (for example, the rating) with no rule applicable. You can drill down to view the winning sub-rule for each rule set and review the criteria.

Details

To view a detailed report, click the Details link after the Summary report displays. For the detailed report, you must enter a value for one the following fields: Lead Number, Lead Name, or Lead Status.

See the procedure in [Section 7.2, "Lead Processing History Reports"](#) for the steps to run the report.

7.4 Rule Diagnostics Reports

The Rule Diagnostics report provides the ability to troubleshoot and manage rule configuration in the Leads Engine. This stores details of each engine run for each lead, and details include failure, tied, and winning executions of the rules and their precedence and guards for setup analysis.

Search By

Processing stage (including all), Lead Name and Lead ID, Rule Set Name, Guard fields, Creation and evaluation dates, and Result.

Group By

Stage, then Rule Set.

Summary Reports

- Total number of executions where leads matched
- Total number of executions and % Failed
- Total number of executions and % Passed
- Total number of executions and % Tied
- Total number of executions and % Won

Each report shows record-by-record details of the executions.

Details

To view a detailed report, click the Details link after the Summary report displays.

See the procedure in [Section 7.2, "Lead Processing History Reports"](#) for the steps to run the report.

7.5 Lead Linking Report

The Lead Linking report enables you to assess the effectiveness of the Interaction rule sets. The report displays the leads linked against standard guards used for interaction selections. The chief goal of these reports is to help you to optimize the logic of which interactions to capture and to set the thresholds for sales follow up.

You can view how many interactions are captured in a given time frame. The report also displays the type of interactions that are captured and how many leads are linked to those interactions. The percentage of interactions linked is also displayed.

Note: Only customer-related interactions are considered while generating this report.

Search By

Date range, Country, and Source Code.

Group By

Interaction Type, Campaign

Summary Reports

- Number of interactions captured between the dates given by the filter parameters.
- Number of interactions linked.
- Percentage of interactions linked.
- Number of leads linked to interactions.
- Leads converted and percentage of leads linked

Prerequisites

Interaction rule sets must exist.

Steps

1. Log in as an administrator, and navigate to Administration > Leads > Operational Reports > Lead Linking.
2. Enter the interaction creation date range for the period when the details must be displayed.
3. To select a Country, click the search icon.
This is the customer's country recorded in the interaction.
4. To select a Source Code, click the search icon.
The source code is the code of the campaign from which the interactions were generated.
5. From the Group By drop-down list, select the attribute based on which the report must be grouped.
6. Click **Run Report**.
The report appears below.

A

System Profile Options

This appendix provides a table of all of the system profile options used by Oracle Leads Management. Each system profile option in this table is also described in the implementation step where it is set. This table compiles all system profile options and arranges them in alphabetical order by name.

Topics included are:

- [Section A.1, "Setting System Profile Options"](#)
- [Section A.2, "System Profile Options in Oracle Leads Management"](#)

A.1 Setting System Profile Options

The procedure for setting up and changing system profile options is the same for all Oracle applications. For a detailed description of the procedures, refer to the *Oracle Applications System Administrator's Guide*.

Use this procedure to set or change profile options.

Prerequisites

None

Steps

1. Log into Oracle Forms as a System Administrator, and navigate to Profile > System.
2. In the Find System Profile Values window, enter your search criteria in the Display region.
3. Click in the **Profile** Field. Enter a partial name of the profile using "%" as a wild card.
4. Click **Find**.

The profiles are displayed in the System Profile Values window.

5. Set at least one of the following:

- a. Set the Site (S) value.

This field displays the current value, if set, for all users at the installation site.

- b. Set the Application (A) value.

This field displays the current value, if set, for all users working under responsibilities owned by the application identified in the Find Profile Values block.

- c. Set the Responsibility (R) value.

This field displays the current value, if set, for all users working under the responsibility identified in the Find Profile Values block.

- d. Set the User (U) value.

This field displays the current value, if set, for the application user identified in the Find Profile Values block.

6. Click in the field of the profile you want to set or change.
7. Select a value from the List of Values (LOV).

Click **Save** on the toolbar.

A.2 System Profile Options in Oracle Leads Management

The system profile options used by Oracle Leads Management has the following prefixes:

- OS: Oracle Sales
- OSO: Oracle Field Sales
- PV: Oracle Partner Management
- OTS: Oracle TeleSales

The following tables list the system profiles used by Oracle Leads Management by category name and in alphabetical order . The tables include the following columns from left to right:

- **Req?** (Required): "Y" in this column indicates the profile is required for the applications or functions.
- **New?** - "Y" in this column indicates that the profile is new for this release.
- **Profile Name:** Name of the profile.
- **Description:** Explains what the profile does.
- **Program:** Lists the functional areas affected by the profile.
- **Level:** Level at which this profile option can be set. A = Application, S = Site, R = Responsibility, U = User.
- **Default:** Lists the default value for the profile, if any.

The following are the list of tables with profiles grouped by their category.

- [Table A-1, "Default Profiles"](#)
- [Table A-2, "Profiles for Lead Import"](#)
- [Table A-3, "Profiles for the Leads Interaction Matching Engine"](#)
- [Table A-4, "Profiles for the Leads Rule Engines"](#)
- [Table A-5, "Profiles for Leads Assignment"](#)

- [Table A-6, "Profiles for Leads Followup"](#)
- [Table A-7, "Profiles for Leads Conversion"](#)
- [Table A-8, "Profiles for Leads Security"](#)
- [Table A-9, "Leads Profiles in Oracle TeleSales"](#)
- [Table A-10, "Lead Profiles Options for Multiple Currencies"](#)
- [Table A-11, "Obsolete Profiles"](#)

[Table A-1](#) lists the default profiles in Oracle Leads Management.

Table A-1 Default Profiles

Req.?	New?	Profile Name	Description	Program	Level	Default
Y	N	OS: Sales Lead Default Close Reason	The Update Sales Lead API assigns statuses to leads based on the value in the OS: Lead Link Status profile. If the status in the profile is 'Closed', the API picks up the value in this profile.	Leads	S	-
N	N	OS: Default Decision Timeframe for Leads	If the lead record does not have a timeframe value, the value in this profile is set. If this profile is not set, a NULL value is inserted.	Leads	S, R, U	Within 1 week
N	N	OS: Default Lead Contact Role	Used for defaulting the contact role on the contact page.	Leads	S,R,U, A	END_USER
Y	N	OS: Default Status for Leads	Used as the default status for a lead in the UI and API. If not set, then a NULL value is inserted, and an error notification is displayed.	Leads	S, R, U	New
N	N	OS: Default Vehicle Response Code for Leads	Used as the default value in the UI and API. If not set, then a NULL value is inserted in the database.	Leads	S, R, U	User
Y	N	OSO: Default Country	Used for defaulting the country selected on the Create Lead page. If not set, the first country in the drop-down list will be chosen by default.	Leads	S, R, A, U	United States
N	N	OSO: Default Lead Contact Title	Sets the title for the lead contact. For example, Mr., Ms., and so on.	Leads	S,A,R, U	Ms.

Table A-2 lists the profiles related to Lead Import.

Table A-2 Profiles for Lead Import

Req.?	New?	Profile Name	Description	Program	Level	Default
Y	N	OS: Use DQM Rule code to match Party	Used during Lead Import. The rule associated with this profile decides whether the imported record is a duplicate by checking for a matching Party in the TCA database.	DQM	A,S,R, - U	
Y	N	OS: Use DQM Rule code to match Contact	Used during Lead Import. The rule associated with this profile decides whether the imported record is a duplicate by checking for a matching Contact in the TCA database.	DQM	A,S,R, - U	
Y	N	OS: Use DQM Rule code to match Person	Used during Lead Import. The rule associated with this profile decides whether the imported record is a duplicate by checking for a matching Person in the TCA database.	DQM	A,S,R, - U	
Y	Y	OS: Execute Custom Code from Lead Import	Controls the execution of the custom user hook from the Import Sales Lead concurrent program.	Lead	S	Yes
Y	N	OM: Item Validation Organization	Manufacturing organization items are validated. This profile is used to set the AS_SALES_LEAD_LINES.organization_id.	Lead	S, R	-
Y	Y	OS: Minimum Number of Records for Parallel Processing in Lead Import	Determines the number of child processes to be spawned by the Import Sales Lead concurrent program during parallel lead import.	Lead	S	400

Table A-3 lists the profiles related to the Interaction Matching Engine.

Table A-3 Profiles for the Leads Interaction Matching Engine

Req.?	New?	Profile Name	Description	Program	Level	Default
N	Y	OS: Default Source for Interaction Matching Engine if Interaction does not have Source Code	Source promotion is not mandatory for Interaction. The default value is used when no promotion is specified.	Interaction	S	-
Y	Y	OS: Look Back Period in Days for Interactions when Interaction Matching Engine is run for the first time	The Interaction Matching Engine looks for interactions created for the past <i>n</i> days. The value in this profile is used when the engine is run for the very first time.	Interaction	S	30
Y	Y	OS: Look Back Period in Days for Leads in Interaction Matching Engine	The Interaction Matching Engine looks for leads created during the past <i>n</i> days. If not specified, the number of days is used from this profile.	Interaction	S	30
Y	Y	OS: Last Interaction ID processed in Interaction Matching Engine	This profile will be invisible and cannot be updated by users. The Interaction Matching Engine updates this profile to denote which interaction ID was the last interaction processed. If the value of this profile is 0, the interactions logged in the last <i>n</i> days as defined by the above profile are processed.	Interaction	S	0
Y	Y	OS: Default Interaction Score in Interaction Matching Engine	If the interaction does not have any business type matched in the rule, the default score from this profile is used.	Interaction	S	1
Y	Y	OS: Interaction Score Threshold	When the Import Sales Lead concurrent program processes records in the <code>as_import_interface</code> table, the value from this profile is used as the threshold for creating lead. If the interaction score is less than this value, a lead is not created.	Interaction	S	20
Y	Y	OS: Interaction Matching Engine matches contact for B2B lead	If set to Y, the Interaction Matching Engine matches primary contacts for B2B leads. If set to N, primary contacts are ignored.	Interaction	S	Y

Table A-4 lists the profiles related to the Leads Rule Engines.

Table A-4 Profiles for the Leads Rule Engines

Req.?	New?	Profile Name	Description	Program	Level	Default
Y	Y	OS: Default Qualified Flag for Lead Qualification Engine	When none of the Qualification rule sets match the lead, the default qualification flag is based on this profile.	Leads	S	No
Y	Y	OS: Enable Rule Engine Log	If set to 'Y', details about every rule engine run are logged into the <code>pv_entity_rules_applied</code> table.	Leads	S	Y
Y	Y	PV: Run lead deduplication rule	This profile controls the execution of the Lead Deduplication rule. If set to 'No', then lead deduplication will not be performed during lead import.	Lead Import	S	Y
Y	Y	OS: Execute custom code from lead import	This profile controls the invocation of custom user hook that can be executed for each record in a lead import batch. If the profile value is set to 'No', the custom code will not be executed.	Lead Import	S	Y
Y	Y	OS: Escalation Manager for Leads	Set this profile when there is one resource to act as an escalation manager. During the process of assigning a lead to a sales representative, if the maximum number of re-routes are reached, an e-mail is sent to the escalation manager.	Lead Assignment	S	-
Y	N	OS: Run Lead Monitor Engine	If set to Y, the lead is processed by the monitoring engine after it is created/updated.	Monitoring Engine	S	Y
Y	N	OS: Maximum Reroutes per leads	The number assigned to this profile defines the maximum number of reroutes allowed to a lead that is timed out by the monitoring engine or during automatic assignment.	Leads	S	3
N	N	OS: Lead View Scorecard data	Set to Y to view the old Score Card Mapping and rank data.	Leads	A,S,R, U	N

Table A-4 Profiles for the Leads Rule Engines

Req.?	New?	Profile Name	Description	Program	Level	Default
	N	OS: Default Channel for Lead Channel Selection Engine	This sets the default channel for each lead if no channel selection rule sets evaluate to true.	Leads	S	Direct
Y	N	OS: Default Rank for Lead Rating Engine	This sets the default if no rating rule sets evaluate to true.	Leads	S	Cold Lead
N	N	OS: Partners Enabled for Leads	If set to Yes, the Preferred Partner LOV in the Lead Details page is displayed. This is useful when the Channel of the lead is Indirect and is routed to a Partner.	Leads	S,A,R, U	No

[Table A-5](#) lists the profiles related to the assignment of leads.

Table A-5 Profiles for Leads Assignment

Req.?	New?	Profile Name	Description	Program	Level	Default
N	N	OS: Assign New Lead	<p>if this profile is set to No, then the application uses the Territory Manager to automatically assign resources to the lead. The first person the Territory Manager assigns becomes the owner. The rest of the resources in the territory become sales team members on the lead. If this profile is set to Yes, then the agent must enter the owner manually using the Owner LOV. If the agent does not make an entry, then the lead is assigned to the default user set in OS: Default Resource ID Used for Sales Leads. If no default resource is set, then application assigns ownership to the user updating or importing the lead.</p> <p>Note: If both the resource in this profile and the user who created or updated the lead do not have a valid sales role assigned to them, then the leads you import will not be accessible from either Oracle Sales Online or Oracle TeleSales.</p>	Leads	S	No
N	N	OS: Calendar setting required for assignment	The calendar setting for assigning an owner to a lead. If set to Yes, then the routing APIs check the availability of the resource in JTF Calendar before assigning the lead owner.	Leads	S	N
Y	N	OS: Default Lead Marketing Owner	The immature lead owner assignment API will use the default marketing lead owner if a owner cannot be found in the owner table.	Leads	S	-

Table A-5 Profiles for Leads Assignment

Req.?	New?	Profile Name	Description	Program	Level	Default
Y	N	OS: Sales Team Creator Keep Flag	Used for defaulting the keep flag in the lead sales team page.	Leads	S,A,R, U	Yes
N	N	OS: Default Resource ID Used for Sales Lead Assignment	<p>If the owner of a lead declines the lead, the owner will be the next available resource defined in territory. If all the resources are used up, the lead owner will be the one defined in this profile. The original lead owner, if not defined in territory, will be removed from the sales team.</p> <p>If this profile is not set, the ownership of any unassigned lead is automatically assigned to the user who is currently logged in to the application.</p>	Territorie s	S	-
N	N	OS: Lead Assignment User Hook	Used by the Sales Online Territory Assignment concurrent program. The program uses the value in this profile to identify if the user hook is enabled.	Territorie s	S,A,R, U	No

[Table A-6](#) lists the profiles related to the follow-up of leads.

Table A-6 Profiles for Leads Followup

Req.?	New?	Profile Name	Description	Program	Level	Default
N	N	OS: Lead New State Transition	If set to Y, the new state transitioning logic controls the transition of a lead state from one status to another, based on common business logic. If you do not want any additional restrictions, set this profile to N.	Leads	S	Y
Y	N	OS: Auto Run Lead Engines While Update	If set to Y, the lead engines are run automatically while updating a lead. If set to N, then the lead is processed manually. However, you can click Run Engines to process the lead via the lead engines.	Leads	S	Y
Y	N	OS: Address Required for Person	If set to Y, the Lead Contact page in Oracle Sales Online displays the Create Person button. The person or contact has to be created by clicking this button and entering address information on the Create Person page. If set to N, you can enter contact information in empty rows.	Leads	S	'N
N	N	OS: Address Required for Sales Lead	Makes address entry mandatory for leads. If set to Y, then the address is required. The API gives users an error if the address is required but not present in the record.	Leads	S	No
N	N	OS: Default Budget Status for Leads	The value in this profile will be the default value in the UI and API. If not set, this profile inserts a NULL value in the database table.	Leads	S, R, U	Pending
N	N	OS: Source Code Mandatory for Leads	Used to make the Campaign Source Code entry mandatory for leads. A Yes setting causes an error to appear if a user tries to save a lead without a source code.	Leads	S	No
N	N	OS: Customer Address Required for Customer, Opportunity and Lead	Indicates whether customer address is required for a customer.	Leads	S	No

Table A-6 Profiles for Leads Followup

Req.?	New?	Profile Name	Description	Program	Level	Default
Y	N	OS: Auto-relate Lead note to Customer	While creating a note, this profile is used to automatically relate the note to the customer.	Leads	S,A,R, U	Yes
Y	N	OS: Auto-relate Lead note to Primary Contact	While creating a note, this profile is used to automatically relate the note to the primary contact.	Leads	S,A,R, U	No
Y	N	OS: Auto-relate Lead task to Customer	While creating a task, this profile option is used to automatically relate the task to the customer.	Leads	S,A,R, U	Yes
Y	N	OS: Auto-relate Lead task to Primary Contact	While creating a task, this profile option is used to automatically relate the task to the primary contact.	Leads	S,A,R, U	No

Table A-7 lists the profiles related to the conversion of leads.

Table A-7 Profiles for Leads Conversion

Req.?	New?	Profile Name	Description	Program	Level	Default
N	N	OS: Auto Convert Lead to Opportunity	If a channel is defined as INDIRECT in the Channel Setup page, and this profile is set to 'Yes', an opportunity is created for the lead, and the partner matching workflow is started.	Leads	S	Yes
	N	OS: Lead Incubation Channel	Immature lead owner assignment will be used if the sales channel in this profile matches the channel assigned by the Channel Selection Engine.	Leads	S	-
Y	N	OS: Lead Link Status	The value in this profile is used to set the status of the lead after linking the lead to an opportunity.	Leads	S	Converted to Opportunity

Table A-7 Profiles for Leads Conversion

Req.?	New?	Profile Name	Description	Program	Level	Default
Y	N	OS: Lead to Opportunity Move Sales Team	When converting a lead to an opportunity, you can govern whether all the sales team members are copied to the opportunity sales team by setting this profile. If set to Yes, all sales team members are copied to the new opportunity. If set to No, the sales team is limited to the sales representative who converted the lead, plus those sales representatives who are included as per the Territory Manager settings. The Keep flag is copied as is, and the Owner flag is copied as No.	Leads	S	No

Table A-8 lists the profiles related to Lead Security.

Table A-8 Profiles for Leads Security

Req.?	New?	Profile Name	Description	Program	Level	Default
Y	N	OS: Customer Access Privilege	This profile determines who can view or update customer data.	Leads	S,R,U	Full
Y	N	OS: Manager Update Access	If a manager has subordinates in the sales team, the manager's view or update privileges will be determined by this profile.	Leads	S,R,U	View data
Y	N	OS: Opportunity Access Privilege	This profile determines who can view or update opportunity data.	Leads	S,R,U	Global Full
N	N	OS: Privilege to Change Lead Owner	If set to Yes, you can change the owner of a lead to which you have Update access. Users who do not have this privilege can change the owner of only those leads that they own. This profile is used to allow the user to change lead owner even if the user is not the current owner for lead.	Leads	S, R, U	No
Y	N	OS: Sales Admin Update Access	This profile determines whether an administrator can view or update data.	Leads	S,R,U	View data

Table A-8 Profiles for Leads Security

Req.?	New?	Profile Name	Description	Program	Level	Default
Y	N	OS: Sales Lead Access Privilege	This profile determines who can view or update lead data.	Leads	S, R, U	Full
Y	N	OS: Create Organization Privilege	If set to Yes, the Create Organization button is displayed on the Create page.	Leads	S,R,U	Yes
Y	N	OS: Create Person Privilege	If set to Y, the Create Person button is displayed on the Create page.	Leads	S,R,U	Yes

Table A-9 lists leads profiles in Oracle TeleSales (OTS).

Table A-9 Leads Profiles in Oracle TeleSales

Req.?	New?	Profile Name	Description	Program	Level	Default
Y	N	OTS: Default Country	Used for Lead Center in OTS.	TeleSales	S,A,R, U	US
Y	N	OTS: Default Party Type	Used to default the party type in Lead Center.	TeleSales	S,A,R, U	PARTY_RELATIONSHIP
Y	N	OTS: Default Tab in Sales Lead Center	Default tab in Lead Center.	TeleSales	S,A,R, U	AST_SL C_PUR
Y	N	OTS: Default Universal Search Tab	Default Universal Search tab.	TeleSales	S,A,R, U	QUICK SEARCH
Y	N	OTS: Default Universal Search Type	Default Universal Search type.	TeleSales	S,A,R, U	PARTY_RELATIONSHIP
Y	N	OTS: Interactions- Enable Automatic Start	Used to automatically start interactions.	TeleSales	S,A,R, U	N
Y	N	OTS: Show Campaign and Offer Names instead of Codes	If set to Y, Campaign Name is shown in the UI instead of Source Name.	TeleSales	S,A,R, U	N
N	N	OTS: Default User Role	Used to decide what the user's default role must be when the user logs in.	TeleSales	S,A,R, U	-
N	N	OTS: Default User Sales Group	Used to decide what the user's default sales group must be when the user logs in.	TeleSales	S,A,R, U	-

Table A-9 Leads Profiles in Oracle TeleSales

Req.?	New?	Profile Name	Description	Program	Level	Default
N	N	OTS: Minimum Number of Characters for Lookup	Minimum characters for long LOV lookup.	TeleSales	S,A,R, U	-
Y	N	OTS: Validation Level for Sales Leads	<p>If the value in the OS: Lead New State Transition is <i>Yes</i>, then the status of a lead that is converted to opportunity should not be updateable.</p> <p>In order to achieve this in the Universal Work Queue (UWQ), set this profile to 90. This ensures consistent behavior in the Lead Center, eBusiness Center, and the UWQ.</p>	TeleSales	S	90
Y	N	OTS: Default Tab in Sales Lead Center	Determines which tab is active when the Oracle Telesales Lead Center is launched.	TeleSales	S,A,R, U	AST_ SLC_ PUR
Y	N	OTS: Interactions-Generate Lead Activity	<p>Yes: Tracks activities related to leads.</p> <p>No: Disables tracking of activities related to leads.</p> <p>The default is set at the system level. Option can be set at system, application, responsibility, and user levels. If the default value for the profile option is removed, then the application functions as if the value is No.</p>	TeleSales	S,A,R, U	No
Y	N	OTS: Number of Days Prior to View	Sets the default value of days for leads and opportunities	TeleSales	S,A,R, U	-
Y	N	OTS: Use Primary Address to Create Lead in UWQ	Set to Yes to use the primary address when creating a lead.	TeleSales	S,A,R, U	-
Y	N	OTS: UWQ - auto launch lead and opppty center	Set to Yes to open Lead Center and Opportunity Center when a new lead or opportunity is created in UWQ.	TeleSales	S,A,R, U	-

Table A-10 lists profiles related to leads that enable calculations in multiple currencies.

Table A–10 Lead Profiles Options for Multiple Currencies

Req.?	New	Profile Option	Description	Program	Level	Default
Y	N	OS: Forecast Calendar	The name of the calendar you are using to manage your forecasts. Used to get the period set name used in the <code>as_period_rates</code> table.	Calendar	S	-
Y	N	OS: Daily Conversion Type	Daily conversion type for currency conversion. Used in the amount calculation.	Currency	V	Corporate
Y	N	OS: Date Mapping Type	Used to get the pseudo period rate from the start or end of the period.	Currency	S	S
Y	N	OS: Maximum Roll Days for Converting Amount	Maximum Roll Days for converting amount if a conversion rate does not exist for a day. Used to get maximum rollup days for currency conversion.	Currency	S	-
Y	Y	OS: Default Period Type for Currency Conversion	Default period type for currency conversion.	Currency	S	-

[Table A–11](#) lists the obsolete profiles.

Table A–11 Obsolete Profiles

Req.?	New?	Profile Name	Description	Program	Level	Default
N	N	OS: Auto assign from lead import	-	-	-	-
N	N	OS: Auto ranking from lead import	-	-	-	-
N	N	OS: Auto Qualify Lead	Turns automatic qualification for sales leads on or off. A setting of Yes, causes the application to attempt to qualify a lead when the Qualified check box is null. A setting of No means the user must qualify the lead manually by selecting the Qualified check box. If this profile is not set, it is defaulted to 'N'.	-	S	Yes

Table A-11 Obsolete Profiles

Req.?	New?	Profile Name	Description	Program	Level	Default
N	N	OS: Budget status required	Used by auto qualification process in leads API. If value is Yes, then the user must enter a budget status to qualify the lead. If the profile option value is No, then lead qualification does not depend budget status entry.	-	S	Yes
N	N	OS: Campaign code required	Used by auto qualification process in leads API. If value is Yes, then the user must enter a campaign code to qualify the lead. If the profile option value is No, then lead qualification does not depend on the campaign code.	-	S	Yes
N	N	OS: Contact phone required	Used by auto qualification process in leads API. If value is Yes, then the user must enter a contact phone to qualify the lead. If the profile option value is No, then lead qualification does not depend on the contact phone.	-	S	Yes
N	N	OS: Contact role required	Used by auto qualification process in leads API. If value is Yes, then the user must enter a contact role to qualify the lead. If the profile option value is No, then lead qualification does not depend on the contact role.	-	S	Yes
NOT USED	N	OS: Dead Lead Status	-	-	S	Dead Lead
N	N	OS: Decision time frame required	Used by auto qualification process in leads API. If value is Yes, then the user must enter a decision time frame to qualify the lead. If the profile option value is No, then lead qualification does not depend on the decision time frame.	-	S	Yes
Y	N	OS: Default Channel for Leads	Used for defaulting value in UI and also in API. If not set, then the application inserts a null value and notifies the user of an error.	-	S, R, U	Direct

Table A-11 Obsolete Profiles

Req.?	New?	Profile Name	Description	Program	Level	Default
n/a	N	OS: Default Lead Scorecard	Scorecard to be used for lead ranking. If not set, the scoring engine will not function. While creating leads, the UI retrieves this value and passes it to the scoring APIs.	-	SRA	1
Y	N	OS: Lead Routing Status	When the routing engine finds a sales lead owner, then the sales lead status is reset to this profile	-	S	New
N	N	OS: Project name required	Used by auto qualification process in leads API. If value is Yes, then the user must enter a project name to qualify the lead. If the profile option value is No, then lead qualification does not depend project name.	-	S	Yes
N	N	OS: Rank Lead	The Rating Engine is run when this is set to SYSTEM, if the Rank ID is null.	-	S	System
	N	OS: Run New Lead Qualification, Rating, Channel Selection Engines	If set to Y, the lead is processed by the Leads Rules Engine. If set to N, qualification and ranking engines qualify and rank the lead based on profiles.	-	S	Y
N	N	OS: Sales channel required	Used by auto qualification process in leads API. If value is Yes, then the user must enter a sales channel to qualify the lead. If the profile option value is No, then lead qualification does not depend on sales channel entry.	-	S	Yes
N	N	OS: Total budget amount required	Used by auto qualification process in leads API. If value is Yes, then the user must enter a total budget amount to qualify the lead. If the profile option value is No, then lead qualification does not depend on the total budget amount.	-	S	Yes

B

Seeded Data

This appendix chapter details seeded attributes and lookup values in Oracle Leads Management.

Topics included are:

- [Section B.1, "Seeded Attributes"](#)
- [Section B.2, "Seeded Lookups"](#)

B.1 Seeded Attributes

An attribute is a column in the database, and is represented as a field in the User Interface. Every object has its own set of seeded attributes.

Topics included in this section are:

- [Section B.1.1, "Seeded Lead Attributes"](#)
- [Section B.1.2, "Seeded Interaction Matching Engine Attributes"](#)

B.1.1 Seeded Lead Attributes

[Table B-1](#) lists the seeded attributes for a lead. The Remarks column indicates other functions that use the same attributes.

Table B-1 Seeded Attributes for a Lead

Attribute Name	Return Type	Description	Remarks
Customer	STRING	Compare as_sales_ leads.customer_id	-
Customer Address	NULL_CHECK	Check whether as_ sales_ leads.address_id is NULL.	Also used in the Deduplication Rule.
Customer Annual Revenue	CURRENCY	Compare customer revenue and currency amount.	-
Primary Contact	NUMBER	Compare as_sales_ leads.primary_ contact_party_id.	Also used in the Deduplication Rule.
Primary Contact Specified	NULL_CHECK	Check whether as_ sales_ leads.primary_ contact_party_id is NULL.	-
Primary Contact Role	STRING	Compare as_sales_ lead_ contacts.contact_ role_code.	-
Purchase Timeframe	STRING	Compare as_sales_ leads.decision_ timeframe_code.	Also used in the Deduplication Rule.

Table B-1 Seeded Attributes for a Lead

Attribute Name	Return Type	Description	Remarks
Budget Status	STRING	Compare <code>as_sales_</code> <code>leads.budget_</code> <code>status_code</code> .	Also used in the Deduplication Rule.
Lead Score	STRING	Compare <code>as_sales_</code> <code>leads.marketing_</code> <code>score</code> .	-
Lead Status	STRING	Compare <code>as_sales_</code> <code>leads.status_</code> <code>code</code> .	Also used in the Guards.
Total Budget	CURRENCY	Compare <code>as_sales_</code> <code>leads.budget_</code> <code>amount, currency</code> <code>amount comparison</code> .	Also used in the Deduplication Rule.
Product Category	STRING	Compare <code>as_sales_</code> <code>lead_</code> <code>lines.category_</code> <code>id</code> .	Also used in the Deduplication Rule, and the Guards.
Purchase Quantity - Product	NUMBER	Compare <code>as_sales_</code> <code>lead_</code> <code>lines.category_id</code> and <code>as_sales_</code> <code>lead_</code> <code>lines.quantity</code> .	-
Purchase Amount - Product	CURRENCY	Compare <code>as_sales_</code> <code>lead_</code> <code>lines.category_id</code> and <code>as_sales_</code> <code>lead_</code> <code>lines.budget_</code> <code>amount, currency</code> <code>amount comparison</code>	-
Response Channel	STRING	Compare <code>as_sales_</code> <code>leads.vehicle_</code> <code>response_code</code> .	Also used in the Deduplication Rule.
Project	STRING	Compare <code>as_sales_</code> <code>leads.parent_</code> <code>project</code> .	Also used in the Deduplication Rule.

Table B-1 Seeded Attributes for a Lead

Attribute Name	Return Type	Description	Remarks
Country	STRING	Compare lead customer location country	Also used in the Deduplication Rule, the Guards, and the Monitoring Engine.
Campaign	STRING	Compare as_sales_leads.source_promotion_id.	Also used in the Deduplication Rule, and the Guards.
Qualify Flag	STRING	Compare as_sales_leads.qualified_flag.	-
Lead Rating	STRING	Compare as_sales_leads.lead_rank_id.	Also used in the Monitoring Engine.
Sales Channel	STRING	Compare as_sales_leads.channel_code.	-
Creation Date	DATE	Compare as_sales_leads.creation_date.	Also used in the Guards.
Total Purchase Amount - Product	CURRENCY	Compare sum of as_sales_lead_lines.budget_amount, currency amount comparison.	Also used in the Deduplication Rule.
Customer/Account Type	STRING	Compare lead customer account type.	-
Phone Number Specified	NULL_CHECK	Check whether as_sales_lead_contacts.phone_id is NULL.	-
State	STRING	Compare lead customer location state.	Also used in the Guards.
Area Code	STRING	Compare lead customer or primary contact area code.	Also used in the Guards.

Table B-1 Seeded Attributes for a Lead

Attribute Name	Return Type	Description	Remarks
County	STRING	Compare lead customer location county.	Also used in the Guards.
Province	STRING	Compare lead customer location province.	Also used in the Guards.
City	STRING	Compare lead customer location city.	Also used in the Guards.
Postal Code	STRING	Compare lead customer location postal code.	Also used in the Guards.
Email Address Specified	NULL_CHECK	Check whether customer or primary contact e-mail address is NULL.	-
Email Address or Phone Specified	NULL_CHECK	Check whether customer or primary contact e-mail address or phone is NULL.	Also used in the Deduplication Rule.
Customer Category	STRING	Compare lead customer category.	Also used in the Guards.
Customer_Name	STRING	Compare lead customer name.	Also used in the Deduplication Rule.
Primary Contact Name	STRING	Compare primary contact name.	-
Primary Contact Job Title	STRING	Compare primary contact job title.	-
Email Domain	STRING	Compare customer or primary contact e-mail domain.	-
Email Address	STRING	Compare customer or primary contact e-mail address.	-
All	STRING	Matches everything.	Only used in the Guards.

Table B-1 Seeded Attributes for a Lead

Attribute Name	Return Type	Description	Remarks
Lead Note/Type	STRING	Compare lead note.	Also used in the Deduplication Rule.
Created Within	NUMBER	Compare the days when the lead was created.	Only used in the Deduplication Rule.
Interaction Score	NUMBER	Compare as_sales_ - leads.interaction_score.	
Custom Setup	STRING	Compare custom setup of as_sales_ - leads.source_promotion_id.	Only used in the Guards.

B.1.2 Seeded Interaction Matching Engine Attributes

Table B-2 lists the seeded attributes for the Interaction Matching Engine.

Table B-2 Seeded Attributes for the Interaction Matching Engine

Attribute Name	Return Type	Description
All	STRING	Matches everything.
Interaction Type	STRING	The combination of action and action item.
Campaign	STRING	Compare jtf_ih_interactions.source_code.
Country	STRING	Compare interaction customer location country.

B.2 Seeded Lookups

Lookups appear as drop-down values in the User Interface. This section discusses the seeded lookup values in Oracle Leads Management.

Topics included in this section are:

- [Section B.2.1, "Time Frame"](#)
- [Section B.2.2, "Lead Rank"](#)
- [Section B.2.3, "Lead Status"](#)

B.2.1 Time Frame

Target Table/View: AML_SALES_LEAD_TIMEFRAMES

Table B-3 gives the seeded lookup values for Timeframe.

Table B-3 Timeframe Seeded Lookup Values

Days	Timeframe Code
7	WITHIN 1 WEEK
30	WITHIN 1 MONTH
90	1 - 3 MONTHS
180	3 - 6 MONTHS
365	6 - 12 MONTHS
3650	MORE THAN 1 YEAR

B.2.2 Lead Rank

Target Table/View: AS_SALES_LEAD_RANKS_VL.

Table B-4 gives the seeded lookup values for Lead Rank.

Table B-4 Lead Rank Seeded Lookup Values

Description	Min Score	Max Score
Cold Lead	1	25
Low Lead	26	50
Medium Lead	51	75
Hot Lead	76	99

B.2.3 Lead Status

Target Table/View: AS_STATUSES_VL.

Table B-5 gives the seeded lookup values for Lead status.

Table B-5 Lead Status Seeded Lookup Values

Status Code	Open Status	Forecast Rollup	Win Loss	Usage Indicator	Meaning	Description
NEW	Y	N	N	ALL	New	New
LOSS	N	N	L	ALL	Loss	Loss
DEAD_LEAD	N	N	N	ALL	Dead Lead	Dead Lead
CONVERTED_TO_OPPORTUNITY	N	N	N	-	Converted to Opportunity	Lead Converted to Opportunity
IN_PROGRESS	Y	Y	Y	ALL	In Progress	Progress

C

Concurrent Programs

This appendix provides a table of all of the concurrent programs used by Oracle Leads Management.

Topics included are:

- [Section C.1, "Running Concurrent Programs"](#)
- [Section C.2, "Concurrent Programs in Oracle Leads Management"](#)

C.1 Running Concurrent Programs

The procedure for running concurrent programs is the same for all Oracle applications.

Prerequisites

Need-based.

Steps

1. Log in to Oracle Forms as Oracle Sales Administrator, and navigate to Concurrent Requests > Run.
2. Select the Single Request radio button.
3. Click **OK**.
4. Search for the concurrent request that you want to run.
5. Click **Submit**.
6. Select from the search results.
7. Click **OK**.
8. Enter parameters, if any, for the concurrent program.
9. Schedule the program to run as required.
10. Click **OK** to run the concurrent program.

For a detailed description of the procedures, see the *Oracle Applications System Administrator's Guide*.

C.2 Concurrent Programs in Oracle Leads Management

The following table lists by name and in alphabetical order the concurrent programs used by the Oracle Sales Family of eBusiness Suite applications. The table includes the following columns from left to right:

- **Concurrent Program Name:** Name of the concurrent program.
- **Description:** Explains what the concurrent program does.

Mandatory	Concurrent Program Name	Description
No	Assign Territory Accesses	<p>This concurrent program assigns new territory access to sales force employees. The program prepares database s for parallel processing. It requires setting three profile options:</p> <p>OS: Territory Minimum Number of Records for Parallel Processing</p> <p>OS: Territory Number of Child Processes</p> <p>OS: Territory Records to Open for Processing Changed Accounts</p> <p>Run this program after completing the setups in Setting Up Territory Management, and after the JTF Concurrent program Generate Territory Package has run.</p> <p>Parameters:</p> <p>Run Mode (New/Restart/Total)</p> <p>Lead Status: (All/Open/Closed)</p> <p>Previous Request ID for restart mode only.</p>
No	Autocreate Opportunity from Sales lead	<p>Creates opportunities from existing sales leads. This program should be run after the Import Sales Leads concurrent program. Parameters:</p> <p>Debug Mode</p> <p>Trace Mode</p>
No	Generate Access Records	<p>This is a child program of Assign Territory Accesses and does not need to be run separately.</p>
No	Generate Territory Packages	<p>This concurrent program, available by logging in under the CRM Administration responsibility, builds the API that returns the winning territories which are defined in territory setup. It must be run at least once before you import leads and each time after you modify the territory setup. You need not run this program every time you import leads.</p>

Mandatory	Concurrent Program Name	Description
Yes	Initial Build of Opportunity and Lead Bins	<p>Used for the Opportunity and New Leads home page bins and reports. This program must be run initially before users can set up opportunity and new leads bins and reports. This program loads the opportunity and leads materialized view with sales credit information. A new parameter: Next Extent Size has been added. You can choose the value of the next extent to be allocated for all parameters and indexes created. Possible values for the parameter are: Small (1M), Medium (5M), Large (10M)</p> <p>Note: This program must be run anytime the as_sales_credit_denorm is rebuilt or when values such as sales stages and statuses have changed. You should also run when Refresh AS_PERIOD_DAYS is run.</p>
Yes	Initial Load for Lead Reports	<p>This program is used to build Leads bins and reports. This should also be run following Refresh of Leads Bins Data so that you can see the latest values in leads reports.</p>
No	Load Interest Types and Codes to Inventory Categories (Load Categories)	<p>This concurrent program creates inventory categories under the inventory category set Oracle Sales and Marketing for each combination of interest types and codes. In order to use this concurrent program, the OS: Inventory Category Integration profile value must be set to Yes.</p>
No	OTS: Load Sales Lead Interface from Flat File	<p>Use this program to import sales leads from the interface. This program must be run before Auto Create Opportunity from Sales Lead. This moves data from the interface to AS_SALES_LEADS, AS_SALES_LEADS_LINES, and AS_SALES_LEAD_CONTACTS. If you want to import data from a flat file, you must first run OTS: Load Sales Lead, which will move data from the flat file to the interface .</p>
No	Refresh of Leads Bin Data	<p>This program is used to run an incremental refresh of the materialized view used for the Leads bins and reports. This program should be scheduled to run periodically.</p>
No	Refresh of Opportunity Bins Data	<p>This program is used to run an incremental refresh of the materialized view used for the opportunity bins and reports. This program should be scheduled to run periodically.</p> <p>Note: Users will not see opportunities in bins created after the last refresh of Refresh Sales Credit.</p>
No	Setup Checking for Oracle Sales application	<p>This program validates Sales Setups and produces an error log that the system administrator can use to diagnose invalid setups.</p> <p>Parameters: Upgrade - Yes/No</p>

Mandatory	Concurrent Program Name	Description
Y	Import Sales Leads	The lead import concurrent program allows you to import leads into Oracle Sales from other systems. While importing leads, the program also imports data on customers, addresses, and contacts into the customer model (TCA) tables.
Y	Workflow Background Process	This program sends notifications and reminders from triggered monitors. The system administrator must schedule it to run everyday, or twice a day, if required.
Y	Synchronization program	The synchronization concurrent program must be run after every lead import activity. This updates the DQM staging schema with new entries that were created during the lead import.
No	DQM Compile All Rules	This program must be run when any rules are modified in DQM.
Yes	DQM Staging Program	The DQM Staging Program must be run when DQM is set up for the first time. It creates the staging schema and is vital for existence checking.
No	Purge Lead Import Interface Table	The AS_IMPORT_INTERFACE table is a temporary location where the imported records are stored before unique records are moved to the AS_SALES_LEAD table. Run the Purge Lead Import Interface Table concurrent program to delete the records from this table.
Yes	Run Interaction Matching Engine to Match or Create Leads	The Run Interaction Matching Engine to Match or Create Leads concurrent program runs the Interaction Matching Engine. The Interaction Matching Engine provides the mechanism to mine and evaluate customer interactions and responses for sales follow up. The Interaction Matching Engine is driven by a rule with an activation date range.
No	Purge Unqualified Sales Leads	<p>This program removes unqualified leads from the AS_SALES_LEADS table. Unqualified leads are those leads whose Qualified attribute is not selected in the application. Unqualified leads satisfying the following conditions are removed:</p> <ul style="list-style-type: none"> ■ The lead status has not changed since the creation date. ■ The lead's original status is the same as that defined in the OS: Default Status for Leads profile.
Yes (applicable only after upgrade to release 11.5.10)	Product Catalog Migration for Leads	<p>Run this program only after upgrading to release 11.5.10. This program migrates the interest_type_id and interest_code_id details to a new product hierarchy.</p> <p>This program is not applicable for new 11.5.10 users.</p>

Oracle Leads Management API Reference

The public APIs provided by Oracle Leads Management are described and grouped according to procedures using them.

Topics included are:

- [Section D.1, "Oracle Leads Management Procedures"](#)
- [Section D.2, "Parameter Specifications"](#)
- [Section D.3, "Type Declarations"](#)
- [Section D.4, "Create Sales Lead"](#)
- [Section D.5, "Update Sales Lead"](#)
- [Section D.6, "Update Sales Lead Lines"](#)
- [Section D.7, "Delete Sales Lead Lines"](#)
- [Section D.8, "Lead Process After Create"](#)
- [Section D.9, "Lead Process After Update"](#)
- [Section D.10, "Run Lead Engines"](#)
- [Section D.11, "Build Lead Sales Team"](#)
- [Section D.12, "Rebuild Lead Sales Team"](#)
- [Section D.13, "Update Sales Lead Contacts"](#)
- [Section D.14, "Delete Sales Lead Contacts"](#)

D.1 Oracle Leads Management Procedures

Table D–1 lists the procedures which make up the Leads Public APIs.

Table D–1 Leads Public API Procedures

Procedure Name	Description
Create Sales Lead	Creates a new sales lead with the specified parameters. A unique sales lead ID will be created. This API calls the Create Sales Lead Lines and Create Sales Lead Contacts APIs internally if the appropriate parameters are passed.
Update Sales Lead	Updates the sales lead record. The sales lead record being updated must refer to a valid sales lead ID in the database and must have a valid last_update_date (who column) passed in. This is used to check if the record has not been updated by someone else since it was last loaded. If a g_miss value is passed for other fields in the sales lead record type, then these values will not be updated.
Update Sales Lead Lines	Updates one or more sales lead lines. A table of sales lead line records with the parameters needs to be passed. Each sales lead line record must refer to a valid sales lead line ID in the database and must have a valid last_update_date (who column passed in). This is used to check if the sales lead line record has not been updated by someone else since it was last loaded. If a g_miss value is passed for other fields in the sales lead line record type, then these will not be updated.
Delete Sales Lead Lines	Deletes one or more sales lead lines. A table of sales lead line records needs to be passed in . Each sales lead line record must refer to a valid sales lead line ID in the database.
Lead Process After Create	This is a wrapper of the Run Lead Engines and the Build Lead Sales Team APIs. It calls the above API based on specific logic to keep lead integrity, and launches workflow process if users want to monitor the lead.
Lead Process After Update	This is a wrapper of the Run Lead Engines and the Rebuild Lead Sales Team APIs. It calls the above APIs based on specific logic to keep lead integrity.
Run Lead Engines	Runs the qualification engine, rating engine, and channel selection engine.
Build Lead Sales Team	Builds lead sales team based on territory definition and adds lead creator as one of lead sales team members.

Table D–1 Leads Public API Procedures

Procedure Name	Description
Rebuild Lead Sales Team	Rebuilds lead sales team to reflect the latest lead information.
Update Sales Lead Contacts	Updates one or more sales lead contacts. A table of sales lead contacts records with the parameters needs to be passed in. Each sales lead contact record must refer to a valid sales lead contact ID in the database and must have a valid last_update_date (who column passed in). This is used to check if the sales lead contact record has not been updated by someone else since it was last loaded. If a g_miss value is passed for other fields in the sales lead contact record type, then these will not be updated.
Delete Sales Lead Contacts	Deletes one or more sales lead contacts. A table of sales lead contact records needs to be passed in . Each sales lead contact record must refer to a valid lead contact ID in the database.

D.2 Parameter Specifications

The specifications for the public APIs provided by the Oracle CRM Application Foundation define four categories of parameters:

- Standard IN
- Standard OUT
- Procedure specific IN
- Procedure specific OUT

Standard IN and OUT parameters are specified by the Oracle Applications business object API Coding Standards, and are discussed in the following sections.

Procedure specific IN and OUT parameter are related to the API being specified, and are discussed with that individual API.

D.2.1 Standard IN Parameters

[Table D–2](#) describes standard IN parameters which are common to all APIs provided by Oracle Leads Management.

Table D-2 Standard IN Parameters

Parameter	Data Type	Required	Description
p_api_version	NUMBER	Yes	This must match the version number of the API. An unexpected error is returned if the calling program version number is incompatible with the current API version number.
p_init_msg_list	VARCHAR2	Yes	Default = FND_API.G_FALSE If set to true, then the API makes a call to fnd_msg_pub.initialize to initialize the message stack. If set to false the calling program must initialize the message stack. This action is required to be performed only once, even in the case where more than one API is called.
p_commit	VARCHAR2	No	Default = FND_API.G_FALSE If set to true, the API commits before returning to the calling program. If set to false, then it is the calling program's responsibility to commit the transaction.
p_validation_level	NUMBER	No	Level of validation required. If set to NONE, no validation will be done in the API. If set to FULL, all validations (item level and record level) will be performed.
P_check_access_flag	VARCHAR2	No	If set to Y, access security check is performed. If set to N, access security check is not performed.
P_Admin_Flag	VARCHAR2	No	If set to Y, the current user has administrator privileges. If set to N, the current user does not have administrator privileges.
P_Admin_Group_Id	NUMBER	No	If the current user has administrator privileges, the user's SalesgroupID.

Table D–2 Standard IN Parameters

Parameter	Data Type	Required	Description
P_Identity_ salesforce_Id	NUMBER	No	Resource ID of the current user.

D.2.2 Standard OUT Parameters

Table D–3 describes standard OUT parameters, which are common to all public APIs provided by Oracle Leads Management.

Note: All standard OUT parameters are required.

Table D–3 Standard OUT Parameters

Parameter	Data Type	Description
x_return_status	VARCHAR2(1)	Indicates the return status of the API. The values returned are one of the following: FND_API.G_RET_STS_SUCCESS which indicates the API call was successful. FND_API.G_RET_STS_ERROR which indicates there was a validation error or a missing data error. FND_API.G_RET_STS_UNEXP_ERROR which indicates the calling program encountered an unexpected or unhandled error.
x_return_status	VARCHAR2	Default = FND_API.G_FALSE If set to true, then the API makes a call to fnd_msg_pub.initialize to initialize the message stack. If set to false the calling program must initialize the message stack. This action is required to be performed only once, even in the case where more than one API is called.

Table D-3 Standard OUT Parameters

Parameter	Data Type	Description
x_msg_count	NUMBER	Holds the number of messages in the message list.If the error message returned is one, then the message count will be zero.
x_msg_data	VARCHAR2(2000)	Error message returned by the API. If the number of messages is more than one, this parameter will be NULL and the messages must be extracted from the message stack.

D.3 Type Declarations

```

TYPE SALES_LEAD_Rec_Type IS RECORD
(
    SALES_LEAD_ID                NUMBER := FND_API.G_MISS_NUM,
    LAST_UPDATE_DATE             DATE := FND_API.G_MISS_DATE,
    LAST_UPDATED_BY              NUMBER := FND_API.G_MISS_NUM,
    CREATION_DATE                DATE := FND_API.G_MISS_DATE,
    CREATED_BY                   NUMBER := FND_API.G_MISS_NUM,
    LAST_UPDATE_LOGIN            NUMBER := FND_API.G_MISS_NUM,
    REQUEST_ID                   NUMBER := FND_API.G_MISS_NUM,
    PROGRAM_APPLICATION_ID       NUMBER := FND_API.G_MISS_NUM,
    PROGRAM_ID                   NUMBER := FND_API.G_MISS_NUM,
    PROGRAM_UPDATE_DATE          DATE := FND_API.G_MISS_DATE,
    LEAD_NUMBER                  VARCHAR2(30) := FND_API.G_MISS_CHAR,
    STATUS_CODE                  VARCHAR2(30) := FND_API.G_MISS_CHAR,
    CUSTOMER_ID                  NUMBER := FND_API.G_MISS_NUM,
    ADDRESS_ID                   NUMBER := FND_API.G_MISS_NUM,
    SOURCE_PROMOTION_ID          NUMBER := FND_API.G_MISS_NUM,
    INITIATING_CONTACT_ID       NUMBER := FND_API.G_MISS_NUM,
    ORIG_SYSTEM_REFERENCE        VARCHAR2(240) := FND_API.G_MISS_CHAR,
    CONTACT_ROLE_CODE            VARCHAR2(30) := FND_API.G_MISS_CHAR,
    CHANNEL_CODE                 VARCHAR2(30) := FND_API.G_MISS_CHAR,
    BUDGET_AMOUNT                NUMBER := FND_API.G_MISS_NUM,
    CURRENCY_CODE                VARCHAR2(15) := FND_API.G_MISS_CHAR,
    DECISION_TIMEFRAME_CODE     VARCHAR2(30) := FND_API.G_MISS_CHAR,
    CLOSE_REASON                 VARCHAR2(30) := FND_API.G_MISS_CHAR,
    LEAD_RANK_ID                NUMBER := FND_API.G_MISS_NUM,
    LEAD_RANK_CODE               VARCHAR2(30) := FND_API.G_MISS_CHAR,
    PARENT_PROJECT               VARCHAR2(80) := FND_API.G_MISS_CHAR,
    DESCRIPTION                   VARCHAR2(2000) := FND_API.G_MISS_CHAR,

```

ATTRIBUTE_CATEGORY	VARCHAR2(30) := FND_API.G_MISS_CHAR,
ATTRIBUTE1	VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE2	VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE3	VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE4	VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE5	VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE6	VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE7	VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE8	VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE9	VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE10	VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE11	VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE12	VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE13	VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE14	VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE15	VARCHAR2(150) := FND_API.G_MISS_CHAR,
ASSIGN_TO_PERSON_ID	NUMBER := FND_API.G_MISS_NUM,
ASSIGN_TO_SALESFORCE_ID	NUMBER := FND_API.G_MISS_NUM,
ASSIGN_SALES_GROUP_ID	NUMBER := FND_API.G_MISS_NUM,
ASSIGN_DATE	DATE := FND_API.G_MISS_DATE,
BUDGET_STATUS_CODE	VARCHAR2(30) := FND_API.G_MISS_CHAR,
ACCEPT_FLAG	VARCHAR2(1) := FND_API.G_MISS_CHAR,
VEHICLE_RESPONSE_CODE	VARCHAR2(30) := FND_API.G_MISS_CHAR,
TOTAL_SCORE	NUMBER := FND_API.G_MISS_NUM,
SCORECARD_ID	NUMBER := FND_API.G_MISS_NUM,
KEEP_FLAG	VARCHAR2(1) := FND_API.G_MISS_CHAR,
URGENT_FLAG	VARCHAR2(1) := FND_API.G_MISS_CHAR,
IMPORT_FLAG	VARCHAR2(1) := FND_API.G_MISS_CHAR,
REJECT_REASON_CODE	VARCHAR2(30) := FND_API.G_MISS_CHAR,
DELETED_FLAG	VARCHAR2(1) := FND_API.G_MISS_CHAR,
OFFER_ID	NUMBER := FND_API.G_MISS_NUM,
INCUMBENT_PARTNER_PARTY_ID	NUMBER := FND_API.G_MISS_NUM,
INCUMBENT_PARTNER_RESOURCE_ID	NUMBER := FND_API.G_MISS_NUM,
PRM_EXEC_SPONSOR_FLAG	VARCHAR2(1) := FND_API.G_MISS_CHAR,
PRM_PRJ_LEAD_IN_PLACE_FLAG	VARCHAR2(1) := FND_API.G_MISS_CHAR,
PRM_SALES_LEAD_TYPE	VARCHAR2(30) := FND_API.G_MISS_CHAR,
PRM_IND_CLASSIFICATION_CODE	VARCHAR2(30) := FND_API.G_MISS_CHAR,
QUALIFIED_FLAG	VARCHAR2(1) := FND_API.G_MISS_CHAR,
ORIG_SYSTEM_CODE	VARCHAR2(30) := FND_API.G_MISS_CHAR,
PRM_ASSIGNMENT_TYPE	VARCHAR2(30) := FND_API.G_MISS_CHAR,
AUTO_ASSIGNMENT_TYPE	VARCHAR2(30) := FND_API.G_MISS_CHAR,
PRIMARY_CONTACT_PARTY_ID	NUMBER := FND_API.G_MISS_NUM,
PRIMARY_CNT_PERSON_PARTY_ID	NUMBER := FND_API.G_MISS_NUM,
PRIMARY_CONTACT_PHONE_ID	NUMBER := FND_API.G_MISS_NUM,
REFERRED_BY	NUMBER := FND_API.G_MISS_NUM,

Type Declarations

```
REFERRAL_TYPE                VARCHAR2(30) := FND_API.G_MISS_CHAR,
REFERRAL_STATUS              VARCHAR2(30) := FND_API.G_MISS_CHAR,
REF_DECLINE_REASON           VARCHAR2(30) := FND_API.G_MISS_CHAR,
REF_COMM_LTR_STATUS          VARCHAR2(30) := FND_API.G_MISS_CHAR,
REF_ORDER_NUMBER             NUMBER := FND_API.G_MISS_NUM,
REF_ORDER_AMT                NUMBER := FND_API.G_MISS_NUM,
REF_COMM_AMT                 NUMBER := FND_API.G_MISS_NUM
);
G_MISS_SALES_LEAD_REC        SALES_LEAD_Rec_Type;
TYPE SALES_LEAD_Tbl_Type     IS TABLE OF SALES_LEAD_Rec_Type
INDEX BY BINARY_INTEGER;

G_MISS_SALES_LEAD_TBL       SALES_LEAD_Tbl_Type;

TYPE SALES_LEAD_LINE_Rec_Type IS RECORD
(
    SALES_LEAD_LINE_ID       NUMBER := FND_API.G_MISS_NUM,
    LAST_UPDATE_DATE         DATE := FND_API.G_MISS_DATE,
    LAST_UPDATED_BY          NUMBER := FND_API.G_MISS_NUM,
    CREATION_DATE            DATE := FND_API.G_MISS_DATE,
    CREATED_BY               NUMBER := FND_API.G_MISS_NUM,
    LAST_UPDATE_LOGIN        NUMBER := FND_API.G_MISS_NUM,
    REQUEST_ID               NUMBER := FND_API.G_MISS_NUM,
    PROGRAM_APPLICATION_ID    NUMBER := FND_API.G_MISS_NUM,
    PROGRAM_ID               NUMBER := FND_API.G_MISS_NUM,
    PROGRAM_UPDATE_DATE      DATE := FND_API.G_MISS_DATE,
    SALES_LEAD_ID            NUMBER := FND_API.G_MISS_NUM,
    STATUS_CODE              VARCHAR2(30) := FND_API.G_MISS_CHAR,
    INTEREST_TYPE_ID         NUMBER := FND_API.G_MISS_NUM,
    PRIMARY_INTEREST_CODE_ID NUMBER := FND_API.G_MISS_NUM,
    SECONDARY_INTEREST_CODE_ID NUMBER := FND_API.G_MISS_NUM,
    INVENTORY_ITEM_ID        NUMBER := FND_API.G_MISS_NUM,
    ORGANIZATION_ID          NUMBER := FND_API.G_MISS_NUM,
    UOM_CODE                 VARCHAR2(3) := FND_API.G_MISS_CHAR,
    QUANTITY                 NUMBER := FND_API.G_MISS_NUM,
    BUDGET_AMOUNT            NUMBER := FND_API.G_MISS_NUM,
    SOURCE_PROMOTION_ID      NUMBER := FND_API.G_MISS_NUM,
    ATTRIBUTE_CATEGORY        VARCHAR2(30) := FND_API.G_MISS_CHAR,
    ATTRIBUTE1               VARCHAR2(150) := FND_API.G_MISS_CHAR,
    ATTRIBUTE2               VARCHAR2(150) := FND_API.G_MISS_CHAR,
    ATTRIBUTE3               VARCHAR2(150) := FND_API.G_MISS_CHAR,
    ATTRIBUTE4               VARCHAR2(150) := FND_API.G_MISS_CHAR,
    ATTRIBUTE5               VARCHAR2(150) := FND_API.G_MISS_CHAR,
    ATTRIBUTE6               VARCHAR2(150) := FND_API.G_MISS_CHAR,
    ATTRIBUTE7               VARCHAR2(150) := FND_API.G_MISS_CHAR,
```

```

ATTRIBUTE8                VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE9                VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE10               VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE11               VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE12               VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE13               VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE14               VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE15               VARCHAR2(150) := FND_API.G_MISS_CHAR,
OFFER_ID                  NUMBER       := FND_API.G_MISS_NUM
CATEGORY_SET_ID           NUMBER       := FND_API.G_MISS_NUM
CATEGORY_ID                NUMBER       := FND_API.G_MISS_NUM
);

G_MISS_SALES_LEAD_LINE_REC SALES_LEAD_LINE_Rec_Type;
TYPE SALES_LEAD_LINE_Tbl_Type IS TABLE OF SALES_LEAD_LINE_Rec_Type
INDEX BY BINARY_INTEGER;

G_MISS_SALES_LEAD_LINE_TBL SALES_LEAD_LINE_Tbl_Type;

TYPE SALES_LEAD_LINE_OUT_Rec_Type IS RECORD
(
    SALES_LEAD_LINE_ID          NUMBER,
    RETURN_STATUS               VARCHAR2(1)
);

TYPE SALES_LEAD_LINE_OUT_Tbl_Type IS TABLE OF SALES_LEAD_LINE_OUT_Rec_Type
INDEX BY BINARY_INTEGER;

TYPE SALES_LEAD_CONTACT_Rec_Type IS RECORD
(
    LEAD_CONTACT_ID            NUMBER := FND_API.G_MISS_NUM,
    SALES_LEAD_ID              NUMBER := FND_API.G_MISS_NUM,
    CONTACT_ID                  NUMBER := FND_API.G_MISS_NUM,
    LAST_UPDATE_DATE            DATE := FND_API.G_MISS_DATE,
    LAST_UPDATED_BY             NUMBER := FND_API.G_MISS_NUM,
    CREATION_DATE               DATE := FND_API.G_MISS_DATE,
    CREATED_BY                  NUMBER := FND_API.G_MISS_NUM,
    LAST_UPDATE_LOGIN           NUMBER := FND_API.G_MISS_NUM,
    REQUEST_ID                  NUMBER := FND_API.G_MISS_NUM,
    PROGRAM_APPLICATION_ID      NUMBER := FND_API.G_MISS_NUM,
    PROGRAM_ID                  NUMBER := FND_API.G_MISS_NUM,
    PROGRAM_UPDATE_DATE         DATE := FND_API.G_MISS_DATE,
    ENABLED_FLAG                 VARCHAR2(1) := FND_API.G_MISS_CHAR,
    RANK                         VARCHAR2(30) := FND_API.G_MISS_CHAR,
    CUSTOMER_ID                 NUMBER := FND_API.G_MISS_NUM,

```

Type Declarations

```
ADDRESS_ID                NUMBER := FND_API.G_MISS_NUM,
PHONE_ID                  NUMBER := FND_API.G_MISS_NUM,
CONTACT_ROLE_CODE        VARCHAR2(30) := FND_API.G_MISS_CHAR,
PRIMARY_CONTACT_FLAG     VARCHAR2(1) := FND_API.G_MISS_CHAR,
ATTRIBUTE_CATEGORY       VARCHAR2(30) := FND_API.G_MISS_CHAR,
ATTRIBUTE1                VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE2                VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE3                VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE4                VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE5                VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE6                VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE7                VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE8                VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE9                VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE10               VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE11               VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE12               VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE13               VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE14               VARCHAR2(150) := FND_API.G_MISS_CHAR,
ATTRIBUTE15               VARCHAR2(150) := FND_API.G_MISS_CHAR,
CONTACT_PARTY_ID         NUMBER := FND_API.G_MISS_NUM
);

G_MISS_SALES_LEAD_CONTACT_REC    SALES_LEAD_CONTACT_Rec_Type;

TYPE SALES_LEAD_CONTACT_Tbl_Type  IS TABLE OF SALES_LEAD_CONTACT_Rec_Type
INDEX BY BINARY_INTEGER;

G_MISS_SALES_LEAD_CONTACT_TBL    SALES_LEAD_CONTACT_Tbl_Type;
TYPE SALES_LEAD_CNT_OUT_Rec_Type  IS RECORD
(
    LEAD_CONTACT_ID              NUMBER,
    RETURN_STATUS                 VARCHAR2(1)
);

TYPE SALES_LEAD_CNT_OUT_Tbl_Type  IS TABLE OF SALES_LEAD_CNT_OUT_Rec_Type
INDEX BY BINARY_INTEGER;   camp_rec_type IS
TYPE LEAD_ENGINES_OUT_Rec_Type    IS RECORD
(
    qualified_flag                VARCHAR2(1),
    lead_rank_id                  NUMBER,
    channel_code                   VARCHAR2(30),
    indirect_channel_flag          VARCHAR2(1),
    sales_team_flag                VARCHAR2(1)
);
```

D.4 Create Sales Lead

The Create Sales procedure creates a sales lead with the parameters specified. A unique sales lead ID is created.

Note: The Create Sales Lead API is an atomic API in the AML_SALES_LEAD_V2_PUB package. It is not the same as the Create Sales Lead API in the AS_SALES_LEADS_PUB package.

D.4.1 Procedure Specification

```

PROCEDURE Create_sales_lead (
  P_Api_Version_Number      IN  NUMBER,
  P_Init_Msg_List           IN  VARCHAR2      := FND_API.G_FALSE,
  P_Commit                  IN  VARCHAR2      := FND_API.G_FALSE,
  P_Validation_Level        IN  NUMBER        := FND_API.G_VALID_LEVEL_FULL,
  P_Check_Access_Flag       IN  VARCHAR2      := FND_API.G_MISS_CHAR,
  P_Admin_Flag              IN  VARCHAR2      := FND_API.G_MISS_CHAR,
  P_Admin_Group_Id          IN  NUMBER        := FND_API.G_MISS_NUM,
  P_Identity_Salesforce_Id  IN  NUMBER        := FND_API.G_MISS_NUM,
  P_Sales_Lead_Profile_Tbl  IN  AS_UTILITY_PUB.Profile_Tbl_Type
                               := AS_UTILITY_PUB.G_MISS_PROFILE_TBL,
  P_SALES_LEAD_Rec          IN  AS_SALES_LEADS_PUB.SALES_LEAD_Rec_Type
                               := AS_SALES_LEADS_PUB.G_MISS_SALES_LEAD_REC,
  P_SALES_LEAD_LINE_Tbl     IN  AS_SALES_LEADS_PUB.SALES_LEAD_LINE_Tbl_type
                               := AS_SALES_LEADS_PUB.G_MISS_SALES_LEAD_LINE_
Tbl,
  P_SALES_LEAD_CONTACT_Tbl IN  AS_SALES_LEADS_PUB.SALES_LEAD_CONTACT_Tbl_Type
                               := AS_SALES_LEADS_PUB.G_MISS_SALES_LEAD_
CONTACT_Tbl,
  P_Lead_note               IN  VARCHAR2      DEFAULT NULL,
  P_Note_type               IN  VARCHAR2      DEFAULT NULL,
  X_SALES_LEAD_ID           OUT NOCOPY NUMBER,
  X_SALES_LEAD_LINE_OUT_Tbl OUT NOCOPY AS_SALES_LEADS_PUB.SALES_LEAD_LINE_OUT_
Tbl_type,
  X_SALES_LEAD_CNT_OUT_Tbl  OUT NOCOPY AS_SALES_LEADS_PUB.SALES_LEAD_CNT_OUT_
Tbl_Type,
  X_note_id                 OUT NOCOPY NUMBER,
  X_Return_Status           OUT NOCOPY VARCHAR2,
  X_Msg_Count               OUT NOCOPY NUMBER,
  X_Msg_Data                OUT NOCOPY VARCHAR2
)

```

Current Version

2.0

D.4.2 Parameter Descriptions**Notes**

- A unique sales lead ID is generated from the sequence.
- In P_Sales_Lead_Rec, the required parameters are status_code, customer_id and source_promotion_id (based on profile)
- If P_sales_lead_line_tbl and/or P_sales_lead_contact_tbl is passed in, then the appropriate create APIs for sales lead lines and sales lead contacts is also called after creating the sales lead header.

Table D–4 describes the IN parameters for the Create Sales Lead procedure.

Table D–4 IN Parameters

Parameter	Data Type	Required	Description
P_Identity_Salesforce_Id	NUMBER	No	Salesforce Id of logged in user
P_Sales_Lead_Profile_Tbl	AS_UTILITY_PUB.Profile_Tbl_Type	No	Data type to store the access security related profile values (they can be cached mid-tier and passed to the API)
P_sales_lead_rec	AS_SALES_LEADS_PUB.SALES_LEAD_REC_TYPE	No	Sales lead record
P_Sales_lead_line_tbl	AS_SALES_LEADS_PUB.SALES_LEAD_LINE_TBL_TYPE	No	Table of sales lead line records
P_Sales_lead_contact_tbl	AS_SALES_LEADS_PUB.SALES_LEAD_CONTACT_TBL_TYPE	No	Table of sales lead contact records
P_Lead_note	VARCHAR2	No	Note to be associated with the lead
P_Note_type	VARCHAR2	No	Type of lead note

Table D–5 describes the OUT parameters for the Create Sales Lead procedure.

Table D-5 OUT Parameters

Parameter	Data Type	Description
X_sales_lead_ID	NUMBER	Sales lead ID of the sales lead just created
X_sales_lead_line_out_tbl	AS_SALES_LEADS_PUB.SALES_LEAD_LINE_OUT_TBL_TYPE	Table of sales lead line out record types. Each record type consists of the sales lead line ID created and the return status.
X_sales_lead_contact_out_tbl	AS_SALES_LEADS_PUB.SALES_LEAD_CONTACT_OUT_TBL_TYPE	Table of sales lead contact out record types. Each record type consists of the lead contact ID created and the return status.
X_note_id	NUMBER	Generated lead note id for the sales lead

D.5 Update Sales Lead

This procedure updates a sales lead with the parameters specified. A valid sales lead ID must be passed in.

D.5.1 Procedure Specification

```

PROCEDURE Update_sales_lead(
    P_Api_Version_Number    IN    NUMBER,
    P_Init_Msg_List         IN    VARCHAR2    := FND_API.G_FALSE,
    P_Commit                 IN    VARCHAR2    := FND_API.G_FALSE,
    P_Validation_Level      IN    NUMBER       := FND_API.G_VALID_LEVEL_FULL,
    P_Check_Access_Flag     IN    VARCHAR2    := FND_API.G_MISS_CHAR,
    P_Admin_Flag            IN    VARCHAR2    := FND_API.G_MISS_CHAR,
    P_Admin_Group_Id        IN    NUMBER       := FND_API.G_MISS_NUM,
    P_identity_salesforce_id IN    NUMBER       := FND_API.G_MISS_NUM,
    P_Sales_Lead_Profile_Tbl IN    AS_UTILITY_PUB.Profile_Tbl_Type := AS_UTILITY_
PUB.G_MISS_PROFILE_TBL,
    P_SALES_LEAD_Rec        IN    SALES_LEAD_Rec_Type
                                DEFAULT G_MISS_SALES_LEAD_REC,
    X_Return_Status         OUT   VARCHAR2,
    X_Msg_Count             OUT   NUMBER,
    X_Msg_Data              OUT   VARCHAR2
);

```

Current Version

2.0

D.5.2 Parameter Descriptions**Notes**

- A valid sales lead ID must be passed in the sales lead record type.
- Last_update_date must be passed in.

Table D–6 describes the IN parameters for the Update Sales Lead procedure.

Table D–6 IN Parameters

Parameters	Data Type	Required	Description
P_Sales_Lead_Profile_Tbl	AS_UTILITY_PUB.Profile_Tbl_Type	No	Data type to store the access security related profile values (they can be cached in the mid tier and passed to the API)
P_sales_lead_rec	AS_SALES_LEADS_PUB.SALES_LEAD_REC_TYPE	No	Sales lead record type for update

Note: This procedure does not have OUT parameters.

D.6 Update Sales Lead Lines

This procedure updates one or more sales lead lines with the parameters specified.

D.6.1 Procedure Specification

```

PROCEDURE Update_sales_lead_lines(
  P_Api_Version_Number      IN    NUMBER,
  P_Init_Msg_List          IN    VARCHAR2  := FND_API.G_FALSE,
  P_Commit                 IN    VARCHAR2  := FND_API.G_FALSE,
  p_validation_level       IN    NUMBER    := FND_API.G_VALID_LEVEL_FULL,
  P_Check_Access_Flag      IN    VARCHAR2  := FND_API.G_MISS_CHAR,
  P_Admin_Flag             IN    VARCHAR2  := FND_API.G_MISS_CHAR,
  P_Admin_Group_Id        IN    NUMBER     := FND_API.G_MISS_NUM,
  P_identity_salesforce_id IN    NUMBER     := FND_API.G_MISS_NUM,

```

```

P_Sales_Lead_Profile_Tbl    IN    AS_UTILITY_PUB.Profile_Tbl_Type := AS_
UTILITY_PUB.G_MISS_PROFILE_TBL,
P_SALES_LEAD_LINE_Tbl      IN    SALES_LEAD_LINE_Tbl_Type,
X_SALES_LEAD_LINE_OUT_Tbl  OUT   SALES_LEAD_LINE_OUT_Tbl_Type,
X_Return_Status            OUT   VARCHAR2,
X_Msg_Count                OUT   NUMBER,
X_Msg_Data                 OUT   VARCHAR2
);

```

Current Version

2.0

D.6.2 Parameter Descriptions

Notes

- In P_SALES_LEAD_LINE_REC, the required parameters are sales_lead_id, source_promotion_id and either product category (interest_type_id, primary_interest_code_id, secondary_interest_code_id) or inventory item (inventory_item_id and organization_id).
- last_update_date must be passed in for each sales lead line being updated.

[Table D-7](#) describes the IN parameters for the Update Sales Lead Lines procedure.

Table D-7 IN Parameters

Parameter	Data Type	Required	Description
P_Sales_Lead_Profile_Tbl	AS_UTILITY_PUB.Profile_Tbl_Type	No	Data type to store the access security related profile values (they can be cached in the mid tier and passed to the API)
P_Sales_lead_line_tbl	AS_SALES_LEADS_PUB.SALES_LEAD_LINE_TBL_TYPE	No	Table of sales lead line records

[Table D-8](#) describes the OUT parameters for the Update Sales Lead Lines procedure.

Table D-8 OUT Parameters

Parameter	Data Type	Description
X_sales_lead_line_out_tbl	AS_SALES_LEADS_PUB.SALES_LEAD_LINE_OUT_TBL_TYPE	Table of sales lead line out record types. Each record type consists of the sales lead line ID processed and the return status.

D.7 Delete Sales Lead Lines

This procedure deletes one or more sales lead lines.

D.7.1 Procedure Specification

```

PROCEDURE Delete_sales_lead_lines(
    P_Api_Version_Number      IN    NUMBER,
    P_Init_Msg_List           IN    VARCHAR2    := FND_API.G_FALSE,
    P_Commit                  IN    VARCHAR2    := FND_API.G_FALSE,
    p_validation_level        IN    NUMBER       := FND_API.G_VALID_LEVEL_FULL,
    P_Check_Access_Flag       IN    VARCHAR2    := FND_API.G_MISS_CHAR,
    P_Admin_Flag              IN    VARCHAR2    := FND_API.G_MISS_CHAR,
    P_Admin_Group_Id         IN    NUMBER       := FND_API.G_MISS_NUM,
    P_identity_salesforce_id  IN    NUMBER       := FND_API.G_MISS_NUM,
    P_Sales_Lead_Profile_Tbl  IN    AS_UTILITY_PUB.Profile_Tbl_Type := AS_
UTILITY_PUB.G_MISS_PROFILE_TBL,
    P_SALES_LEAD_LINE_Tbl    IN    SALES_LEAD_LINE_Tbl_type,
    X_SALES_LEAD_LINE_OUT_Tbl OUT  SALES_LEAD_LINE_OUT_Tbl_Type,
    X_Return_Status          OUT  VARCHAR2,
    X_Msg_Count              OUT  NUMBER,
    X_Msg_Data               OUT  VARCHAR2
);

```

Current Version

2.0

D.7.2 Parameter Descriptions

Note: In P_SALES_LEAD_LINE_REC, the required parameters are sales_lead_line_id.

Table D-9 describes the IN parameters for the Delete Sales Lead Lines procedure.

Table D-9 IN Parameters

Parameter	Data Type	Required	Description
P_Sales_Lead_Profile_Tbl	AS_UTILITY_PUB.Profile_Tbl_Type	No	Data type to store the access security related profile values (they can be cached in the mid tier and passed to the API)
P_Sales_lead_line_tbl	AS_SALES_LEADS_PUB.SALES_LEAD_LINE_TBL_TYPE	No	Table of sales lead line records

Table D-10 describes the OUT parameters for the Delete Sales Lead Lines procedure.

Table D-10 OUT Parameters

Parameter	Data Type	Description
X_sales_lead_line_out_tbl	AS_SALES_LEADS_PUB.SALES_LEAD_LINE_OUT_TBL_TYPE	Table of sales lead line out record types. Each record type consists of the sales lead line ID processed and the return status.

D.8 Lead Process After Create

This API should be called after lead header, lead line, lead contact, lead notes are created. This is a wrapper of the Run Lead Engines and the Build Lead Sales Team APIs. It calls the above APIs based on specific logic to maintain lead integrity, and launches the workflow process if users want to monitor the lead.

If the Lead Process After Create API is used, please skip the Run Lead Engines and Build Lead Sales Team. API sections.

D.8.1 Procedure Specification

```
PROCEDURE Lead_Process_After_Create (
    P_Api_Version_Number      IN  NUMBER,
    P_Init_Msg_List           IN  VARCHAR2      := FND_API.G_FALSE,
    p_Commit                  IN  VARCHAR2      := FND_API.G_FALSE,
    p_Validation_Level        IN  NUMBER        :=
FND_API.G_VALID_LEVEL_FULL,
```

```

P_Check_Access_Flag      IN  VARCHAR2      := FND_API.G_MISS_CHAR,
p_Admin_Flag             IN  VARCHAR2      := FND_API.G_MISS_CHAR,
P_Admin_Group_Id        IN  NUMBER         := FND_API.G_MISS_NUM,
P_identity_salesforce_id IN  NUMBER         := FND_API.G_MISS_NUM,
P_Salesgroup_id         IN  NUMBER         := FND_API.G_MISS_NUM,
P_Sales_Lead_Id         IN  NUMBER,
X_Return_Status         OUT NOCOPY VARCHAR2,
X_Msg_Count             OUT NOCOPY NUMBER,
X_Msg_Data              OUT NOCOPY VARCHAR2
)

```

Current Version

2.0

D.8.2 Parameter Descriptions

Notes

- Api_version_number will be set to 2.0.
- If p_salesgroup_id is not passed in, this API will find a group_id for the current user.
- If a flag column is passed in, check if it is 'Y' or 'N'. Raise exception for invalid flag.
- If a flag column is not passed in, default it to 'Y' or 'N'.

[Table D-11](#) describes the IN parameters for the Lead Process After Create API.

Table D-11 IN Parameters

Parameter	Data Type	Required	Description
P_sales_lead_id	NUMBER	Yes	Sales Lead Identifier.

Note: This procedure does not have OUT parameters.

D.9 Lead Process After Update

This API should be called after lead header, lead line, lead contact, lead notes are updated. This is a wrapper of the Run Lead Engines and the Rebuild Lead Sales Team APIs. It calls the above APIs based on specific logic to maintain lead integrity.

If the Lead Process After Update API is used, please skip the sections for the Run Lead Engines and the Rebuild Lead Sales Team APIs.

D.9.1 Procedure Specification

```

PROCEDURE Lead_Process_After_Update (
  P_Api_Version_Number      IN  NUMBER,
  P_Init_Msg_List           IN  VARCHAR2      := FND_API.G_FALSE,
  p_Commit                  IN  VARCHAR2      := FND_API.G_FALSE,
  p_Validation_Level        IN  NUMBER        :=
FND_API.G_VALID_LEVEL_FULL,
  P_Check_Access_Flag       IN  VARCHAR2      := FND_API.G_MISS_CHAR,
  p_Admin_Flag              IN  VARCHAR2      := FND_API.G_MISS_CHAR,
  P_Admin_Group_Id          IN  NUMBER        := FND_API.G_MISS_NUM,
  P_identity_salesforce_id  IN  NUMBER        := FND_API.G_MISS_NUM,
  P_Salesgroup_id           IN  NUMBER        := FND_API.G_MISS_NUM,
  P_Sales_Lead_Id           IN  NUMBER,
  X_Return_Status           OUT NOCOPY VARCHAR2,
  X_Msg_Count               OUT NOCOPY NUMBER,
  X_Msg_Data                OUT NOCOPY VARCHAR2
)

```

Current Version

2.0

D.9.2 Parameter Descriptions

Notes

- `p_api_version_number` will be set to 2.0.
- If `p_salesgroup_id` is not passed in, this API will find a `group_id` for the current user.
- If a flag column is passed in, check if it is 'Y' or 'N'. Raise exception for invalid flag.
- If a flag column is not passed in, default it to 'Y' or 'N'.

[Table D-12](#) describes the IN parameters for the Lead Process After Update API.

Table D–12 IN Parameters

Parameter	Data Type	Required	Description
P_sales_lead_id	NUMBER	Yes	Sales Lead Identifier.

Note: This procedure does not have OUT parameters.

D.10 Run Lead Engines

This API should be called after lead is created, or Run Engine button is clicked. If user doesn't specify qualified flag, rank, or sales channel, and profile setting is to do them automatically, this API will run qualification engine, rating engine, and channel selection engine.

D.10.1 Procedure Specification

```

PROCEDURE Run_Lead_Engines (
    P_Api_Version_Number      IN  NUMBER,
    P_Init_Msg_List           IN  VARCHAR2 := FND_API.G_FALSE,
    p_Commit                  IN  VARCHAR2 := FND_API.G_FALSE,
    p_Validation_Level        IN  NUMBER := FND_API.G_VALID_LEVEL_FULL,
    P_Admin_Group_Id          IN  NUMBER := FND_API.G_MISS_NUM,
    P_identity_salesforce_id  IN  NUMBER := FND_API.G_MISS_NUM,
    P_Salesgroup_id           IN  NUMBER := FND_API.G_MISS_NUM,
    P_Sales_Lead_Id           IN  NUMBER,
    X_Lead_Engines_Out_Rec    OUT  LEAD_ENGINES_OUT_Rec_Type,
    X_Return_Status           OUT  VARCHAR2,
    X_Msg_Count               OUT  NUMBER,
    X_Msg_Data                 OUT  VARCHAR2
);

```

Current Version

2.0

D.10.2 Parameter Descriptions

Notes

- `api_version_number` will be set to 2.0.

- If p_salesgroup_id is not passed in, this API will find a group_id for the current user.
- If a flag column is passed in, check if it is 'Y' or 'N'. Raise exception for invalid flag.
- If a flag column is not passed in, default it to 'Y' or 'N'.

Table D–13 describes the IN parameters for the Run Lead Engines API.

Table D–13 IN Parameters

Parameter	Data Type	Required	Description
P_sales_lead_id	NUMBER	Yes	Sales Lead Identifier that user wants to build sales team for

Table D–14 describes the OUT parameters for the Run Lead Engines API.

Table D–14 OUT Parameters

Parameter	Data Type	Description
X_Lead_Engines_Out_Rec	LEAD_ENGINES_OUT_Rec_Type	Result of qualification, rating, and channel selection engines.

D.11 Build Lead Sales Team

This API should be called after Run_Lead_Engines API is called. It builds lead sales team based on territory definition and adds lead creator as one of lead sales team member.

D.11.1 Procedure Specification

```

PROCEDURE Build_Lead_Sales_Team(
  P_Api_Version_Number      IN  NUMBER,
  P_Init_Msg_List           IN  VARCHAR2 := FND_API.G_FALSE,
  p_Commit                  IN  VARCHAR2 := FND_API.G_FALSE,
  p_Validation_Level        IN  NUMBER := FND_API.G_VALID_LEVEL_FULL,
  P_Admin_Group_Id          IN  NUMBER := FND_API.G_MISS_NUM,
  P_identity_salesforce_id  IN  NUMBER := FND_API.G_MISS_NUM,
  P_Salesgroup_id           IN  NUMBER := FND_API.G_MISS_NUM,
  P_Sales_Lead_Id           IN  NUMBER,
  X_Return_Status           OUT  VARCHAR2,
  X_Msg_Count               OUT  NUMBER,

```

```
X_Msg_Data          OUT VARCHAR2
);
```

Current Version

2.0

D.11.2 Parameter Descriptions

Notes

- Api_version_number will be set to 2.0.
- If p_salesgroup_id is not passed in, this API will find a group_id for the current user.
- If a flag column is passed in, check if it is 'Y' or 'N'. Raise exception for invalid flag.
- If a flag column is not passed in, default it to 'Y' or 'N'.

Table D–15 describes the IN parameters for the Build Lead Sales Team API.

Table D–15 IN Parameters

Parameter	Data Type	Required	Description
P_sales_lead_id	NUMBER	Yes	Sales Lead Identifier that user wants to build sales team for

Note: This procedure does not have OUT parameters.

D.12 Rebuild Lead Sales Team

This API should be called after lead header is updated and lines are created/updated/deleted. When user does change to the lead, the lead may not match the territory it originally met, and match other territories. This API will rebuild lead sales team to reflect the latest lead information.

D.12.1 Procedure Specification

```
PROCEDURE Rebuild_Lead_Sales_Team(
  P_Api_Version_Number  IN  NUMBER,
  P_Init_Msg_List       IN  VARCHAR2 := FND_API.G_FALSE,
```

```

p_Commit                IN  VARCHAR2 := FND_API.G_FALSE,
p_Validation_Level      IN  NUMBER := FND_API.G_VALID_LEVEL_FULL,
P_Admin_Group_Id        IN  NUMBER := FND_API.G_MISS_NUM,
P_identity_salesforce_id IN  NUMBER := FND_API.G_MISS_NUM,
P_Salesgroup_id         IN  NUMBER := FND_API.G_MISS_NUM,
P_Sales_Lead_Id         IN  NUMBER,
X_Return_Status         OUT VARCHAR2,
X_Msg_Count             OUT NUMBER,
X_Msg_Data              OUT VARCHAR2
);

```

Current Version

2.0

D.12.2 Parameter Descriptions

Notes

- Api_version_number will be set to 2.0.
- If p_salesgroup_id is not passed in, this API will find a group_id for the current user.
- If a flag column is passed in, check if it is 'Y' or 'N'. Raise exception for invalid flag.
- If a flag column is not passed in, default it to 'Y' or 'N'.

Table D-16 describes the IN parameters for the Rebuild Lead Sales Team API.

Table D-16 IN Parameters

Parameter	Data Type	Required	Description
P_sales_lead_id	NUMBER	Yes	Sales Lead Identifier that user wants to build sales team for

Note: This procedure does not have OUT parameters.

D.13 Update Sales Lead Contacts

This API is used to update sales contact information in the table. The API will raise an exception if the record matching the Sales Lead Contact ID and Object Version Number passed does not exist.

D.13.1 Procedure Specification

```

PROCEDURE Update_sales_lead_contacts(
    P_Api_Version_Number      IN    NUMBER,
    P_Init_Msg_List           IN    VARCHAR2      := FND_API.G_FALSE,
    P_Commit                  IN    VARCHAR2      := FND_API.G_FALSE,
    p_validation_level        IN    NUMBER         := FND_API.G_VALID_LEVEL_
FULL,
    P_Check_Access_Flag       IN    VARCHAR2      := FND_API.G_MISS_CHAR,
    P_Admin_Flag              IN    VARCHAR2      := FND_API.G_MISS_CHAR,
    P_Admin_Group_Id          IN    NUMBER         := FND_API.G_MISS_NUM,
    P_identity_salesforce_id  IN    NUMBER         := FND_API.G_MISS_NUM,
    P_Sales_Lead_Profile_Tbl  IN    AS_UTILITY_PUB.Profile_Tbl_Type := AS_
UTILITY_PUB.G_MISS_PROFILE_TBL,
    P_SALES_LEAD_CONTACT_Tbl  IN    SALES_LEAD_CONTACT_Tbl_Type,
    X_SALES_LEAD_CNT_OUT_Tbl  OUT   SALES_LEAD_CNT_OUT_Tbl_Type,
    X_Return_Status           OUT   VARCHAR2,
    X_Msg_Count               OUT   NUMBER,
    X_Msg_Data                OUT   VARCHAR2
);

```

Current Version

1.0

D.13.2 Parameter Descriptions

Notes

Raise an exception if the object_version_number does not match.

[Table D-17](#) describes the IN parameters for the Update Sales Lead Contacts API.

Table D-17 IN Parameters

Parameter	Data Type	Required	Description
p_api_version	Number	Y	Caller version number. This will be compared against the API version number to detect incompatibility .
p_init_msg_list	VarChar2	N	Flag to indicate if the message stack should be initialized. Default : FND_API.g_false.
p_commit	VarChar2	N	Flag to indicate if the changes should be committed on success. Default : FND_API.g_false.

Table D–17 IN Parameters

Parameter	Data Type	Required	Description
p_validation_level	Number	N	Level of validation required. NONE means no validation will be done in the API and FULL means all the validations (item level, record level) will be performed.
p_check_access_flag	VarChar2	N	-
p_admin_flag	VarChar2	N	-
p_admin_group_id	VarChar2	N	-
p_identity_salesforce_id	Number	N	-
p_sales_lead_profile_tbl	AS_UTILITY_PUB.Profile_tbl_type	N	-
P_sales_lead_id	Number	Y	The unique identifier of the sales lead.

[Table D–18](#) describes the OUT parameters for the Update Sales Lead Contacts API.

Table D–18 OUT Parameters

Parameter	Data Type	Description
x_return_status	VarChar2	See Section D.2.2, "Standard OUT Parameters" .
x_msg_count	Number	See Section D.2.2, "Standard OUT Parameters" .
x_msg_data	VarChar2	See Section D.2.2, "Standard OUT Parameters" .
X_sales_lead_cnt_out_tbl	sales_Lead_cnt_out_tbl_type	Contains the record which contains the Lead Contact ID for the sales lead.

D.14 Delete Sales Lead Contacts

This API calls the table handler `Delete_Sales_Lead_Contacts` and then calls a procedure to update the `AS_SALES_LEAD_CONTACT` table, if the primary contact is deleted and other contact is marked as primary.

D.14.1 Procedure Specification

```

PROCEDURE Delete_sales_lead_contacts(
    P_Api_Version_Number      IN    NUMBER,
    P_Init_Msg_List          IN    VARCHAR2      := FND_API.G_FALSE,
    P_Commit                 IN    VARCHAR2      := FND_API.G_FALSE,
    p_validation_level       IN    NUMBER        := FND_API.G_VALID_LEVEL_
FULL,
    P_Check_Access_Flag      IN    VARCHAR2      := FND_API.G_MISS_CHAR,
    P_Admin_Flag             IN    VARCHAR2      := FND_API.G_MISS_CHAR,
    P_Admin_Group_Id        IN    NUMBER         := FND_API.G_MISS_NUM,
    P_identity_salesforce_id IN    NUMBER         := FND_API.G_MISS_NUM,
    P_Sales_Lead_Profile_Tbl IN    AS_UTILITY_PUB.Profile_Tbl_Type := AS_
UTILITY_PUB.G_MISS_PROFILE_TBL,
    P_SALES_LEAD_CONTACT_Tbl IN    SALES_LEAD_CONTACT_Tbl_Type,
    X_SALES_LEAD_CNT_OUT_Tbl OUT    SALES_LEAD_CNT_OUT_Tbl_Type,
    X_Return_Status         OUT    VARCHAR2,
    X_Msg_Count             OUT    NUMBER,
    X_Msg_Data              OUT    VARCHAR2
);

```

Current Version

1.0

D.14.2 Parameter Descriptions

Notes

Raise an exception if the object_version_number does not match.

[Table D-19](#) describes the IN parameters for the Delete Sales Lead Contacts API.

Table D-19 IN Parameters

Parameter	Data Type	Required	Description
p_api_version	Number	Y	Caller version number. This will be compared against the API version number to detect incompatibility.
p_init_msg_list	VarChar2	N	Flag to indicate if the message stack should be initialized. Default : FND_API.g_false.
p_commit	VarChar2	N	Flag to indicate if the changes should be committed on success. Default : FND_API.g_false.

Table D–19 IN Parameters

Parameter	Data Type	Required	Description
p_validation_level	Number	N	Level of validation required. NONE means no validation will be done in the API and FULL means all the validations (item level, record level) will be performed.
p_check_access_flag	VarChar2	N	-
p_admin_flag	VarChar2	N	-
p_admin_group_id	VarChar2	N	-
p_identity_salesforce_id	Number	N	-
p_sales_lead_profile_tbl	AS_UTILITY_PUB.Profile_tbl_type	N	-
P_sales_lead_id	Number	Y	The unique identifier of the sales lead.

Table D–20 describes the OUT parameters for the Delete Sales Lead Contacts API.

Table D–20 OUT Parameters

Parameter	Data Type	Description
x_return_status	VarChar2	See Section D.2.2, "Standard OUT Parameters" .
x_msg_count	Number	See Section D.2.2, "Standard OUT Parameters" .
x_msg_data	VarChar2	See Section D.2.2, "Standard OUT Parameters" .
X_sales_lead_cnt_out_tbl	sales_Lead_cnt_out_tbl_type	Contains the record which contains the Lead Contact ID for the sales lead.

Glossary

Acquisition

Acquisition is the part of the DQM matching process that matches input record attributes against the attributes in the staged schema to get a smaller group of records that form the work unit. This process narrows down the records that can be scored in the scoring part of the matching process.

Attribute

An attribute corresponds to a column in a TCA registry table, and the attribute value is the value that is stored in the column. For example, party name is an attribute and the actual values of party names are stored in a column in the HZ_PARTIES table.

Control File

A control file is used to give instructions to SQL*Loader.

Concurrent Process

A task in the process of completing. Each time you submit a task, you create a new concurrent process. A concurrent process runs simultaneously with other concurrent processes (and other activities on your computer) to help you complete multiple tasks at once with no interruptions to your terminal.

CSV

Stands for Comma Separated Variable. A file format used to transfer basic data between databases and spreadsheets. Each line (up to the carriage return) is considered a record. Fields within each record are divided by a comma. Each line must have the same number of fields (commas). If a comma or leading and/or

trailing blanks appear in any field value, the field must be enclosed by quotes (") to indicate the information is data and not a field divider.

Deduplication

Deduplication is identifying identical records from a list. For example, you will deduplicate the records in the `AS_SALES_LEAD` table to avoid multiple occurrences of a lead record.

DQM

Stands for Data Quality Management. DQM manages duplicate parties in TCA. Parties are entities, of type Person, Organization, or Relationship, that can enter into business relationships. Party information includes the party name, addresses, contacts, and contact points.

DTD

Stands for Document Type Definition. The purpose of a DTD is to define the legal building blocks of an XML document. It defines the document structure with a list of legal elements.

Flat File

Flat file is a tilde (~) delimited text file with data to be imported as leads.

FTP

Stands for File Transfer Protocol. FTP is the protocol used on the Internet for exchanging files. FTP uses the Internet's TCP/IP protocols to enable data transfer.

Grading

See Rating.

Match Rule

A match rule is a set of rules that determine the records that are selected and displayed as matches for the input record. A match rule consists of acquisition attributes that are used for matching and can also include scoring attributes to score the matched records.

Organization Person

An organization person is one who acts on behalf of or in the context of an organization.

Party

A party is a person, organization, or collection of parties that can enter into relationships with other parties.

Ranking

See Rating

Rating

The rating assigned to a lead by the Rating Engine based on the lead attributes. Examples of rating are Hot Lead, Cold Lead, and so on.

TCA

Stands for Trading Community Architecture. TCA is the Oracle Customer Model where all customer records are stored. It is a common repository which is accessed by the Oracle E-Business Suite and ERP applications.

URL

Stands for Uniform Resource Locator. It is the World Wide Web address of a site on the Internet.

User Hook

User hook is part of the source code of an application that is exposed for the purpose of customization.

XML

Stands for Extensible Markup Language. XML is a metalanguage --a language for describing other languages--which lets you design your own customized markup languages for limitless different types of documents.

Index

A

APIs

- build lead sales team, D-2
- create sales lead, D-2
- delete sales lead contacts, D-3
- delete sales lead lines, D-2
- lead process after create, D-2
- lead process after update, D-2
- public, D-1
- rebuild lead sales team, D-3
- run lead engines, D-2
- update sales lead, D-2
- update sales lead contacts, D-3
- update sales lead lines, D-2

AS_IMPORT_INTERFACE

- delete, 3-56
- import leads, 3-4
- purge, 3-56

AS_LEAD_IMPORT_ERRORS

- errors, 3-6

assignment

- automatic, 5-15
- immature, 5-16
- profiles, A-9
- setting up, 5-14

attributes, 1-5

- custom, 5-23
- derived, 1-5
- monitoring engine, 6-3
- scope of monitor, 6-3

B

best practices, 5-9

business event

- definition, 3-50

business flow, 2-2

- capturing, 2-2
- cleaning leads, 2-2
- processing leads, 2-4

C

channel selection

- functioning, 5-7
- overview, 5-7
- setting up, 5-8

channels

- setting up, 5-8

concurrent program

- delete leads, 3-56
- flat file, 3-8
- import sales lead, 3-21
- interaction matching engine, 4-7
- process imported leads, 3-13
- staging schema, 3-30
- workflow background process, 6-6

conditions

- monitoring engine, 6-9

creating

- monitor rule, 6-7
- conditions, 6-9
- rule sets, 5-9

currency

- codes, 3-22

- profiles, A-16
- custom attributes
 - setting up, 5-23
- custom code
 - business events, 3-50
 - execution control, 3-51
 - handling errors, 3-51
 - post, 3-50
 - pre, 3-50
 - routing leads, 5-18
 - sample, 3-52
 - user hook, 3-55
- customize
 - time frames, 5-24

D

- data quality, 3-24
 - customer, 3-25
 - DQM
 - duplicate contact points, 3-28
 - leads, 3-43
 - OSR, 3-25
- dedupe, 1-4
 - custom deduplication, 3-47
 - leads, 3-43
 - rule-based, 1-4
- deduplication rule, 3-43
 - attributes, 3-44
 - customize, 3-45
 - flow, 3-46
 - seeded, 3-44
- definitions
 - business event, 3-50
 - event subscription, 3-50
 - guards, 5-2
 - precedence, 5-3
 - rules, 5-4
- delete
 - leads, 3-56
 - unqualified leads, 5-14
 - impact, 5-14
- derived attributes, 5-23
- disqualify leads, 5-4
 - delete, 5-14

- dqm
 - customer, 3-25
 - duplicate addresses, 3-27
 - duplicate contacts, 3-28
 - duplicate customers, 3-27
 - logic, 3-29
 - matching rules, 3-15, 3-30
 - procedure, 3-16
 - prior user hook, 3-54
 - setting staging schema, 3-30

E

- event description
 - definition, 3-50

F

- find
 - monitor, 6-10
 - rule set, 5-12
- flexfields, 3-10
 - reference, 3-11
- followup
 - profiles, A-11
- functioning
 - channel selection, 5-7
 - interaction mining engine, 4-4
 - monitoring engine, 6-3
 - qualification engine, 5-5
 - rating engine, 5-6

G

- generating lead, 4-6
 - process, 4-6
- guards
 - definition, 5-2

I

- immature assignment
 - setting up, 5-16
- import
 - correct errors, 3-7

- csv, txt, 3-3, 3-4
- errors, 3-6
- flat file, 3-4, 3-7
 - concurrent program, 3-8
 - concurrent program prerequisites, 3-8
 - concurrent program table, 3-8
 - run SQL loader, 3-9
 - run SQL loader prerequisites, 3-9
 - sample file, 3-10
- HTML UI, 3-4
- loading tables, 3-20
- parallel, 3-22
 - child process, 3-23
 - parent process, 3-23
- performance, 3-22
- process, 3-3
- profiles, A-5
- tables, 3-11
- import errors
 - AS_LEAD_IMPORT_ERRORS, 3-6
- import sales lead, 3-13
 - creating lookups, 3-19
 - currency codes, 3-22
 - custom code, 3-50
 - dqm rules, 3-15
 - flow, 3-13
 - leading interface tables, 3-20
 - limitation, 3-23
 - parameters, 3-21
 - performance, 3-22
 - prerequisites, 3-15
 - running, 3-21
 - setting profiles, 3-18
 - setting territories, 3-20
 - user hook, 3-54
- interaction matching engine
 - concurrent program, 4-7
 - functioning, 4-4
 - creating lead, 4-6
 - linking existing leads, 4-5
 - mining, 4-4
 - source code, 4-5
 - generating lead, 4-6
 - generating lead process, 4-6
 - interaction score threshold, 4-3

- interaction types, 4-2
 - overview, 4-2
 - profiles, A-6
 - report, 7-6
 - scores, 4-2
 - setting interaction types, 4-7
- interaction types
 - setting, 4-7

L

- lead source, 3-2
 - campaigns, 3-2
 - interactions, 3-2
 - Oracle iStore, 3-2
 - referrals, 3-2
 - scripting, 3-2
- lead status
 - setting up, 5-22
- leads
 - assignment, 5-14
 - data quality, 3-43
 - dedupe, 3-43
 - delete, 3-56
- leads assignment
 - profiles, A-9
- leads engines
 - best practices, 5-9
 - channel selection, 5-7
 - overview, 5-2
 - profiles, A-7
 - qualification engine, 5-4
 - rating engine, 5-5
 - reports
 - rule flow, 5-12
 - rule flow, 5-12
- log
 - monitor, 6-10

M

- matching rules
 - duplicate contacts, 3-39
 - duplicate parties, 3-31
 - duplicate persons, 3-35

- samples, 3-30
- monitoring engine
 - absolute monitor, 6-2
 - attributes, 6-3
 - conditions, 6-2, 6-9
 - assignment state, 6-2
 - creation date, 6-2
 - creating a monitor rule, 6-7
 - expiration date, 6-4
 - functioning, 6-3
 - monitor log, 6-10
 - overview, 6-2
 - relative monitor, 6-2
 - scope, 6-3
 - search, 6-10
 - time frame, 6-4
 - timeout, 6-5
 - upgrade, 1-6, 6-4
 - workflow background process concurrent program, 6-6

O

- obsolete
 - profiles, A-16
- Oracle Leads Management, 1-3
 - dependencies, 1-6
 - enhancements, 1-4
 - features, 1-3
 - new features, 1-3
 - profiles, A-3
 - upgrading, 1-6
- OSR
 - data quality, 3-25
 - logic, 3-26
 - recommendation, 3-27
- overview
 - channel selection, 5-7
 - interaction matching engine, 4-2
 - leads engine, 5-2
 - monitoring engine, 6-2
 - Oracle Leads Management, 1-2
 - qualification engine, 5-4
 - rating engine, 5-5
 - role of lead, 1-2

- time frames, 5-24

P

- performance
 - import, 3-22
- precedence
 - definition, 5-3
- profiles, A-3
 - default, A-4
 - interaction matching engine, A-6
 - lead import, A-5
 - leads assignment, A-9
 - leads conversion, A-12
 - leads engines, A-7
 - leads followup, A-11
 - leads security, A-13
 - multiple currency, A-16
 - obsolete, A-16
 - setting, A-2
 - telesales, A-14
- purge
 - AS_IMPORT_INTERFACE, 3-56
 - staged leads, 3-56

Q

- qualification engine, 1-4
 - best practices, 5-9
 - disqualify leads, 1-4, 5-4
 - functioning, 5-5
 - overview, 5-4
 - qualify leads, 5-4
 - recommendation, 5-4
- qualify leads, 5-4

R

- rating
 - setting up, 5-6
- rating engine
 - functioning, 5-6
 - overview, 5-5
 - ratings, 5-6
- reports

- import records manager, 7-2
- lead linking, 7-6
- lead processing history, 7-2
- rule diagnostic manager, 7-5
- rule flows, 5-12
- rule performance, 7-4
- routing leads
 - user hook, 5-18
- rule sets, 1-5
 - campaign-specific, 1-5
 - creating, 5-9
 - finding, 5-12
 - searching, 5-12
- rules
 - definition, 5-4
- running
 - import sales lead concurrent program, 3-21

S

- sample
 - custom code, 3-52
 - matching rules, 3-30
 - user hook, 3-55
- scenarios, 2-6
 - disqualifying leads, 2-14
 - qualifying leads, 2-6
- schema
 - concurrent program, 3-30
 - dqm, 3-30
- scope
 - monitoring engine, 6-3
- security
 - profiles, A-13
- seeded
 - lookups
 - lead rank, B-7
 - lead status, B-7
- setting
 - profiles, A-2
- setting up
 - channels, 5-8
 - custom attributes, 5-23
 - lead status, 5-22
 - leads assignment, 5-14

- ratings, 5-6
- time frames, 5-24
- staged leads
 - delete, 3-56
- status
 - seeded, B-7

T

- telesales
 - profiles, A-14
- time frames
 - customizing, 5-24
 - expiration date, 6-4
 - monitoring engine, 6-4
 - overview, 5-24
 - setting up, 5-24
- timeout
 - monitoring engine, 6-5

U

- unqualified leads
 - purge, 5-14
- upgrade, 1-6
 - monitoring engine, 1-6, 6-4
 - single product catalog, 1-6
- user hook, 3-54
 - dedupe, 3-47
 - performance, 3-55
 - prior to DQM, 3-54
 - purpose, 3-54
 - recommendation, 3-55
 - routing leads, 5-18
 - seeded code, 3-55

