

Oracle® Trading Community Architecture Data Quality Management

User Guide

Release 11*i*

April, 2002

Part No. A97626-01

ORACLE®

Copyright © 2002, Oracle Corporation. All rights reserved.

Contributors: Charlie Ahern, Carol Chow, Charity Kocher, Shannon Kreps, Essan Ni

The Programs (which include both the software and documentation) contain proprietary information of Oracle Corporation; 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.

Program Documentation is licensed for use solely to support the deployment of the Programs and not for any other purpose.

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. Oracle Corporation does not warrant that this document is 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, without the express written permission of Oracle Corporation.

If the Programs are delivered to the U.S. Government or anyone licensing or using the programs on behalf of the U.S. Government, the following notice is applicable:

Restricted Rights Notice Programs delivered subject to the DOD FAR Supplement are commercial computer software and use, duplication and disclosure of the Programs including documentation, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement. Otherwise, Programs delivered subject to the Federal Acquisition Regulations are restricted computer software and use, duplication and disclosure of the Programs shall be subject to the restrictions 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 licensee's responsibility to take all appropriate fail-safe, back up, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and Oracle disclaims liability for any damages caused by such use of the Programs.

Oracle is a registered trademark, and Oracle8i, Oracle9i, Oracle MetaLink, Oracle Store, PL/SQL, Pro*C, SQL*Net, and SQL*Plus are trademarks or registered trademarks of Oracle Corporation. Other names may be trademarks of their respective owners.

Contents

Send Us Your Comments	vii
Preface.....	ix
Documentation Accessibility	xi
Online Documentation	xii
Related Guides	xii
Guides Related to All Products	xiii
Guides Related to This Product	xiii
Other Implementation Documentation	xv
1 Overview	
Introduction	1-2
Major Features.....	1-3
Attribute Search in Logical Entities	1-3
Predefined and Customizable Search and Match Functionality	1-3
Staged Schema and <i>interMedia</i> Indexes	1-5
Duplicate Party Identification	1-5
Examples.....	1-6
Data Quality Management Process	1-9
Setting Up and Using Data Quality Management	1-11
2 Attributes, Transformation Functions, and Word Replacements	
Attributes Overview.....	2-2
Transformation Functions Overview	2-3

Defining Attributes and Transformation Functions.....	2-5
Creating Custom Attributes.....	2-8
Creating Custom Transformation Functions.....	2-11
Attribute and Transformation Function Tips.....	2-13
Word Replacements Overview	2-14
Creating a Word Replacement List.....	2-16
Word Replacements Tips.....	2-18

3 Match Rules

Match Rules Overview	3-2
The Matching Process.....	3-3
Setting Up the Match Rule	3-5
Defining the Acquisition Component of a Match Rule.....	3-6
Prerequisites.....	3-7
Defining the Scoring Component of a Match Rule.....	3-10
Prerequisites.....	3-11
DQM Compile All Rules Program	3-15
Prerequisites.....	3-15
Match Rule Example	3-16
Preparing for the Match Rule.....	3-16
Defining the Acquisition and Scoring Components.....	3-17
Setting the Match Threshold.....	3-19
Considering the Impact of Transformation Functions on Your Thresholds.....	3-20
Copying Match Rules	3-22
Editing Match Rules	3-24

4 Staged Schema and *interMedia* Index

Staged Schema and <i>interMedia</i> Index Overview	4-2
Staged Schema Features.....	4-2
<i>interMedia</i> Index Features.....	4-3
DQM Staging Program	4-4
Prerequisites.....	4-5
Program Parameters.....	4-5
DQM Index Optimization Program	4-6
Prerequisites.....	4-6

DQM Synchronization Program	4-7
Prerequisites	4-7
5 Batch Duplicate Identification	
Batch Duplicate Identification Overview	5-2
Batch Duplicate Identification Process	5-2
Prerequisites	5-4
Defining Duplicate Identification Batches.....	5-5
Reviewing Duplicates and Creating Merge Batches	5-8
Prerequisites	5-8
Submitting Merge Batches.....	5-13
Prerequisites	5-13
A Seed Data	
Seeded Attributes	A-2
Party Attributes.....	A-3
Address Attributes	A-11
Contact Attributes	A-13
Contact Point Attributes.....	A-15
Seeded Transformation Functions.....	A-17
Seeded Match Rules.....	A-25
SAMPLE: IDENTICAL_PERSON	A-26
SAMPLE: IDENTICAL_ORGANIZATIONS.....	A-27
SAMPLE: ADDRESS_ORGANIZATIONS	A-28
SAMPLE: ADDRESS_PERSONS.....	A-30
SAMPLE: SIMILAR_ORGANIZATION	A-32
SAMPLE: SIMILAR_PERSON	A-34
SAMPLE: SEARCH	A-36
B Standard Navigation Paths	
Standard Navigation Paths	B-2
C Profile Options	
Profile Options.....	C-2

Profile Options Summary	C-2
Profile Options in Data Quality Management	C-3

Glossary

Index

Send Us Your Comments

Oracle Trading Community Architecture Data Quality Management User Guide, Release 11*i*
Part No. A97626-01

Oracle Corporation 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: (801) 659-7164 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 Release 11i of the *Oracle® Trading Community Architecture Data Quality Management User Guide*.

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

- The principles and customary practices of your business area.
- Oracle Trading Community Architecture Data Quality Management

If you have never used Oracle Trading Community Architecture Data Quality Management, Oracle suggests you attend one or more of the Oracle Applications training classes available through Oracle University.

- 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 Trading Community Architecture Data Quality Management User Guide contains the information you need to understand and use Oracle Trading Community Architecture Data Quality Management. This user guide includes:

- Chapter 1 provides an overview of Oracle Trading Community Architecture Data Quality Management.
- Chapter 2 describes setting up and using attributes, transformation functions, and word replacement pairs used in the matching process.
- Chapter 3 describes setting up match rules to identify and score matches in the TCA registry.
- Chapter 4 describes the staged schema and *interMedia* indexes, including creating and updating both.
- Chapter 5 describes setting up and using batch duplicate identification to identify duplicates that exist within the TCA registry.
- Appendix A describes the seeded attributes, transformation functions, and match rules.
- Appendix B describes how to navigate to each window in Oracle Trading Community Architecture Data Quality Management.
- Appendix C lists the profile options that affect the operation of Oracle Trading Community Architecture Data Quality Management.
- A glossary provides definitions of terms specific to Oracle Trading Community Architecture Data Quality Management that are used in this guide.

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 Corporation 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.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle Corporation does not own or control. Oracle Corporation neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Other Information Sources

You can choose from many sources of information, including online documentation, training, and support services, to increase your knowledge and understanding of Oracle Trading Community Architecture Data Quality 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).

- **Online Help** - The new features section in the HTML help describes new features in 11*i*. This information is updated for each new release of Oracle Trading Community Architecture Data Quality Management. The new features section also includes information about any features that were not yet available when this guide was printed. For example, if your administrator has installed software from a mini-packs an upgrade, this document describes the new features. Online help patches are available on MetaLink.
- **11*i* Features Matrix** - This document lists new features available by patch and identifies any associated new documentation. The new features matrix document is available on MetaLink.
- **Readme File** - Refer to the readme file for patches that you have installed to learn about new documentation or documentation patches that you can download.

Related Guides

Oracle Trading Community Architecture Data Quality 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 Trading Community Architecture Data Quality 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) available with this release of Oracle Trading Community Architecture Data Quality Management (and any other Oracle Applications products). This guide also includes information on setting user profiles, as well as running and reviewing reports and concurrent processes.

Guides Related to This Product

Oracle Trading Community Architecture Party Merge User Guide

Use this user guide to learn how to merge parties and their related entities in the Oracle Trading Community Architecture registry. The Oracle Trading Community Architecture Party Merge User Guide describes how to set up and process party merge batches as well as how to identify merge errors.

Oracle *interMedia* User's Guide and Reference

This user guide and reference provides information about Oracle *interMedia*. This product enables Oracle9i to store, manage, and retrieve geographic location information, images, audio, video, or other heterogeneous media data in an integrated fashion with other enterprise information. Oracle Trading Community Architecture Data Quality Management uses *interMedia* indexes to facilitate search and matching.

Oracle Receivables User Guide

Use this user guide to learn how to implement flexible address formats for different countries. You can use flexible address formats in the suppliers, customers, banks, invoices, and payments windows in both Oracle Payables and Oracle Receivables. This user guide also explains how to set up your system, create transactions, and run reports in Oracle Receivables.

Oracle Receivables Tax Manual

This manual provides information about calculating tax within Oracle Receivables, Oracle Order Management, Oracle Sales and Marketing, and Oracle Web Customers. It includes information about implementation procedures, setup forms and windows, the Oracle Receivables tax calculation process, tax reports and listings, and tax-specific open interfaces.

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, the Oracle8 technology stack, and the Oracle8*i* Server 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.

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. It describes the Oracle Application Object Library components needed to implement the Oracle Applications user interface described in the *Oracle Applications User Interface Standards for Forms-Based Products*. It also provides information to help you build your custom Oracle Forms Developer 6i forms so that they 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 Guide

This guide explains how to define new workflow business processes as well as customize existing Oracle Applications-embedded workflow processes. You also use this guide to complete the setup steps necessary for any Oracle Applications product that includes workflow-enabled processes.

Oracle Applications Flexfields Guide

This guide provides flexfields planning, setup and reference information for the Oracle Trading Community Architecture Data Quality 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 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 Trading Community Architecture Data Quality 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 Trading Community Architecture Data Quality 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.

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 Corporation 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.

Your Feedback

Thank you for using Oracle Trading Community Architecture Data Quality Management and this user guide.

Oracle values your comments and feedback. In this guide is a reader's comment form that you can use to explain what you like or dislike about Oracle Trading Community Architecture Data Quality Management or this user guide. Mail your comments to the following address or call us directly at (650) 506-7000.

Oracle Applications Documentation Manager
Oracle Corporation
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Or, send electronic mail to appsdoc_us@oracle.com.

1

Overview

This chapter provides an overview of Oracle Trading Community Architecture Data Quality Management.

Introduction

Use Oracle Trading Community Architecture (TCA) Data Quality Management (DQM) to manage duplicate parties in your TCA registry. 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.

The TCA registry is the single source of party information for multiple Oracle E-Business Suite applications. These applications provide windows, online forms, batch data entry functionality, and other features for you to enter party information. Although Customer Relationship Management (CRM) and Enterprise Resource Planning (ERP) applications share the same registry, each application uses TCA differently depending on the context and need for particular party information.

Each application must quickly, accurately, and consistently retrieve information from the TCA registry for transaction processing. Duplicate data in the registry can reduce the efficiency and accuracy of party processing and reports.

DQM provides flexible features that let you identify duplicate entries, taking into account that information in the registry can contain typographical errors, spelling errors, or incomplete data. DQM copies the registry entries into a staged schema separate from the actual TCA registry and transforms and standardizes the staged schema records. With fuzzy searches through the transformed data in the staged schema, you can establish a set of possible duplicates that currently exist in your TCA registry.

After you determine duplicate entries in your registry, you can use the TCA Party Merge feature to merge the duplicate parties or party entities.

See also: *Oracle Trading Community Architecture Party Merge User Guide*

In addition to finding existing duplicates in your registry, you can use other Oracle applications that implement DQM to:

- Search for parties, addresses, contacts, and contact points in the TCA registry
- Prevent the entry of duplicates into the registry

Major Features

Data Quality Management provides various features that work together as a powerful and flexible data transformation and matching tool for the TCA registry. These features are integrated with TCA and can be used by any Oracle E-Business Suite application to search the registry and identify duplicates.

Attribute Search in Logical Entities

DQM uses attributes in these four logical entities, which are a subset of the TCA registry, to search and identify matches:

- Party
- Address
- Contact
- Contact point

Each attribute corresponds to a table column in the TCA registry. For example, party name is an attribute of the party logical entity, and the value for this attribute is obtained from the HZ_PARTIES table. The party name can be one attribute that you use to evaluate if two parties are duplicates or matches, for example John Smith and Jon Smith.

You can add custom attributes if the predefined attributes do not meet your needs.

Predefined and Customizable Search and Match Functionality

You can use predefined transformation functions and match rules to search and identify matching parties. Transformation functions transform party, address, contact, and contact point attribute values into standardized representations for high quality matching. In addition to the seeded transformation functions, DQM provides seeded word replacement lists in American English that transformation functions can optionally call upon. Match rules determine and evaluate the results of the matching process.

You can define and implement custom transformation functions to optimize the match rules for your environment and application. You can also set up your match rules to meet your business needs.

Transformation Functions and Word Replacements

Transformation functions transform data into representations that are more similar for matching purposes. For example, transformation functions can replace double letters with only one of the letters, as well as remove nonalphanumeric characters and vowels. A transformation function that removes vowels would change, for example, both values for an address attribute, 123 Main Rd and 123 Mane Road, into 123 MN RD. This transformation allows the two attributes to be determined a match, minimizing the effects of potential spelling errors, such as Mane, and data entry inconsistencies, such as Rd versus Road.

Transformation functions use word replacement lists which create synonyms that are treated as equivalents for searching and matching. You can use word replacements to standardize certain portions of your party data to minimize the effect of data entry inconsistencies on searches. For example, a word replacement transformation function can replace California, Calif, and ca with CA.

Match Rules

Match rules determine whether two parties should be identified as a match or potential duplicate. DQM match rules are applied when you run the DQM Duplicate Identification Program to find existing duplicates in the TCA registry.

Other Oracle applications that implement DQM can also apply match rules when you:

- Enter or update party information through windows, online forms, import interfaces, or APIs.
- Search for TCA registry information using search interfaces.

Match rules support score-based matching and consist of two parts, acquisition and scoring criteria. Acquisition provides an initial set of possible matches, and scoring assigns scores to attributes to further filter matches and rank the results.

When you define match rules, you can specify which attributes are important for matching and configure how the attributes should be evaluated. For example, you can define a match rule so that only records with the same transformed party name and party number attribute values are considered matches. You can also determine how matching or similar attributes are scored to produce a total score that you can use to evaluate a record. To each attribute that you use in a match rule, you also assign transformation functions.

Staged Schema and *interMedia* Indexes

The staged schema is a mirror of a portion of the TCA registry and contains data that transformation functions have cleansed and standardized. Matches are determined by comparing an input record against the transformed records in the staged schema, which is separate from the TCA registry. The registry itself is not transformed or searched at any time. The staged schema is shared across all Oracle E-Business Suite applications.

Searches through the staged schema are actually searches against the *interMedia* indexes that index the schema and speed up the search process. During the acquisition phase of matching, the indexes quickly limit the number of parties to evaluate for scoring. DQM provides four *interMedia* indexes, one for each logical entity.

You can run programs in DQM to generate or update the staged schema and the *interMedia* indexes.

Duplicate Party Identification

DQM lets you find duplicate entries that already exist in your TCA registry. Use the Submit Duplicate Identification Batch window to set up and run the DQM Duplicate Identification Program that applies match rules for identifying duplicates. The Duplicate Identification: Batch Review window displays matches for your review so that you can determine the entities that are in fact duplicates and submit the duplicates for merge.

Examples

These examples illustrate using Data Quality Management to identify existing duplicates, as well as how other Oracle applications can implement DQM so that you can search for parties or prevent duplicate entries.

Identifying Duplicate Parties

For a party called International Party Duplicates, the party name could have been entered in different ways, such as International Party Duplicates, Inc. or IPD.

You use a transformation function to remove all nonalphanumeric characters and terms such as Corp and Inc, and then normalize party names into three-letter acronyms in the staged schema. Define match rules and use the Submit Duplicate Identification Batch window to search for duplicates.

As a result, these entries are displayed as possible duplicates:

IPD	=	IPD
I.P.D.	=	IPD
IPD Corp.	=	IPD
Intl Party Duplicates	=	IPD
International Party Duplicates	=	IPD
International Party Duplicates, Inc.	=	IPD

Searching for Parties

Other Oracle applications that implement DQM can create search interface that you enter search criteria in. The Oracle application can create transformation functions, including one that uses the DQM PL/SQL function Soundex, and populate the results into the staged schema.

For example, you want to search for parties of type Person with the last name Smyth who are located in California. The last name could have been misspelled and entered in different ways, such as Smith or Smythe. You enter Smyth and California in the search interface, and the Oracle application applies a match rule that includes a filter for the state of California and searches through the staged schema for search results.

This table shows the original entries in your TCA registry.

Name	Location
Smith, James	Redwood Shores, California
Smith, Virginia	Seattle, Washington
Smyth, Edward	Los Angeles, California
Smyth, Helen	New York City, New York
Smythe, Joseph	Los Angeles, California

The Soundex representation for these names is:

Smith	S260
Smyth	S260
Smythe	S260

The search finds the entries in this table as possible matches:

Name	Location
Smith, James	Redwood Shores, California
Smyth, Edward	Los Angeles, California
Smythe, Joseph	Los Angeles, California

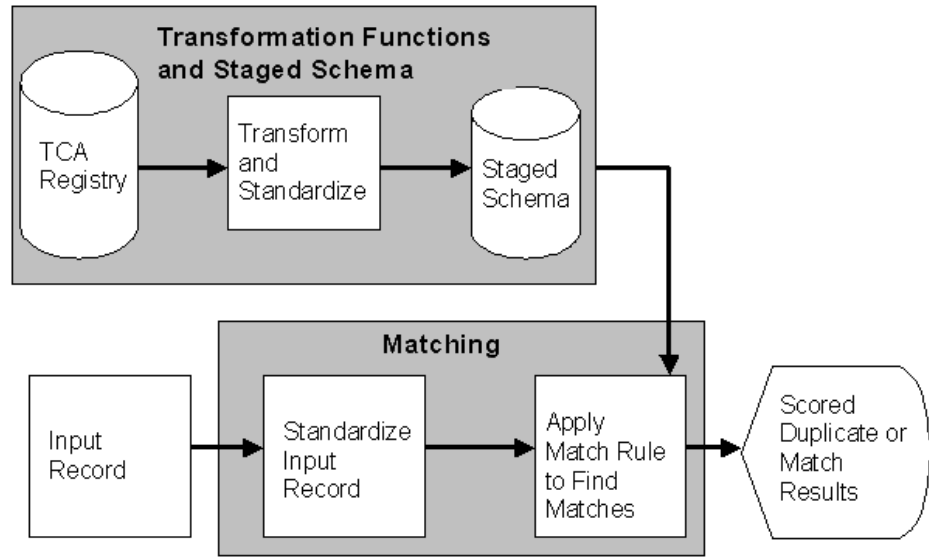
Preventing Duplicate Entries

Other Oracle applications that implement DQM can, for example, enforce the policy that no duplicates are allowed for matches on the combination of cleansed party name and address. The Oracle application can define transformation functions that remove all vowels and double letters and also define a match rule that best expresses the policy.

As a result, you avoid populating the duplicate entries in your TCA registry, shown in this table, because their transformed records are identical:

Original Record	Transformed Record
Allison Byrne, 123 S. Main Str. 11111	ALSN BRN, 123 MAIN, 11111
Alisen Burn, 123 South Main Street, 11111	ALSN BRN, 123 MAIN, 11111

Data Quality Management Process



This diagram illustrates how the different features of Data Quality Management work together to find matches and duplicates.

1. The TCA registry contains party information that could have been entered with typographical errors, spelling errors, and aliases.
2. You run the DQM Staging Program to transform and standardize the attribute values, such as party name and number, in a copy of the registry, the staged schema. You specify which attributes as well as the transformation functions to use on each attribute in the Attributes and Transformation Functions window.
3. The staged schema stores the transformed attribute values, separate from the original registry.
4. Input records come in when you use the Submit Duplicate Identification Batch window to find existing duplicates, as well as when you enter or search for party information in other Oracle applications that implement DQM.
5. The transformation functions in the match rule transform the attribute values in the input record.

6. The same match rule is applied to compare the transformed input record attributes against the attributes in the staged schema.
7. Based on how the attributes match up, potential matches from the staged schema are identified for the input record. If the match rule contains scoring criteria, the selected records from the staged schema are also scored.

Setting Up and Using Data Quality Management

To set up and use Data Quality Management, follow these steps:

1. Define attributes and transformation functions, on page 2-5.
2. Create word replacements (optional), on page 2-16.
3. Generate the staged schema and interMedia indexes, on page 4-4.
4. Define and compile match rules, on page 3-2.
5. Synchronize the staged schema with the TCA registry, on page 4-7.
6. Identify existing duplicate entries in the registry, on page 5-2.

Attributes, Transformation Functions, and Word Replacements

This chapter describes setting up and using attributes, transformation functions, and word replacement pairs used in the matching process.

Attributes Overview

Attributes make up DQM's four logical entities: party, address, contact, and contact point. Each attribute in the logical entities is derived from a specific column in one of the TCA registry tables. These four logical entities make up the staged schema.

For example, the Party logical entity in DQM is named HZ_STAGED_PARTIES and contains attributes from the TCA registry tables:

- HZ_PARTIES
- HZ_ORGANIZATION_PROFILES
- HZ_PERSON_PROFILES

Examples of attributes in the Party logical entity include:

- PARTY_NUMBER from HZ_PARTIES
- DUNS_NUMBER from HZ_ORGANIZATION_PROFILES
- MARITAL_STATUS from HZ_PERSON_PROFILES

Attributes in the staged schema contain values from the TCA registry tables that have been transformed into representations that are more alike for the purposes of matching. Attributes, therefore, are used to search for possible matches between an input record and the TCA registry data.

You cannot extend the staged schema's list of attributes. You can, however, use up to 30 CUSTOM_ATTRIBUTE columns in each logical entity to store your additional attributes.

For a complete list of the seeded attributes in each of the logical entities, see Seeded Attributes on page A-2.

Transformation Functions Overview

Your raw party information can include typographical errors, spelling errors, inconsistent formats, and abbreviations due to input errors. For this reason, searches performed on raw values often are inconclusive or miss potential matches.

Transformation functions transform attribute values in the staged schema so that the values are more similar and useful for the purpose of matching records. Transformation functions neutralize the effects of data errors on your searches. For example, a transformation function that removes all of the double letters in a party name transforms a name such as *Allied Freight* into *Alied Freight*. This transformation makes it easier to match to a party if a typographical error, such as a missing double letter, exists.

After your raw data has been transformed and populated into the staged schema, match rules can use the transformed data to score each record to determine if the record is considered alike enough to be a match to an input value.

DQM includes several standard, or seeded, transformation functions. You can also develop your own transformation functions. For more information on seeded transformation functions, see *Seeded Transformation Functions* on page A-17.

Seeded Transformation Functions

DQM provides several predefined transformation functions, such as *Cleanse* and *Exact*.

For example, the *Cleanse* transformation function performs these actions:

1. Capitalizes all letters.
Zannardi-Montoya is transformed to *ZANNARDI-MONTOYA*.
2. Removes nonalphanumeric characters.
ZANNARDI-MONTOYA is transformed to *ZANNARDIMONTOYA*.
3. Reduces any white space to one space.
4. Removes double letters, retaining only one of the double letters.
ZANNARDIMONTOYA is transformed to *ZANARDIMONTOYA*.
5. Removes vowels except initial vowels.
ZANARDIMONTOYA is transformed to *ZNRDMNTY*.

The final result from the Cleanse transformation function is *ZNRDMNTY*. This result would be populated into the staged schema.

All seeded transformation functions are available in packages. Not all seeded transformation functions are automatically populated into the staged schema. During the installation of DQM, you can decide if you want to use all of the available transformation functions. To ensure that a seeded transformation function is populated into the staged schema, you must enter the name of the transformation function in the Transformations region of the Attributes and Transformation Functions window, and then run the DQM Staging program. For more information, see *Defining Attributes and Transformation Functions* on page 2-5 and *DQM Staging Program* on page 4-4.

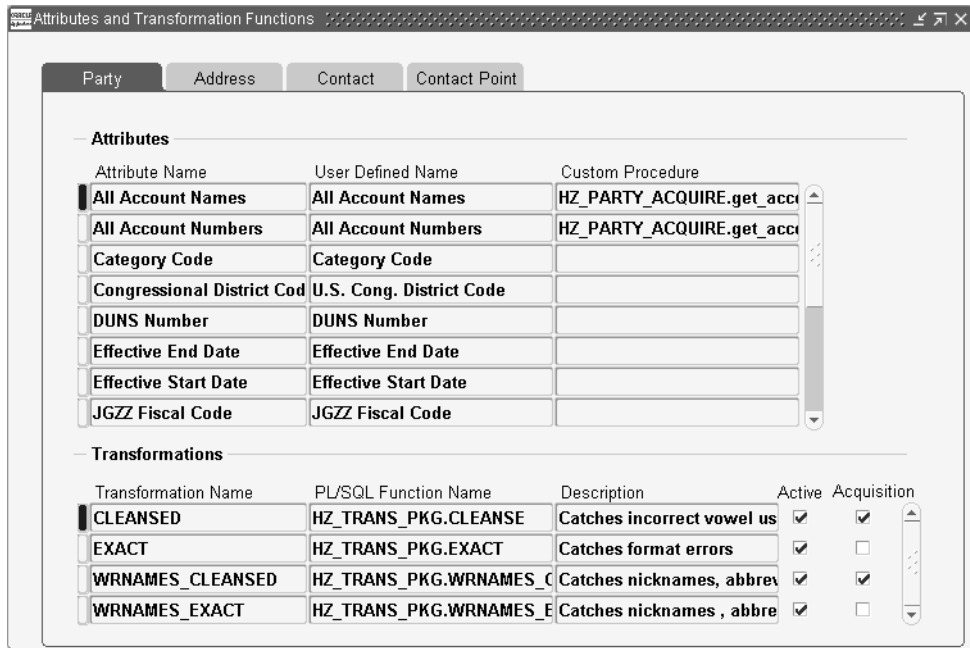
Defining Attributes and Transformation Functions

Use the Attributes and Transformation Functions window to specify the attributes of logical entities that are used for matching and the transformation functions to assign to each attribute. Only attributes that you define in this window can be used in the staged schema for matching.

When you define transformation functions, you must specify which PL/SQL function should be used. You assign transformation functions to the attributes of each type of logical entity in the appropriate tabbed region:

- Party
- Address
- Contact
- Contact Point

Note: Seeded attributes and transformation functions cannot be deleted. You can, however, deactivate transformation functions for an attribute.



To define attributes and transformation functions:

1. Navigate to the Attributes and Transformation Functions window.
2. Select the tabbed region for the logical entity you want to define an attribute or transformation function for.
3. In the Attributes region, select an unused row or create a new record.
4. In the Attribute Name field, select an attribute that you want to transform from the list of values available for this logical entity.
5. In the User Defined Name field, enter your own name for this transformed attribute.
6. If you are using one of the 30 custom attributes, in the Custom Procedure field, enter the name of the PL/SQL procedure for a custom attribute identified in the Attribute Name field. For more information, see [Creating Custom Attributes on page 2-8](#).
Leave the Custom Procedure field blank for seeded attributes.
7. In the Transformations region, select an unused row or create a new record.

8. In the Transformation Name field of the blank row, enter the name of a seeded transformation function or a name for the custom transformation function that you want to assign.
9. In the PL/SQL Function Name field, for custom transformation functions, enter the name of the PL/SQL function that is called to transform the attribute values and to return a formatted result.
10. In the Description field for custom transformation functions, enter your description of the purpose of the transformation function. For seeded transformation functions, the description defaults in this field.

Note: This description appears in the Transformations region of the Match Rules window.

11. Check the Active check box to activate the transformation function for the selected attribute.
12. Check the Acquisition check box if you want to use this transformation function in the acquisition phase of matching.

Note: If too many attributes are used for acquisition, performance can be negatively affected. Uncheck any attributes that you do not require to be used as an acquisition attribute in a match rule. For more information, see *Defining the Acquisition Component of a Match Rule* on page 3-6.

13. Repeat steps 7 to 12 for all transformation functions that you want to assign to the selected attribute.
14. Repeat steps 2 to 13 for all attributes that you want to define.
15. Save your work.

Warning: Before you can use new or updated attributes and transformation functions, you must generate or update your staged schema. For more information see, *Staged Schema and interMedia Index Overview* on page 4-1.

Creating Custom Attributes

Use the Attributes and Transformation Functions window to define custom attributes for the logical entities:

- Party
- Address
- Contact
- Contact point

When a CUSTOM_ATTRIBUTE column needs information about a party, address, contact, or contact point, DQM calls a procedure with this signature:

```
FUNCTION <custom_attribute_proc> (
    p_record_id    IN    NUMBER,
    p_entity_name  IN    VARCHAR2,
    p_attrib_name  IN    VARCHAR2)
RETURN VARCHAR2;
```

The p_record_id value in the signature is the primary key of the table where the CUSTOM_ATTRIBUTE_PROCEDURE is defined.

This table displays the appropriate p_record_id value for each logical entity attribute.

Logical Entity Attribute	p_record_id
Party	PARTY_ID
Address	PARTY_SITE_ID
Contact	ORG_CONTACT_ID
Contact Point	ORG_CONTACT_POINT_ID

Attributes and Transformation Functions

Party Address Contact Contact Point

Attributes

Attribute Name	User Defined Name	Custom Procedure
All Account Names	All Account Names	HZ_PARTY_ACQUIRE.get_acc
All Account Numbers	All Account Numbers	HZ_PARTY_ACQUIRE.get_acc
Category Code	Category Code	
Congressional District Cod	U.S. Cong. District Code	
DUNS Number	DUNS Number	
Effective End Date	Effective End Date	
Effective Start Date	Effective Start Date	
JGZZ Fiscal Code	JGZZ Fiscal Code	

Transformations

Transformation Name	PL/SQL Function Name	Description	Active	Acquisition
EXACT	HZ_TRANS_PKG.RM_SPLCHA	Catches format errors	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
			<input type="checkbox"/>	<input type="checkbox"/>
			<input type="checkbox"/>	<input type="checkbox"/>
			<input type="checkbox"/>	<input type="checkbox"/>

To create a custom attribute:

1. Write the custom procedure that obtains the custom attribute.
2. Navigate to the Attributes and Transformation Functions window.

Note: Because the fields and functionality of the tabbed regions are the same, the following steps are consistent for all of the tabbed regions.

3. In the Attribute Name field, select one of the available CUSTOM ATTRIBUTE values from the list of values. Up to 30 custom attributes are available in each logical entity.
4. In the User Defined Name field, enter a descriptive name for the attribute.
5. In the Custom Procedure field, enter the name of the PL/SQL procedure that DQM uses to obtain the custom attribute.

6. Define transformation functions for the attribute. For more information, see [Defining Attributes and Transformation Functions](#) on page 2-5.

Note: If you do not plan to transform the party information for a custom attribute, then use an Exact transformation function. For more information, see [Transformation Functions Overview](#) on page 2-3.

7. Save your work.

Warning: Before you can use a custom attribute, you must generate or update your staged schema. For more information see, [Staged Schema and interMedia Index Overview](#) on page 4-1.

Creating Custom Transformation Functions

Use the Attributes and Transformation Functions window to create custom transformation functions. First you create a PL/SQL function for transforming a TCA table column and then assign the PL/SQL procedure as a transformation function to an attribute in the Attribute and Transformation Functions window.

With your transformation functions, you can transform any of the TCA registry attributes available in these logical entities:

- Party
- Address
- Contact
- Contact point

To create custom transformation functions:

1. Create your transformation function in the APPS schema as a PL/SQL function that conforms to the following standard signature:

```
FUNCTION <transformation_function_name> (
    p_original_value      IN      <ATTRIBUTE_DATATYPE>
    p_attribute_name      IN      VARCHAR2,
    p_entity_name         IN      VARCHAR2)
RETURN VARCHAR2;
```

The data type of the parameter `p_original_value` depends on the data type of the attribute that the parameter is transforming. The return value, however, must be `VARCHAR2`.

This table shows an example of `p_original_value` data types for different attributes.

Attribute Name	Attribute Code	Data Type of <code>p_original_value</code>
Party Name	PARTY_NAME	VARCHAR2
Person Name	PERSON_NAME	VARCHAR2
Last Update Date	LAST_UPDATE_DATE	DATE
Date of Birth	DATE_OF_BIRTH	DATE

You can use any PL/SQL procedure that adheres to the above signature as a transformation function.

2. Assign your custom PL/SQL function to a transformation function in the Attributes and Transformations window. To use the transformation function, you must also assign the transformation function to at least one attribute. For more information, see *Defining Attributes and Transformation Functions* on page 2-5.

Attribute and Transformation Function Tips

When you create and use transformation functions, consider these issues:

- Design and create as many attributes and transformation functions as possible before populating or updating the staged schema.

After defining any new transformation functions for any attribute, you must run the DQM Staging Program to generate or update the staged schema. Processing the staged schema can require significant computing resources and time. If you are uncertain about initially adding a transformation function or an attribute with the initial group of transformation functions and attributes, the best strategy is to add the transformation function.

- Do not transform lookups. Lookups have a predetermined or known set of values, so you cannot transform them.
- Consider using the seeded Cluster transformation function.

The Cluster transformation function is useful if city names have been abbreviated. For example, Cluster transforms Burlingame into BURL.

- Remember that word replacements are called from transformation functions.
- Consider the possibility that your party information might contain compound names that were entered with the modifier following a comma.

For example, *The New Datsonville* could have been entered into your party information as *Datsonville, The New*. You can create a transformation function that reverses the text before the comma and the text after the comma in any city name.

Word Replacements Overview

Transformation functions can use word replacement lists. With word replacement, you can identify words that act like synonyms of other words for the purpose of searching and matching.

Word replacement is a particularly useful technique for standardizing certain portions of your party information to minimize the effect of data entry inconsistencies on searches. You can use these word replacements in the development of transformation functions. In the PL/SQL functions used in transformation functions, code can be written to call a word replacement list.

Note: For word replacement, a word is defined as a set of alphanumeric characters preceded and followed by white space.

For example, this table displays possible original words for a replacement word in a word list for a specified language.

Word List Name	Language	Original Word	Replacement Word
Person Name	American English	Bob	Robert
Person Name	American English	Rob	Robert
Person Name	American English	Robbie	Robert
Person Name	American English	Roberto	Robert
Person Name	American English	Bobby	Robert

When you enter *Rob* as the value for a first name attribute of a search, the search returns records in which the first name attribute matches to the replacement word for *Rob*. The result of your search would return records containing Bob, Robbie, Roberto, Bobby, and Robert as well as Rob.

After you add or modify word replacement pairs, run the DQM Staging Program to update the staged schema to include the new or revised word replacement pairs. Word replacement pairs become immediately effective for any record that you add to or update in the TCA registry.

Seeded and Custom Word Replacement Pairs

DQM provides word replacement lists containing over 3,000 word replacement pairs in American English. The included word lists are:

- Person Name
- Address
- Organization
- State (US only)
- Domain (used for e-mail and URLs)

See also: *Oracle Trading Community Architecture Data Quality Management Word Replacement Lists*

You can add to the lists of seeded word replacement pairs in American English or create lists in any other languages supported by Oracle Applications.

Creating a Word Replacement List

Use the Word Replacement window to define word replacement lists. Although over 3,000 word replacement pairs in American English are provided, you can modify the provided lists or create lists of word replacement pairs that include words that users in your organization often enter with errors or as shortcuts.

You can specify word pairs consisting of an original word and a replacement word. During the search for matches, the replacement word is used.

Note: You can enter not only whole words, but also abbreviations, word fragments, and numeric characters. For example, you can create a word replacement pair by entering *1* in the Original Word field and *one* in the Replacement Word field. If a user enters *1* to perform a search, then *one* is used to search your party information.

The screenshot shows a window titled "Word Replacements" with a standard Windows-style title bar. Below the title bar, there are three input fields: "Word List" (with a dropdown arrow), "Language" (with a dropdown arrow), and "Source" (with a dropdown arrow). Below these fields is a section titled "Replacements" which contains a table with two columns: "Original Word" and "Replacement Word". The table has 10 empty rows and a vertical scrollbar on the right side.

Original Word	Replacement Word

To define a word replacement list:

1. In the Word List field, enter a name for the list of word replacements.
2. In the Language field, select a language from the list of values.
3. If you want to identify a word replacement list that you created or obtained from a third party, enter information about that source in the Source field.
4. In the Original Word field, enter the original word.

Original words must be unique in this word list. For example, you cannot enter *St.* twice as an original word to be replaced by the replacement words *Street* and *Saint*.

5. In the Replacement Word field, enter the replacement word.

Replacement words do not have to be unique. This is the word that will be used in searches.

You can create several word replacement pairs that have different unique words in the Original Word field and the same word in the Replacement Word field. This table shows an example:

Original Word	Replacement Word
Bob	Robert
Rob	Robert
Robbie	Robert
Roberto	Robert
Bobby	Robert

6. Save your work.
7. Include the word replacement list in your code for a transformation function if it is not already there.
8. Run the DQM Staging Program.

You must run this program to update the staged schema with the transformation function using the updated or new word replacements. Word replacement pairs become immediately effective for any record that you add to or update in the TCA registry.

Word Replacements Tips

When you create and use word replacements, consider these issues:

- Avoid recursive word replacement pairs that can generate inconclusive results.

For example, you need to create word replacements for common variations of a name such as Caroline, Carolyn, Carolan, Carole, and Carol.

This table shows an example of effective word replacement pairings of original and replacement words.

Effective Word Replacement Pairs

Original Word	Replacement Word
Caroline	Carol
Carolyn	Carol
Carolan	Carol
Carole	Carol

This table shows an example of ineffective word pairings using recursive word replacement.

Ineffective Word Replacement Pairs

Original Word	Replacement Word
Caroline	Carolyn
Carolyn	Carolan
Carolan	Carole
Carole	Carol

- Create word replacement pairs consistent with the type of party information in the word lists for person names, addresses, organization names, or states. For example, if you apply a word replacement pair for a person name to an organization name, *Jonathan Deere* might be used as a replacement for the company name, *John Deere*.

- Create a limited number of word replacement pairs between person names and organization names.

Because some organizations are named after people, a limited number of common word replacements of person names to organization names can be appropriate. For example, your organization has regular contact with a group of professional firms such as accountants and lawyers. You can create word replacement pairs specifically for these organizations that would match person names to organization names. Word replacement pairs between different types of party information should be limited to reduce inappropriate matches.

- Create a limited number of word replacement pairs between address-related words and organization names.

Because some organizations are named after geographical locations, a limited number of word replacements of address elements to organization names can be appropriate. For example, your organization has regular contact with the Main Street Cafe. You can create a word replacement pair specifically for this organization that would match an address element to the organization name. Word replacement pairs between different types of party information should be limited to reduce inappropriate matches.

- Consider replacing the original word, or symbol, & with the replacement word *and*.

Several of the seeded transformation functions remove symbols such as &, that are not alphanumeric characters. If some form of conjunction is important to your searches, you should replace & with *and*. You can also avoid or customize the seeded transformation functions that remove &.

- Consider mapping generic address-related words such as street, avenue, and so on to a blank Replacement Word field.

People often know the proper name for an address but might not know the generic portion of the address. For example, you can broaden the search criteria so that both *Market Street* and *Market Avenue* are included in the search results as possible matches. By mapping the words *street* and *avenue* to a blank Word Replacement field, you would broaden the possible matches.

See also: *Oracle Trading Community Architecture Data Quality Management Word Replacement Lists*

3

Match Rules

This chapter describes setting up match rules to identify and score duplicates in the TCA registry.

Match Rules Overview

Match rules determine whether two parties should be identified as a match or potential duplicate. DQM applies match rules when you run the DQM Batch Duplicate Identification Program to find existing duplicates in the TCA registry.

Other Oracle applications that implement DQM can also apply match rules when you:

- Enter or update party information through windows, online forms, import interfaces, or APIs.
- Search for TCA party registry information using search interfaces.

Match rules support score-based matching and consist of two parts, acquisition and scoring. Acquisition provides an initial set of possible matches, and scoring assigns scores to attributes to further filter matches and rank the results.

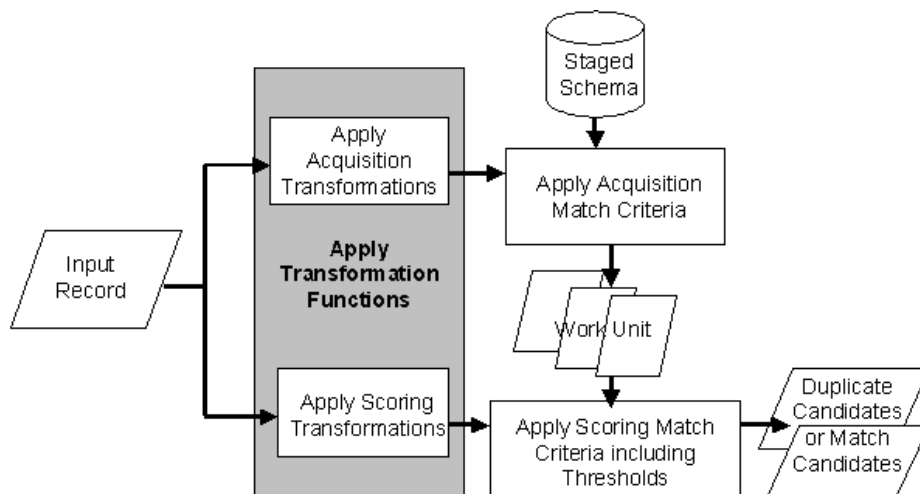
For your business needs, you do not need to use all of the functions of match rules.

- For the simplest type of matching, use only acquisition.
- For ranking your results, use scoring.
- For filtering results based on the scores, use thresholds.

When you define match rules, you can specify which attributes are important for matching and configure how the attributes should be evaluated. For each attribute that you use in a match rule, you also assign one or more transformation functions. For example, you can define a match rule so that only records with the same transformed party name and party number attribute values are considered matches. You can also determine how matching or similar attributes are scored to produce a total score that you can use to evaluate a record.

For a list of seeded match rules, see *Seeded Match Rules* on page A-25. You can copy and modify these seeded match rules to meet your specific requirements.

The Matching Process



This diagram describes the DQM matching process.

1. The staged schema stores the transformed TCA registry attribute values for comparison in the matching process. The included attributes and the transformation functions applied to each are defined in the Attributes and Transformation Functions window.
2. Input records come in when you use the Submit Duplicate Identification Batch window to find existing duplicates, as well as when you enter or search for party information in other Oracle applications that implement DQM.
3. Transformation functions from the acquisition and scoring phases of the match rule transform the input record for the attributes defined in the match rule.
4. The attribute values in the input record that are transformed by the acquisition transformation functions in the match rule are compared against the attribute values in the staged schema.

Matched acquisition attribute values determine the most relevant subset of records from the staged schema to form the work unit.

5. The work unit consists of all records from the staged schema with attribute values that match the transformed acquisition attribute values of the input record.

The work unit saves you time and resources because this relevant subset of records, not the entire staged schema, can be compared against the input record for scoring.

6. The attribute values in the input record that are transformed by the scoring transformation functions in the match rule are compared against the attribute values in the work unit. Based on the match rule, a score is calculated for each record in the work unit.
7. The score of each work unit record is compared against the match, override, and automatic merge thresholds defined in the match rule.

Records with scores above the match threshold are selected as matches for the input record.

For batch duplicate identification, which finds existing duplicates in the TCA registry, if the score also exceeds the automatic merge threshold, the record is automatically marked as a candidate for merge.

In other Oracle applications that implement DQM, the input record can be prevented from entering the registry if a duplicate is found with a score that exceeds the override threshold.

Setting Up the Match Rule

Use the header region of the Match Rule window to enter basic information about the match rule before completing the Acquisition or Scoring tabbed regions.

To set up a match rule:

1. Navigate to the Match Rules window.

The Status field shows whether the match rule is compiled, new, or uncompiled.

- **New** - Match rule is new and needs definition.
 - **Compiled** - PL/SQL code for the match rule has been generated. The match rule is locked against changes, but available for use.
 - **Uncompiled** - Match rule is previously saved, but now unlocked. An uncompiled match rule cannot be used until it is compiled.
2. In the Rule Name field, enter the name that you want to give to your match rule.
 3. In the Description field, enter your description for this match rule.
 4. In the Purpose region, select the Identify Duplicates purpose to designate the match rule for use in batch duplicate identification.

Note: Other Oracle applications that implement DQM can use the Search purpose to identify match rules used to search the staged schema to find records that match search criteria, or the Identify Duplicates purpose for match rules used for duplicate prevention.

Defining the Acquisition Component of a Match Rule

Use the Acquisition tabbed region in the Match Rules window to choose the attributes and transformation functions for comparing an input record to the records in the staged schema to obtain an initial set of matched records.

Select attributes that help you obtain all relevant matches. For example, if you want to find records that have matching names, include the Party Name attribute in the match rule. You can only use attributes that are assigned in the Attributes and Transformation Functions window. For more information, see *Defining Attributes and Transformation Functions* on page 2-5.

You must select at least one transformation function for each attribute. DQM applies the selected transformation functions to that attribute before the input record is compared to the records in the staged schema. You can choose more than one transformation function for each of the attributes in the match rule.

If you specify more than one acquisition attribute in the match rule, in the Attribute Match region you can choose whether all or any attribute must match to determine if that input record is a match to a record in the staged schema.

You can use an attribute as a filter so that the attribute always appears as a match in the returned records. The search engine groups attributes of a logical entity, under a match-all condition, when the attributes are used as filters. For example, in the table below, the Address1, Country, and Identifying Address Flag attributes are grouped together because they belong to the address logical entity. This grouping overrides the match requirement selected in the Attributes Match region.

Note: If the Type of an attribute is Lookup, then the DQM search engine defaults that attribute as a filter because the values of the attribute are usually not unique across records.

For example, this table displays a set of possible attributes:

Attribute Name	Logical Entity	Type	Default as Filter
Party Name	Party		No
Address1	Address		No
Country	Address	Lookup	Yes
Identifying Address Flag	Address	Lookup	Yes

If you select Match Any Attributes in the Attribute Match region, all of the returned records will match either of the following criteria:

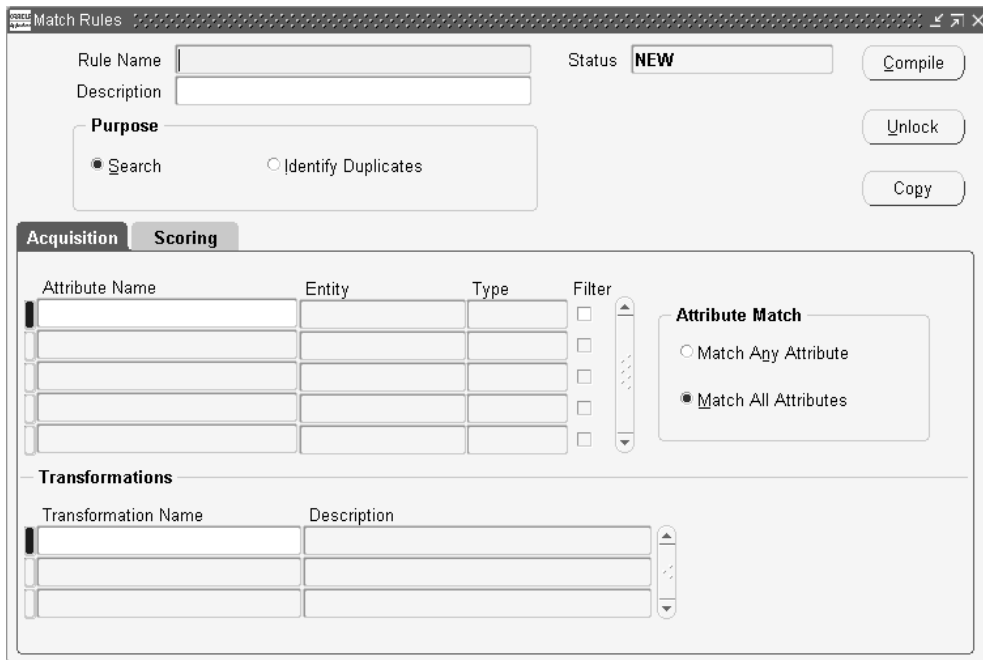
- Party Name
- Address1 and Country and Identifying Address

The initial set of matched records obtained by acquisition is called the work unit. A work unit includes all records with acquisition attribute values, transformed using the transformation functions, that match attribute values of the input record. The work unit can be the final set of matches or used for scoring.

If you only want to consider this initial set of matched records, you do not have to specify thresholds or scores in the Scoring tabbed region. You only have to complete fields in the Acquisitions tabbed region.

Prerequisites

Before you can define the acquisition component of a match rule, you must complete the header region of the Match Rules window. For more information, see [Setting Up the Match Rule on page 3-5](#).



To define the acquisition component of a match rule:

1. Navigate to the Match Rules window and query for the uncompiled match rule that you set up.
2. In the Acquisition tabbed region, select an attribute from the list of values in the Attribute Name field for matching consideration. The list of values contains the attributes in the staged schema.

The Entity and Type fields are automatically populated with the values you assigned when you defined the attribute. You cannot enter values into these fields.

3. Check the Filter check box for each attribute that you want to use as a filter.

Note: Attributes used as filters should not be used for the Scoring component of the match rule because all records contain them.

4. In the Attribute Match region, select either the Match Any Attribute or Match All Attributes option.

5. In the Transformation Name field, select a transformation function from the list of values for the attribute. You can choose more than one transformation function for an attribute.

The Description field is populated with the transformation function description that you entered in the Attributes and Transformation window. For more information, see *Defining Attributes and Transformation Functions* on page 2-5.

Note: While you are defining a match rule, you can save it before proceeding to the next step.

Warning: Before proceeding to the next step you must make sure the staged schema has been generated. For more information see *Staged Schema and interMedia Index Overview* on page 4-1.

6. Press the Compile button to compile, lock, and save the match rule.

Note: The compile function includes a locking feature to ensure that the match rule and its compiled PL/SQL code are always synchronized. Before you can edit a compiled match rule, you must unlock it. For more information see, *Editing Match Rules* on page 3-24.

At this point, you have completed a match rule that requires that a record matches any or all attributes you have selected. You can define the scoring component of a match rule so that records that do not exactly match the values of the acquisition attributes values are considered a match. For more information, see *Defining the Scoring Component of a Match Rule* on page 3-10.

Defining the Scoring Component of a Match Rule

Use the Scoring tabbed region to select the attributes, scores, and weights used to calculate the match score for each record in the work unit. Scoring and thresholds provide more flexible and granular matching. DQM calculates a weighted score and then compares that score against the match rule thresholds to determine if the record in the work unit:

- Matches the input record.

This is the minimum score, or match threshold, that a record in the work unit must meet to be considered a match to the input record.

- Cannot be overridden.

Other Oracle applications that implement DQM can prevent a record with a score equal to or above the override threshold from entering the registry, because it is considered to be a match or duplicate of an existing record.

- Should be merged.

A record with a score equal to or above the threshold is marked by default as a candidate for party merge without manual intervention. This field affects only the batch duplicate identification process. For more information, see *Batch Duplicate Identification Overview* on page 5-2.

In the Scoring tabbed region, you can select the attributes that you want to use to score the records in the work unit. These attributes do not have to be that same as those in the acquisition tabbed region and you can use more attributes for scoring. You must, however, use attributes that are assigned in the Attributes and Transformation Functions window. For more information, see *Defining Attributes and Transformation Functions* on page 2-5.

You rank your attributes in order of importance by assigning scores. Assign the highest score to the attribute that you consider the most important for a match. The score is awarded to a record if the attribute is evaluated as a match. If the attribute does not match, then a score of zero is assigned.

You must select at least one transformation function for each attribute in the Scoring tabbed region. DQM applies the selected transformation functions to that attribute before the input record is compared to the record in the work unit. You can choose more than one transformation function for each of the attributes in the match rule.

You can assign percentage weights to the transformation functions depending on how similar the transformed value of the attribute would be to the original values of the attribute. For example, you should assign more weight to the Exact transformation function than to the Cleansed transformation function because Exact makes fewer changes to the original data.

If a transformed attribute value is a match, the weight is multiplied by the score to determine the weighted score of the attribute. If an attribute is assigned more than one transformation function, the highest weighted score is awarded to the record for the attribute.

The Similarity matching option lets you create fuzzier matches by applying the similarity algorithm to transformed attribute values. The similarity algorithm compensates for unanticipated errors that the transformation functions do not catch. The algorithm compares the transformed attribute value of the input record to the corresponding attribute value from the work unit record and assigns a percentage for the extent of similarity. If that percentage is greater than or equal to the similarity percentage that you enter in the Scoring tabbed region, the attribute is considered a match.

The match score for the record is the sum of the actual weighted attribute scores. This match score is the value that is compared to the match rule thresholds.

Prerequisites

Before you can define the scoring component of a match rule, you must complete the match rule setup information and complete the Acquisition tabbed region. For more information, see *Defining the Acquisition Component of a Match Rule* on page 3-6.

The screenshot shows the 'Match Rules' window with the 'Scoring' tab selected. At the top, there are fields for 'Rule Name', 'Description', and 'Status' (set to 'NEW'). There are buttons for 'Compile', 'Unlock', and 'Copy'. Below this is the 'Purpose' section with radio buttons for 'Search' (selected) and 'Identify Duplicates'. The 'Acquisition' and 'Scoring' tabs are visible. The 'Scoring' section includes 'Match Threshold' (0), 'Override Threshold', and 'Automatic Merge Threshold'. A table with columns 'Attribute Name', 'Entity', 'Type', and 'Score' is present. Below the table is the 'Transformations' section with columns 'Transformation Name', 'Description', 'Weight (%)', 'Similarity Exact', and 'Similarity (%)'. There are three rows in the transformations table, each with radio buttons for 'Exact' and 'Similarity'.

To define the scoring component of a match rule:

1. Navigate to the Scoring tabbed region of the Match Rules window.
2. In the Attribute Name field, select an attribute from the list of values for matching consideration. The list of values contains seeded attributes in the staged schema and custom attributes.

The Entity and Type fields are automatically populated with the values you assigned when you defined the attribute. You cannot enter values into these fields.

3. In the Score field, assign a score to the attribute expressed in integer form.
4. In the Transformation Name field, select a transformation function from the list of values for that attribute. You can specify multiple transformation functions for an attribute.

Use the fewest transformation functions possible in your match rule. Using more transformation functions than necessary could affect the time required for staging and the performance of your search.

The Description field is populated with the transformation function description that you entered in the Attributes and Transformation window. For more information, see Defining Attributes and Transformation Functions on page 2-5.

5. In the Weight (%) field, enter the percentage of the attribute score that is factored into the weighted total score in the event of a match.
6. Select the Exact or Similar option to indicate if a transformation function requires an exact match or applies the similarity algorithm. If you select the Similarity option, you must enter a percentage in the Similarity (%) field.

Note: By default the Exact option is selected because the Similarity option requires additional computing resources and time.

7. In the Match Threshold field, enter the minimum match score.

Determine the minimum set of attributes required for a match. The total of the attribute scores of this minimum set is the maximum value of the match threshold.

Note: Do not enter a value in the Override Threshold field. This field is used only by other Oracle applications that implement DQM to prevent duplicates.

8. In the Automatic Merge Threshold field, enter the automatic merge threshold.

Determine the minimum set of attributes required for considering two parties for merge. The total of the attribute scores of this minimum set is the maximum value for the automatic merge threshold.

The automatic merge threshold must be greater or equal to the match threshold.

Note: If you specify thresholds in your match rules, be sure that:

- You do not set any thresholds too low. Low thresholds might let combinations of attributes pass as matches that are not significant or relevant to your search.
 - The threshold scores are less than the sum of the possible scores of all attributes.
-
-

9. Press the Compile button to compile, lock, and save the match rule.

DQM Compile All Rules Program

Use the DQM Compile All Rules Program when you initially install DQM to compile the seeded match rules provided by TCA.

You also use this program to compile all of the match rules that you create in the Match Rules window.

If several match rules are complete, but not compiled, you can use the DQM Compile All Rules Program to compile all of the match rules at once instead of querying up and compiling each match rule from the Match Rules window.

The program attempts to compile all match rules even if certain match rules fail to compile. Match rules that are already compiled remain compiled, and match rules that can be successfully compiled are compiled. Match rules that fail to compile are left uncompiled.

You can see the compile status of each match rule in the Status field of the Match Rules window.

Use the Standard Request Submission windows to submit the DQM Compile All Rules Program.

See also: Using Standard Request Submission, *Oracle Applications User Guide*

Prerequisites

Before you can compile your match rules, you must run the DQM Staging Program. For more information, see DQM Staging Program on page 4-4.

Match Rule Example

This section provides an example of how you can develop a match rule. This example focuses on the scoring and threshold components, detailing the thought process you might take to create an effective match rule.

Preparing for the Match Rule

1. Create a list of all of the attributes that should match between two matching records. This list should include attributes that are really important as well as attributes that are just good to have as matches.

For this example, this table shows the following list of attributes:

Attribute Name	Entity	Type
Party Name	Party	
Phone Number	Contact Point	
Address1	Address	
Country	Address	Lookup
Postal Code	Address	
Contact Last Name	Address	

2. Rank the order of importance of the attributes, as shown in this table:

Rank	Attribute Name	Entity	Type
1	Party Name	Party	
2	Phone Number	Contact Point	
3	Contact Last Name	Contact	
4	Address1	Address	
5	Country	Address	Lookup
6	Postal Code	Address	

This ranking indicates that the attribute score you assign to party name is the highest and the scores are lower or stay the same as you go down the ranking.

3. Identify the minimum sets of attributes you require to match for records to be considered matches, for example:

- Only the party name
- Only the phone number

This requirement indicates that your acquisition attributes should at least include party name and phone number and that you should select the Match Any option in the Attribute Match region.

4. Identify the sets of attributes that by themselves are not good enough to indicate that you have matching records, but which, if they were close enough matches, might give additional credence to a match on the minimum set of party name and phone number.

- Only address
- Only country
- Only postal code
- Only address and country
- Only address and postal code
- Only country and postal code
- Only contact last name

This selection determines the attributes that you need to include as scoring attributes.

Defining the Acquisition and Scoring Components

1. Choose attributes from step 1 of Preparing for the Match Rule that would get you all of the possible matches. You must include the attributes from step 3 of Preparing for the Match Rule. For this example, you select:
 - Party Name
 - Phone Number
 - Contact Last Name
2. Choose the Attribute Match criteria. Because you have two sets in step 3 of Preparing for the Match Rule, you should select Match Any.

3. Select attributes from step 1 of Preparing for the Match Rule that you want to use to score the records. You must include the attributes from step 4 of Preparing for the Match Rule.

This table shows the scoring attributes.

Attribute Name	Entity	Type
Party Name	Party	
Phone Number	Contact Point	
Address1	Address	
Country	Address	Lookup
Postal Code	Address	
Contact Last Name	Contact	

4. Assign scores to the scoring attributes following the ranking in step 2 of Preparing for the Match Rule. The most important attributes receive the highest scores. For this example, the score assignments should reflect the following:
 - Matches on party name provide the best match results, so you assign the highest score to party name.
 - Matches on a phone number might be the second best matching criterion, so you assign the next highest score to phone number.
 - Combinations of the address components and contact last name are the third best, so you assign scores by relative importance.
 - The contact last name attribute is estimated to have about the same value as the address1 attribute.

For this example, the scores in this table are assigned to the scoring attributes.

Scoring Attributes	Scores
Party Name	40
Phone Number	30
Address1	15
Country	10
Postal Code	10
Contact Last Name	15

The total score for the attributes in this table is 120.

Setting the Match Threshold

1. Obtain minimum sets from step 3 of Preparing for the Match Rule and total attribute scores from step 4 of Defining the Acquisition and Scoring Components.
 - For Party Name the total attribute score is 40.
 - For Phone Number the total attribute score is 30.
2. Set your match threshold based on the lower score of the two minimum sets, in this example, 30.

With the match threshold at 30, you can interpret scoring as follows:

- If only the phone number is a match, the record is a match because the score equals the match threshold of 30.
- If only the party name is a match, then the record is a match because the score exceeds the match threshold of 30.
- If the country, postal code, and contact last name are a match, then the record is a match because the attributes' combined score of 35 exceeds the match threshold of 30.
- If the address1, country, and postal code are a match, then the record is a match because the attributes' combined score is 35, which exceeds the match threshold of 30.

With the match threshold at 30, this table shows results of possible matches:

Possible Matches	Cumulative Score	Match
Party Name	40	Yes
Phone Number	30	Yes
County, Postal Code, and Contact Last Name	35	Yes
Address1, Country, and Postal Code	35	Yes
Party Name and Phone Number	70	Yes
Phone Number and Country	40	Yes
Address1 and Country	25	No
Country and Postal Code	20	No
Party Name, Address1, and Contact Last Name	70	Yes

Considering the Impact of Transformation Functions on Your Thresholds

If you have transformation function weights other than 100%, then you might need to tune your threshold. With weights other than 100%, the total score for the record can be lower than the match threshold that you assigned. The total score is the sum of attribute scores that are multiplied by the weight.

For example, a minimum set of attributes required for match consists of party name. The following table shows the transformation functions and weights assigned to the Party Name attribute, as well as the weighted attribute scores calculated for each transformation function.

Party Name Attribute with Attribute Score 40

Transformation Function	Weight %	Weighted Attribute Score Calculation
Exact	100	$100\% * 40 = 40$
Reverse	80	$80\% * 40 = 32$
Cleanse	50	$50\% * 40 = 20$

Depending on the transformation functions, a matching party name can have a weighted attribute score below 40. With a weighted score of 20, for example, this minimum set might not exceed the match threshold of 30. If you want all possible matches that originate from any of the transformation functions, you might want to adjust some of your values.

You have three options:

1. Decrease the match threshold to the lowest possible weighted attribute score. Performing this option might affect the scores of other attributes and thresholds.
2. Increase the weight of the transformations so that the lowest possible weighted attribute score exceeds the match threshold. This option might not always be possible because weights must be less than or equal to 100.
3. Increase the attribute score so that the lowest possible weighted attribute score exceeds the match threshold.

For example, you can increase the Party Name attribute score to 60 and the Cleanse transformation function weight to 70%. This table shows the adjusted assignments with each possible weighted attribute score exceeding the match threshold of 30.

Party Name Attribute with Attribute Score 60

Transformation Function	Weight %	Weighted Attribute Score
Exact	100	60
Reverse	80	48
Cleanse	70	42

Copying Match Rules

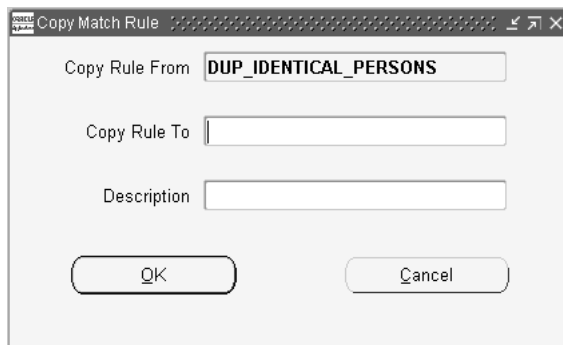
Use the Match Rules window to copy and modify an existing, compiled match rule. You can only copy a compiled match rule. A newly created match rule has the New status and must be compiled before you can use or copy it. You can use the copy functionality to create and test variations of an existing match rule before you improve or modify the original rule.

The screenshot shows the 'Match Rules' window with the 'Scoring' tab selected. At the top, there are input fields for 'Rule Name' and 'Description', and a 'Status' dropdown set to 'NEW'. To the right are buttons for 'Compile', 'Unlock', and 'Copy'. Below these is a 'Purpose' section with two radio buttons: 'Search' (selected) and 'Identify Duplicates'. The main area is divided into 'Acquisition' and 'Scoring' tabs, with 'Scoring' active. It contains a table with columns 'Attribute Name', 'Entity', 'Type', and 'Filter'. To the right of this table is an 'Attribute Match' section with two radio buttons: 'Match Any Attribute' and 'Match All Attributes' (selected). Below the table is a 'Transformations' section with a table with columns 'Transformation Name' and 'Description'.

To copy an existing match rule:

1. Navigate to the Match Rules window.
2. In the Rule Name field, query for the existing compiled match rule that you want to copy.
3. Press the Copy button.

The Copy Match Rule window appears.



4. In the Copy Rule To field of the Copy Match Rule window, enter the name that you want to assign to the new match rule.
5. In the Description field, enter a description for your match rule.
6. Press the OK button.

The Match Rules window is populated with your match rule name and the characteristics of the source match rule.

7. Modify your new match rule as needed in the Match Rules window.

Press the Compile button after you finish entering your edits to generate the PL/SQL code, lock the match rule, and save the match rule.

For more information about editing an existing match rule, see [Editing Match Rules](#) on page 3-24.

Editing Match Rules

Use the Match Rules window to unlock and then edit a compiled match rule. A compiled match rule must be unlocked before you can edit it. Unlocking the match rule disables the PL/SQL code used by the match rule. The match rule cannot be used while it is unlocked. After you finish editing the match rule, you must compile it before it can be used.

The screenshot shows the 'Match Rules' window with the following sections:

- Rule Name:** [Text Field]
- Description:** [Text Field]
- Status:** NEW
- Purpose:**
 - Search
 - Identify Duplicates
- Buttons:** Compile, Unlock, Copy
- Acquisition / Scoring:**
 - Match Threshold: 0
 - Override Threshold: [Text Field]
 - Automatic Merge Threshold: [Text Field]
 - Table with columns: Attribute Name, Entity, Type, Score
- Transformations:**
 - Table with columns: Transformation Name, Description, Weight (%), Exact, Similarity (%)

To edit an existing match rule:

1. Navigate to the Match Rules window.
2. In the Rule Name field, query for the match rule that you want to edit.
3. Press the Unlock button, if the match rule is compiled. If the match rule is uncompiled, go to step 4.

The Status changes to Uncompiled.

Unlocking a match rule disables its PL/SQL code and prevents the match rule from being used.

4. Enter your changes to the match rule in the Match Rules window.
5. Press the Compile button after you finish entering your edits to generate the PL/SQL code, lock the match rule, and save the match rule.

Staged Schema and *interMedia* Index

This chapter describes the staged schema and *interMedia* indexes, including creating and updating both.

Staged Schema and *interMedia* Index Overview

The staged schema is a separate set of database tables with a portion of the data from the TCA registry that transformation functions have standardized for improved matching. During matching, the input record is compared against the records in the staged schema instead of the original registry. To enable efficient searches, the staged schema is indexed using Oracle *interMedia*. DQM searches against the *interMedia* indexes to take advantage of *interMedia*'s high performance capabilities in searching through large quantities of data.

Staged Schema Features

Each table in the staged schema contains attribute values for each logical entity: party, address, contact, and contact point, and can include up to 30 custom attributes. In the Attributes and Transformation Functions window, you define the attributes that you want in the staged schema as well as the transformation functions to apply to each attribute value. For example, for the party logical entity, you can include the party name attribute and specify two transformation functions to transform the party names. Each attribute, in essence, corresponds to a column in a TCA registry table that stores the attribute value, and the staged schema contains the transformed version of those values.

This table shows the mapping of the TCA registry to the staged schema:

Logical Entity	TCA Registry Table	Staged Schema Table
Party	HZ_PARTIES	HZ_STAGED_PARTIES
Address	HZ_PARTY_SITES	HZ_STAGED_PARTY_SITES
Contact	HZ_CONTACTS	HZ_STAGED_CONTACTS
Contact point	HZ_CONTACT_POINTS	HZ_STAGED_CONTACT_POINTS

For more information about defining attributes and transformation functions for the staged schema, see *Defining Attributes and Transformation Functions* on page 2-5. For a complete list of the seeded attributes for each logical entity, see *Seeded Attributes* on page A-2.

Data Quality Management provides programs that you run to:

- Create and update the staged schema. See DQM Staging Program on page 4-4.
- Synchronize the staged schema with the TCA registry. See DQM Synchronization Program on page 4-7.

***interMedia* Index Features**

interMedia indexes speed up the acquisition phase of the matching process by quickly limiting the number of parties to evaluate for scoring. *interMedia* indexes are built using inverted indexes, a structure that differs from normal indexes. DQM provides four *interMedia* indexes, one for each logical entity.

DQM provides programs that you run to:

- Create *interMedia* indexes. See DQM Staging Program on page 4-4.
- Optimize *interMedia* indexes. See DQM Index Optimization Program on page 4-6.
- Update *interMedia* indexes. See DQM Synchronization Program on page 4-7.

DQM Staging Program

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.

When you specify in the Staging Command parameter to create indexes, the DQM Index Creation Program automatically runs to generate *interMedia* indexes for the staged schema. Invoke the DQM Index Creation Program not just to create indexes for the first time, but also to reindex the entire staged schema whenever necessary. You can use the HZ: DQM Index Creation Memory profile option to override the default amount of memory that is allocated for creating *interMedia* indexes. For more information, see Profile Options on page C-2.

To decrease the amount of time it takes to create the staged schema, you can specify how many parallel workers to use when you run the DQM Staging Program. Workers are processes that run at the same time to complete a task that would otherwise take longer with a single process. The DQM Staging Program invokes the DQM Stage Worker, which spins off parallel workers for staging the data.

Aside from creating the entire staged schema and the *interMedia* indexes, you can also run the DQM Staging Program to:

- Stage only new or updated parties that are not currently in the staged schema
- Stage data using only new transformation functions
- Generate only mapping procedures, which map attributes to the source columns of the attribute values

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

Use the Standard Request Submission windows to submit the DQM Staging Program.

See also: Using Standard Request Submission, *Oracle Applications User Guide*

Prerequisites

Before you submit the DQM Staging Program, you must define attributes and transformation functions. For more information, see *Defining Attributes and Transformation Functions* on page 2-5.

Program Parameters

Number of Parallel Staging Workers

Enter the number of parallel staging workers, between one and ten, that you want to use to create the staged schema, or leave the parameter blank. The DQM Stage Worker runs with one worker if you leave the parameter blank.

Staging Command

Enter what you want the program to accomplish. This table shows the options.

Staging Command	Description
CREATE_INDEXES	Create Indexes - To invoke the DQM Index Creation Program to generate <i>interMedia</i> indexes for the staged schema.
GENERATE_MAP_PROC	Generate mapping procedures - To generate only mapping procedures.
STAGE_ALL_DATA	Stage all data, create indexes, and generate map procedures - To stage all data, create <i>interMedia</i> indexes, and generate mapping procedures.
STAGE_MISSING_PARTIES	Stage new/updated parties - To stage all parties that are not in the staged schema. You can also instead run the DQM Synchronization Program.
STAGE_NEW_TRANSFORMATIONS	Stage new transformations - To stage data using only new or modified transformation functions.

Continue previous execution

Enter *Yes* to continue running a previous DQM Staging Program that did not finish. Enter *No* to run a new DQM Staging Program.

DQM Index Optimization Program

Use the DQM Index Optimization Program to optimize the *interMedia* indexes. In the Standard Request Submission windows, you can schedule this program to run on a periodic basis. In general, you can run the DQM Index Optimization Program less frequently than the DQM Synchronization Program. The DQM Index Optimization Program improves the performance of *interMedia* searches, which gradually degrades over time. You should run this program once in a while, but frequent optimization is not mandatory.

The DQM Synchronization Program, however, synchronizes data between the TCA registry and the staged schema, so you should run this program as often as possible to keep the search results accurate. For more information, see DQM Synchronization Program on page 4-7.

Use the Standard Request Submission windows to submit the DQM Index Optimization Program.

See also: Using Standard Request Submission, *Oracle Applications User Guide*

Prerequisites

Before you submit the DQM Index Optimization Program, you must make sure that both the staged schema and *interMedia* indexes exist.

DQM Synchronization Program

Use the DQM Synchronization Program to synchronize an existing staged schema and *interMedia* indexes with the TCA registry. This program ensures that any new or updated data in the TCA registry is reflected in the staged schema and *interMedia* indexes. The program updates the *interMedia* indexes after updating the staged schema.

In the Standard Request Submission windows, you can schedule the DQM Synchronization Program to be run on a periodic basis. The frequency that you want to submit this program with depends on:

- How often the information in the TCA registry usually changes
- How often you search against the TCA registry
- How much processing power you have

If you run the program frequently, the data in the staged schema and *interMedia* indexes are sooner updated for your searches. The *interMedia* indexes, however become less optimal and slower to use. To optimize the indexes, see DQM Index Optimization Program on page 4-6.

Use the Standard Request Submission windows to submit the DQM Synchronization Program.

See also: Using Standard Request Submission, *Oracle Applications User Guide*

Prerequisites

Before you submit the DQM Synchronization Program, you must make sure that both the staged schema and *interMedia* indexes exist.

Batch Duplicate Identification

This chapter describes setting up and using batch duplicate identification to identify duplicates that exist within the TCA registry.

Batch Duplicate Identification Overview

After you define transformation functions and match rules and create the staged schema, you can use batch duplicate identification to identify duplicates that currently exist in the TCA registry and select the parties that you want to merge. The batch duplicate identification process compares entries within the staged schema and determines potential duplicates based on your match rule.

You can select a subset of the parties in the HZ_STAGED_PARTIES table in the staged schema that you want to run through the batch duplicate identification process. You can compare these records only against one another or against all records in the table. The process compares all records for each party. For example, each contact point for a party is compared against all contact points.

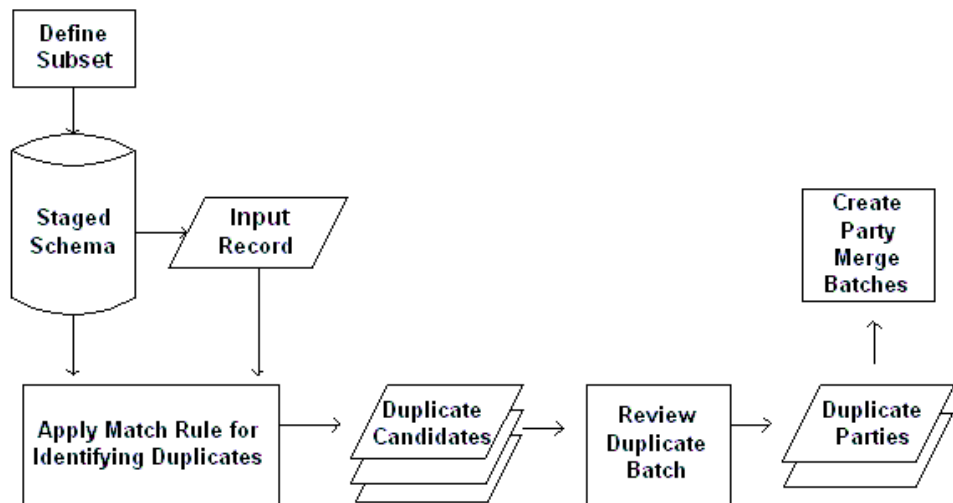
Batch Duplicate Identification Process

The batch duplicate identification process involves creating a duplicate identification batch that can consist of a subset of parties, creating merge batches from the duplicate identification batch, and submitting the merge batches to Party Merge.

Data Quality Management provides windows that you can use to:

- Define and submit duplicate identification batches, on page 5-5
- Review duplicate candidates and create merge batches, on page 5-8
- Submit merge batches, on page 5-13

This diagram illustrates the batch duplicate identification process in more detail:



1. In the Submit Duplicate Identification Batch window, define a duplicate identification batch, which can consist of a subset of parties. Specify whether to compare the parties against one another or against all parties.
2. From the same window, run the DQM Batch Duplicate Identification Program, which searches for duplicates of the batch that you defined. The batch provides the input records that the staged schema matches against.
3. The DQM Batch Duplicate Identification Program applies the match rule in the Submit Duplicate Identification Batch window to identify duplicates.
4. The Duplicate Identification: Batch Review window displays the duplicate candidates for your batch.
5. Review the results for your duplicate identification batch and use the Match Details window to see more information about the matches.
6. In the Duplicate Identification: Batch Review window, specify the duplicate parties that you want to merge from and to and indicate which parties you do not want to be identified as duplicates in the future.
7. In the same window, create a merge batch with the parties that you want to merge.

Use the Review Party Merge Batches window to submit merge batches to Party Merge for the actual merge process.

Prerequisites

Before you can use batch duplicate identification to identify existing duplicates in your TCA registry, you:

- Must define attributes and transformation functions. For more information, see [Defining Attributes and Transformation Functions](#) on page 2-5.
- Must create the staged schema. For more information, see [DQM Staging Program](#) on page 4-4.
- Must define match rules with Identify Duplicates selected in the Purpose region of the Match Rules window. For more information, see [Match Rules Overview](#) on page 3-2.
- Can optionally define the DQM Match Rule ID for Batch Duplicate Identification profile option if you want to default a match rule for the process. This match rule defaults in the Submit Duplicate Identification Batch window. You can always override this default and enter another match rule for the DQM Batch Duplicate Identification Program. For more information, see [Profile Options](#) on page C-2.

Defining Duplicate Identification Batches

Use the Submit Duplicate Identification Batch window to define and submit the batch of subset entries that you want to find duplicates for. When you submit the batch, the DQM Batch Duplicate Identification Program automatically applies the match rule from this window and scores potential duplicates.

If you do not define a subset, the DQM Batch Duplicate Identification Program compares all records in the staged schema against one another. This process can take a long time, depending on the detail of your match rule and the size of your staged schema.

Define a subset of records to compare against the rest of the staged schema or against one another, for two reasons:

- To save time
- If you are familiar with the contents of the TCA registry, for example, if you know about a new influx of data in a specific date range or records that were created by a particular application or individual

You can select up to ten conditions to define the subset, using any of the attributes from the HZ_PARTIES table. You can also manually enter SQL statements to define the subset.

After the DQM Batch Duplicate Identification Program finishes, the results are displayed in the Duplicate Identification: Batch Review window.

Batch Name

Match Rule

Number of Workers

Match within Subset

Find Merged Parties

Define Subset

Attribute		Value
<input type="text"/>	=	<input type="text"/>
<input type="text"/>	=	<input type="text"/>
<input type="text"/>	=	<input type="text"/>
<input type="text"/>	=	<input type="text"/>
<input type="text"/>	=	<input type="text"/>
<input type="text"/>	=	<input type="text"/>
<input type="text"/>	=	<input type="text"/>
<input type="text"/>	=	<input type="text"/>
<input type="text"/>	=	<input type="text"/>

SQL Clause

Submit Batch

To define and submit a duplicate identification batch:

1. Navigate to the Submit Duplicate Identification Batch window.
2. Enter a name for the duplicate identification batch in the Batch Name field.
3. Select a match rule from the list of values to use for identifying and scoring duplicates in the Match Rule field.

The match rule defaults from the DQM Match Rule ID for Batch Duplicate Identification profile option, if defined.

4. In the Number of Workers field, enter the number of parallel workers that you want to use to improve performance.

Workers are processes that run at the same time to complete a task that would otherwise take longer with a single process. The default number of workers is 1.

5. Uncheck the Match within Subset check box if you want to compare the subset against the entire staged schema for duplicates.

By default, the records in the subset are only compared against one another.

6. Check the Find Merged Parties check box if you want to include parties that were previously merged in the search.
7. Navigate to the Define Subset region.
8. In the Attribute fields, select attributes from the list of values that you want to define the subset with.
9. For each attribute, select a condition:
 - >
 - <
 - =
 - CONTAINS

Note: The CONTAINS condition applies only to a word surrounded by white space.

10. In the Value fields, enter a value for each attribute and condition.

For example, if you enter *1001* for the Party Number attribute with a less than condition, the subset includes only parties with a number of 1000 or lower.
11. In the SQL Clause text box, you can manually add to the corresponding SQL clauses that are automatically generated when you define subset conditions. Alternatively, you can enter SQL statements instead of selecting attributes and conditions in the previous fields.
12. Press the Submit Batch button.

The DQM Batch Duplicate Identification Program runs to identify duplicates for the subset of records that you defined, using the match rule that you specified.

Reviewing Duplicates and Creating Merge Batches

Use the Duplicate Identification: Batch Review window to review the potential duplicates that the DQM Batch Duplicate Identification Program found and create a merge batch that consists of parties that you want to merge. For the subset that you defined for the duplicate identification batch, the Duplicate Identification: Batch Review window displays parties with matches as merge-to parties and their potential duplicates as merge-from parties. A merge-from party would be merged into the merge-to party during the party merge process.

You can use the Match Details window to see the reasons why the DQM Batch Duplicate Identification Program selected any pair of merge-to and merge-from parties as duplicate candidates. This information helps you determine whether the parties are in fact duplicates or not.

After you evaluate a pair of duplicate candidates, you can:

- Switch the merge-from party to the merge-to party and the merge-to party to the merge-from party.
- Mark the pair as not duplicates so that the parties are not selected as candidates in the future. You can define a date range for this definition to be active.
- Determine whether to merge the pair or not. Any merge-from party with a score that exceeds the automatic merge threshold defined in the match rule is defaulted to be merged into the merge-to party. You can override this default.

When you finish evaluating a batch, you can create a merge batch with all the duplicate pairs of parties that you select for merge.

Note: After you become familiar with DQM, you might choose to trust your match rules and submit a merge batch with the results of the DQM Batch Duplicate Identification Program without evaluating each duplicate candidate.

Prerequisites

Before you can review potential duplicates and define merge batches, you must use the Submit Duplicate Identification Batch window to define a duplicate identification batch and run the DQM Batch Duplicate Identification Program on that batch. For more information, see *Defining Duplicate Identification Batches* on page 5-5.

To review duplicates and define merge batches:

1. Navigate to the Duplicate Identification: Batch Review window.
2. Query the duplicate identification batch that you want to review.

For the batch, the window displays the information shown in this table :

Field	Value
Batch Name	The name of the duplicate identification batch.
Match Rule	The match rule that the DQM Batch Duplicate Identification Program used to match and score duplicate candidates.
Creation Date	The date that the duplicate identification batch was created.
Match Threshold	The match threshold defined in the match rule. Records with a score that exceeds the match threshold are selected as matches, or potential duplicates.

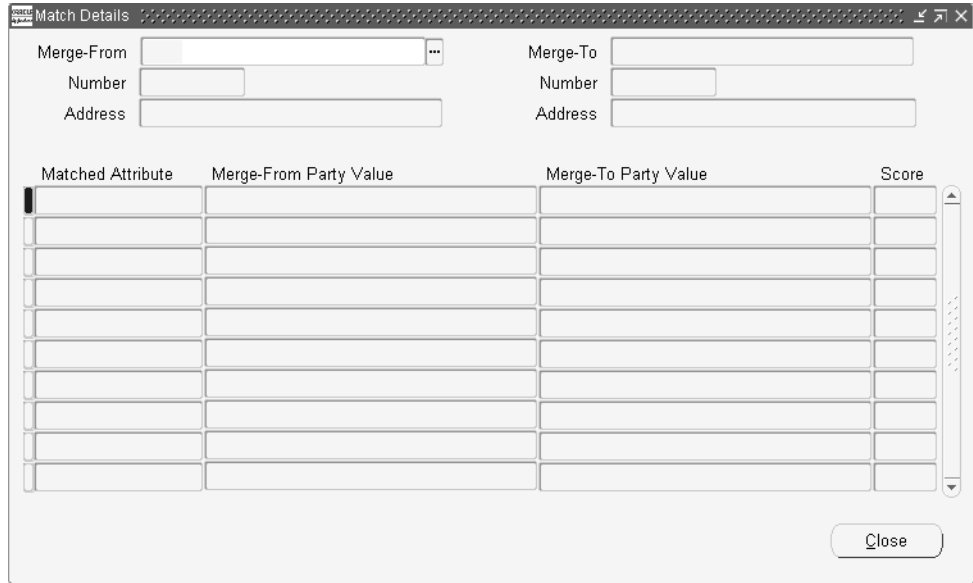
In the Merge-To Parties region, the Duplicate Identification: Batch Review window displays the information shown in this table for each merge-to party:

Field	Value
Name	The party name
Number	The party number
Address	The party address
Duplicates	The number of potential duplicates, or merge-from parties

In the Merge-From Parties region, the window displays all the duplicate candidates for the selected merge-to party. For each potential duplicate, you can see the information shown in this table:

Field	Value
Name	The party name.
Number	The party number.
Address	The party address.
Score	The match score, which signifies how closely the merge-from party matches the merge-to party. The score is the sum of scores that the match rule assigns to attribute matches between the two parties. The match score must exceed the match threshold of the match rule for the party to be selected as a merge-from party.
Merge	Yes or no to indicate whether to merge the party into the merge-to party or not. The default of this value depends on the automatic merge threshold of the match rule. A party with a match score that exceeds the threshold is defaulted to be merged.

- To view information about why a pair of merge-to and merge-from parties was designated as a potential duplicate match, select the merge-to and merge-from party in the window.
- Press the View Match Details button.
The Match Details window appears.



The window shows again the name, number, and address of the merge-from and merge-to parties.

The window also displays the information shown in this table:

Field	Value
Matched Attribute	The attribute that matches in both parties according to the match rule definitions.
Merge-From Party Value	The value of the matched attribute from the merge-from party.
Merge-To Party Value	The value of the matched attribute from the merge-to party.
Score	The score that the match rule assigns to the merge-from party for the particular attribute match to the merge-to party. The sum of these scores makes up the total match score of the merge-from party.

- To view attribute match details between the same merge-to party and another merge-from party, select another merge-from party in the list of values for the Merge-From field.

6. Press the Close button when you finish viewing match details for the selected merge-to party.
7. Repeat steps 3 to 6 for each merge-to party that you want to view match details for.
8. To switch the selected merge-from party to a merge-to party and vice versa, select a merge-to party in the Duplicate Identification: Batch Review window.
9. Press the Change Merge-To Party button and select the merge-from party that you want to replace the selected merge-to party.

Alternatively, you can use the list of values from the Merge-To Parties Name field to change the merge-to party.

10. For each merge-from party, specify in the Merge option whether to merge into the merge-to party or not. You can accept the defaults or, based on your evaluation, select not to merge parties that were defaulted for merge.

Note: You can only override Merge options that are set to Y.

11. If you want to specify that a merge-from party is not a duplicate match for the merge-to party, select the merge-from party and check the Not Duplicate of Merge-To Party check box.

Note: You can only select merge-from parties with the Merge option set to N.

12. You can optionally enter a date range for the merge-from party not to be selected as duplicate of that specific merge-to party. The current date defaults in the From Date field when you check the check box.
13. When you finish evaluating the batch, press the Create Merge Set button to create a merge batch that consists of merge-from parties with the Merge option set to Y and their corresponding merge-to parties.

The Review Party Merge Batches window automatically appears for the next procedure of submitting the sequence of parties in the batch to Party Merge. For more information, see Submitting Merge Batches on page 5-13.

If you decide later to change the Merge option to N for some parties in this batch, you can still do so in the Duplicate Identification: Batch Review window as long as you have not yet submitted the merge batch to Party Merge.

Submitting Merge Batches

Use the Review Party Merge Batches window to submit merge batches to Party Merge. Party Merge is the Oracle Trading Community Architecture feature that performs the actual merging of parties. The Review Party Merge Batches window displays all the merge batches that you created from a specific duplicate identification batch.

For each merge batch, the window also displays the pairs of parties to be merged and the sequence in which they are submitted. DQM automatically determines the sequence when you create the merge batch to ensure that parties are successfully submitted and merged.

For example, if you are merging party A to party B, and party B to party C, you must merge party A to B before merging B to C. If you merge party B to C first, party B does not exist any more for party A to merge into.

Prerequisites

Before you can submit merge batches to Party Merge, you must create a merge batch for the selected duplicate identification batch. For more information, see [Reviewing Duplicates and Creating Merge Batches](#) on page 5-8.

The screenshot shows a web application window titled "Review Party Merge Batches". At the top, there is a table with three columns: "Party Merge Batch", "Request ID", and "Status". Below this table is a section titled "Merge Batch Parties". This section is split into two sub-sections: "From Party" and "To Party". Each sub-section contains a table with two columns: "Party Name" and "Party Number". At the bottom right of the window, there are two buttons: "Submit Party Merge Batch" and "Go to Party Merge".

To submit merge batches:

1. Navigate to the Review Party Merge Batches window.

The window automatically appears when you create a merge batch. Otherwise, navigate to the Duplicate Identification: Batch Review window, query the duplicate identification batch that you want to submit merge batches for, and press the View Merge Set button.

2. You can change the merge batch names in the Party Merge Batch field if you want.

The default names consist of the original duplicate identification batch name and a sequential number. For example, the first merge batch that you create with duplicate identification batch ABC is named 1-ABC. The second merge batch is called 2-ABC, and so on.

The Status field displays the status of the merge batch:

- **Ready to Submit** - This batch is the one that is submitted when you press the Submit Party Merge Batch button.
- **In Queue** - This batch is in the queue for submission.
- **Pending** - This batch has been submitted to Party Merge.
- **Partially Complete** - This batch has been partially merged with success in Party Merge.
- **Complete** - This batch has successfully merged in Party Merge.

For each merge batch, the Merge Batch Parties region displays the names and numbers of all the merge-from and merge-to parties, in the default sequence that the pairs will be submitted.

3. Press the Submit Party Merge Batch button to submit the batch with the Ready to Submit status.

Note: You must submit batches in the order that they are displayed.

When the submission successfully completes, the status for the batch changes to Pending and the next batch gets the Ready to Submit status.

4. To continue to the Party Merge process, press the Go To Party Merge button. The Merge Parties window automatically appears for the selected merge batch.

Note: You can only select merge batches with a Pending status for Party Merge.

When the actual Party Merge process is run on a merge batch, the concurrent request number of the process is displayed in the Request ID field of the Review Party Merge Batches window.

See also: *Oracle Trading Community Architecture Party Merge User Guide*

A

Seed Data

This appendix describes the seeded attributes, transformation functions, and match rules.

Seeded Attributes

Seeded attributes are provided for the logical entities of party, party site, contact, and contact point. These attributes are used for matching in the staged schema. Each attribute corresponds to a column in a HZ table, such as HZ_PARTIES. A table exists in the staged schema for each logical entity, and each of the four tables has 30 custom attribute columns that you can use to add attributes that are not seeded. For more information about attributes, see [Attributes Overview](#).

Seeded attributes are defined as lookup codes, and the lookup meaning is what appears as the attribute name in the Define Attributes and Transformation Functions window.

The staged schema also includes special logical attributes to facilitate matching for more frequently used and ambiguously defined attributes:

- **Address** - Concatenates address information from the ADDRESS1, ADDRESS2, ADDRESS3, and ADDRESS 4 columns in the HZ_LOCATIONS table
- **All Account Names** - Concatenates all account names for a party from the ACCOUNT_NAME column in the HZ_CUST_ACCOUNTS table
- **All Account Numbers** - Concatenates all account numbers for a party from the ACCOUNT_NUMBER column in the HZ_CUST_ACCOUNTS table
- **Concatenated Party Names** - Concatenates all party names including known-as names from the PARTY_NAME, KNOWN_AS, KNOWN_AS2, KNOWN_AS3, KNOWN_AS4, and KNOWN_AS5 columns in the HZ_PARTIES table
- **Phone Number Flexible Format** - Concatenates phone numbers in the format of phone number, raw phone number, phone country code, and raw phone number from the PHONE_NUMBER, RAW_PHONE_NUMBER, PHONE_COUNTRY_CODE, and RAW_PHONE_NUMBER columns in the HZ_CONTACT_POINTS table

You cannot apply similarity algorithms to these attributes because the attributes consist of a concatenation of columns. The similarity algorithm would apply to the entire concatenation and likely produce unusable results. For more information about the similarity algorithm, see [Defining the Scoring Component of a Match Rule](#) on page 3-10.

Party Attributes

The attributes for the party logical entity are stored in the HZ_STAGED_PARTIES table of the staged schema. The source tables for the attribute values include HZ_PARTIES, HZ_ORGANIZATION_PROFILES, and HZ_PERSON_PROFILES. This table shows the seeded attributes and their source table from the TCA registry.

Attribute Name	Attribute Code	Source Table
All Account Names	ALL_ACCOUNT_NAMES	<Logical attribute>
All Account Numbers	ALL_ACCOUNT_NUMBERS	<Logical attribute>
Average High Credit	AVE_HIGH_CREDIT	HZ_ORGANIZATION_PROFILES
Best Time to Begin Contact	BEST_TIME_CONTACT_BEGIN	HZ_ORGANIZATION_PROFILES or HZ_PERSON_PROFILES
Best Time to End Contact	BEST_TIME_CONTACT_END	HZ_ORGANIZATION_PROFILES or HZ_PERSON_PROFILES
Branch Flag	BRANCH_FLAG	HZ_ORGANIZATION_PROFILES
Business Scope	BUSINESS_SCOPE	HZ_ORGANIZATION_PROFILES
Category Code	CATEGORY_CODE	HZ_PARTIES
CEO Name	CEO_NAME	HZ_ORGANIZATION_PROFILES
CEO Title	CEO_TITLE	HZ_ORGANIZATION_PROFILES
Competitor Flag	COMPETITOR_FLAG	HZ_PARTIES
Concatenated Party Names	PARTY_ALL_NAMES	<Logical attribute>
Congressional District Code	CONG_DIST_CODE	HZ_ORGANIZATION_PROFILES
Content Source Number	CONTENT_SOURCE_NUMBER	HZ_ORGANIZATION_PROFILES or HZ_PERSON_PROFILES

Attribute Name	Attribute Code	Source Table
Content Source Type	CONTENT_SOURCE_TYPE	HZ_ORGANIZATION_PROFILES or HZ_PERSON_PROFILES
Control Year	CONTROL_YR	HZ_ORGANIZATION_PROFILES
Corporation Class	CORPORATION_CLASS	HZ_ORGANIZATION_PROFILES
Credit Score	CREDIT_SCORE	HZ_ORGANIZATION_PROFILES
Credit Score Age	CREDIT_SCORE_AGE	HZ_ORGANIZATION_PROFILES
Credit Score Class	CREDIT_SCORE_CLASS	HZ_ORGANIZATION_PROFILES
Credit Score Commentary	CREDIT_SCORE_COMMENTARY	HZ_ORGANIZATION_PROFILES
Credit Score Commentary 2 through Credit Score Commentary 10	CREDIT_SCORE_COMENTARY2 through CREDIT_SCORE_COMMENTARY10	HZ_ORGANIZATION_PROFILES
Credit Score Date	CREDIT_SCORE_DATE	HZ_ORGANIZATION_PROFILES
Credit Score Incident Default	CREDIT_SCORE_INCD_DEFAULT	HZ_ORGANIZATION_PROFILES
Credit Score National Percentile	CREDIT_SCORE_NATL_PERCENTILE	HZ_ORGANIZATION_PROFILES
CUSTOM ATTRIBUTE 1 through CUSTOM ATTRIBUTE 30	CUSTOM_ATTRIBUTES1 through CUSTOM_ATTRIBUTES30	<Custom attribute>
Date of Birth	DATE_OF_BIRTH	HZ_PERSON_PROFILES
Date of Death	DATE_OF_DEATH	HZ_PERSON_PROFILES
Debarment Indicator	DEBARMENT_IND	HZ_ORGANIZATION_PROFILES
Debarments Count	DEBARMENTS_COUNT	HZ_ORGANIZATION_PROFILES
Debarments Date	DEBARMENTS_DATE	HZ_ORGANIZATION_PROFILES

Attribute Name	Attribute Code	Source Table
Declared Ethnicity	DECLARED_ETHNICITY	HZ_PERSON_PROFILES
Disadvantaged Under Title 8A Indicator	DISADV_8A_IND	HZ_ORGANIZATION_PROFILES
Dun & Bradstreet Credit Rating	DB_RATING	HZ_ORGANIZATION_PROFILES
DUNS Number	DUNS_NUMBER	HZ_ORGANIZATION_PROFILES
Effective End Date	EFFECTIVE_END_DATE	HZ_ORGANIZATION_PROFILES
Effective Start Date	EFFECTIVE_START_DATE	HZ_ORGANIZATION_PROFILES
Enquiry DUNS	ENQUIRY_DUNS	HZ_ORGANIZATION_PROFILES
Export Indicator	EXPORT_IND	HZ_ORGANIZATION_PROFILES
Failure Score	FAILURE_SCORE	HZ_ORGANIZATION_PROFILES
Failure Score Age	FAILURE_SCORE_AGE	HZ_ORGANIZATION_PROFILES
Failure Score Class	FAILURE_SCORE_CLASS	HZ_ORGANIZATION_PROFILES
Failure Score Commentary	FAILURE_SCORE_COMMENTARY	HZ_ORGANIZATION_PROFILES
Failure Score Commentary 2 through Failure Score Commentary 10	FAILURE_SCORE_COMMENTARY2 through FAILURE_SCORE_COMMENTARY10	HZ_ORGANIZATION_PROFILES
Failure Score Date	FAILURE_SCORE_DATE	HZ_ORGANIZATION_PROFILES
Failure Score Incident Default	FAILURE_SCORE_INCD_DEFAULT	HZ_ORGANIZATION_PROFILES
Failure Score Override Code	FAILURE_SCORE_OVERRIDE_CODE	HZ_ORGANIZATION_PROFILES
Fiscal Year for Financial Analysis	ANALYSIS_FY	HZ_ORGANIZATION_PROFILES

Attribute Name	Attribute Code	Source Table
Gender	GENDER	HZ_PERSON_PROFILES
Global Failure Score	GLOBAL_FAILURE_SCORE	HZ_ORGANIZATION_PROFILES
Group Type	GROUP_TYPE	HZ_PARTIES
GSA Indicator Flag	GSA_INDICATOR_FLAG	HZ_ORGANIZATION_PROFILES
Head of Household Flag	HEAD_OF_HOUSEHOLD_FLAG	HZ_PERSON_PROFILES
Highest Credit	HIGH_CREDIT	HZ_ORGANIZATION_PROFILES
Household Income	HOUSEHOLD_INCOME	HZ_PERSON_PROFILES
Household Size	HOUSEHOLD_SIZE	HZ_PERSON_PROFILES
HQ Branch Indicator	HQ_BRANCH_IND	HZ_ORGANIZATION_PROFILES
Import Indicator	IMPORT_IND	HZ_ORGANIZATION_PROFILES
Internal Flag	INTERNAL_FLAG	HZ_ORGANIZATION_PROFILES or HZ_PERSON_PROFILES
JGZZ Fiscal Code	JGZZ_FISCAL_CODE	HZ_ORGANIZATION_PROFILES or HZ_PERSON_PROFILES
Known As 1	KNOWN_AS	HZ_ORGANIZATION_PROFILES or HZ_PERSON_PROFILES
Known As 2 through Known As 5	KNOWN_AS2 through KNOWN_AS5	HZ_ORGANIZATION_PROFILES or HZ_PERSON_PROFILES
Labor Surplus Indicator	LABOR_SURPLUS_IND	HZ_ORGANIZATION_PROFILES
Language Name	LANGUAGE_NAME	HZ_PARTIES
Last Known GPS	LAST_KNOWN_GPS	HZ_PERSON_PROFILES
Last Month of Fiscal Year	FISCAL YEAREND_MONTH	HZ_ORGANIZATION_PROFILES

Attribute Name	Attribute Code	Source Table
Legal Structure	LEGAL_STATUS	HZ_ORGANIZATION_PROFILES
Line of Business	LINE_OF_BUSINESS	HZ_ORGANIZATION_PROFILES
Local Activity Classification Code	LOCAL_ACTIVITY_CODE	HZ_ORGANIZATION_PROFILES
Local Activity Classification Code Type	LOCAL_ACTIVITY_CODE_TYPE	HZ_ORGANIZATION_PROFILES
Local Business Identifier	LOCAL_BUS_IDEN_TYPE	HZ_ORGANIZATION_PROFILES
Local Business Identifier Type	LOCAL_BUS_IDENTIFIER	HZ_ORGANIZATION_PROFILES
Marital Status	MARITAL_STATUS	HZ_PERSON_PROFILES
Marital Status Effective Date	MARITAL_STATUS_EFFECTIVE_DATE	HZ_PERSON_PROFILES
Maximum Credit Currency Code	MAXIMUM_CREDIT_CURRENCY_CORE	HZ_ORGANIZATION_PROFILES
Maximum Credit Recommendation	MAXIMUM_CREDIT_RECOMMENDATION	HZ_ORGANIZATION_PROFILES
Min/Max/Ave Number of Employees at Primary Address Indicator	EMP_AT_PRIMARY_ADR_MIN_IND	HZ_ORGANIZATION_PROFILES
Minority Owned Indicator	MINORITY_OWNED_IND	HZ_ORGANIZATION_PROFILES
Minority Type	MINORITY_OWNED_TYPE	HZ_ORGANIZATION_PROFILES
Number of Employees	EMPLOYEES_TOTAL	HZ_ORGANIZATION_PROFILES
Number of Employees at Primary Address	EMP_AT_PRIMARY_ADR	HZ_ORGANIZATION_PROFILES
Number of Employees Estimation Indicator	EMP_AT_PRIMARY_ADR_EST_IND	HZ_ORGANIZATION_PROFILES
Organization Name	ORGANIZATION_NAME	HZ_ORGANIZATION_PROFILES

Attribute Name	Attribute Code	Source Table
Organization Type	ORGANIZATION_TYPE	HZ_ORGANIZATION_PROFILES
Out of Business Indicator	OOB_IND	HZ_ORGANIZATION_PROFILES
Parent/Subsidiary Indicator	PARENT_SUB_IND	HZ_ORGANIZATION_PROFILES
Party Name	PARTY_NAME	HZ_PARTIES
Party Number	PARTY_NUMBER	HZ_PARTIES
Party Type	PARTY_TYPE	HZ_PARTIES
Paydex Norm	PAYDEX_NORM	HZ_ORGANIZATION_PROFILES
Paydex Score	PAYDEX_SCORE	HZ_ORGANIZATION_PROFILES
Paydex Score Three Months Ago	PAYDEX_THREE_MONTHS_AGO	HZ_ORGANIZATION_PROFILES
Person Academic Title	PERSON_ACADEMIC_TITLE	HZ_PERSON_PROFILES
Person First Name	PERSON_FIRST_NAME	HZ_PERSON_PROFILES
Person Initials	PERSON_INITIALS	HZ_PERSON_PROFILES
Person Last Name	PERSON_LAST_NAME	HZ_PERSON_PROFILES
Person Middle Name	PERSON_MIDDLE_NAME	HZ_PERSON_PROFILES
Person Middle Name Phonetic	MIDDLE_NAME_PHONETIC	HZ_PERSON_PROFILES
Person Name	PERSON_NAME	HZ_PERSON_PROFILES
Person Name Suffix	PERSON_NAME_SUFFIX	HZ_PERSON_PROFILES
Person Pre-Name Adjunct	PERSON_PRE_NAME_ADJUNCT	HZ_PERSON_PROFILES
Person Previous Last Name	PERSON_PREVIOUS_LAST_NAME	HZ_PERSON_PROFILES
Person Title	PERSON_TITLE	HZ_PERSON_PROFILES
Personal Identification	PERSON_IDENTIFIER	HZ_PERSON_PROFILES
Personal Income	PERSONAL_INCOME	HZ_PERSON_PROFILES

Attribute Name	Attribute Code	Source Table
Phonetic Representation of Organization Name	ORGANIZATION_NAME_PHONETIC	HZ_ORGANIZATION_PROFILES
Phonetic Representation of Person First Name	PERSON_FIRST_NAME_PHONETIC	HZ_PERSON_PROFILES
Phonetic Representation of Person Last Name	PERSON_LAST_NAME_PHONETIC	HZ_PERSON_PROFILES
Phonetic Representation of Person Name	PERSON_NAME_PHONETIC	HZ_PERSON_PROFILES
Place of Birth	PLACE_OF_BIRTH	HZ_PERSON_PROFILES
Potential Revenue of Current Fiscal Year	CURR_FY_POTENTIAL_REVENUE	HZ_ORGANIZATION_PROFILES
Potential Revenue of the Next Fiscal Year	NEXT_FY_POTENTIAL_REVENUE	HZ_ORGANIZATION_PROFILES
Preferred Functional Currency	PREF_FUNCTIONAL_CURRENCY	HZ_ORGANIZATION_PROFILES
Princinal Title	PRINCIPAL_TITLE	HZ_ORGANIZATION_PROFILES
Principal Name	PRINCIPAL_NAME	HZ_ORGANIZATION_PROFILES
Private Ownership Flag	PUBLIC_PRIVATE_OWNERSHIP_FLAG	HZ_ORGANIZATION_PROFILES
Reference Use Flag	REFERENCE_USE_FLAG	HZ_PARTIES
Registration Type	REGISTRATION_TYPE	HZ_ORGANIZATION_PROFILES
Rent or Own Indicator	RENT_OWN_IND	HZ_ORGANIZATION_PROFILES or HZ_PERSON_PROFILES
Salutation	SALUTATION	HZ_PARTIES
SIC Code	SIC_CODE	HZ_ORGANIZATION_PROFILES
SIC Code Version	SIC_CODE_TYPE	HZ_ORGANIZATION_PROFILES
Small Business Indicator	SMALL_BUS_IND	HZ_ORGANIZATION_PROFILES

Attribute Name	Attribute Code	Source Table
Status	STATUS	HZ_PARTIES
Tax Name	TAX_NAME	HZ_ORGANIZATION_PROFILES or HZ_PERSON_PROFILES
Tax Reference	TAX_REFERENCE	HZ_ORGANIZATION_PROFILES or HZ_PERSON_PROFILES
Text Number of Employees at Primary Address	EMP_AT_PRIMARY_ADR_TEXT	HZ_ORGANIZATION_PROFILES
Third Party Flag	THIRD_PARTY_FLAG	HZ_PARTIES
Total Number of Employees Estimation Indicator	TOTAL_EMP_EST_IND	HZ_ORGANIZATION_PROFILES
Total Number of Employees in Text Format	TOTAL_EMPLOYEES_TEXT	HZ_ORGANIZATION_PROFILES
Total Number of Employees Indicator	TOTAL_EMPLOYEES_IND	HZ_ORGANIZATION_PROFILES
Total Number of Employees Min/Max/Ave Indicator	TOTAL_EMP_MIN_IND	HZ_ORGANIZATION_PROFILES
Total Payments	TOTAL_PAYMENTS	HZ_ORGANIZATION_PROFILES
Type of Personal Identification	PERSON_IDEN_TYPE	HZ_PERSON_PROFILES
Validated Flag	VALIDATED_FLAG	HZ_PARTIES
Woman Owned Index	WOMAN_OWNED_IND	HZ_ORGANIZATION_PROFILES
Year Established	YEAR_ESTABLISHED	HZ_ORGANIZATION_PROFILES
Year Incorporated	INCORP_YEAR	HZ_ORGANIZATION_PROFILES

Address Attributes

The attributes for the address logical entity are stored in the HZ_STAGED_PARTY_SITES table of the staged schema. The source tables for the attribute values include HZ_LOCATIONS and HZ_PARTY_SITES. This table shows the seeded attributes and their source table from the TCA registry.

Attribute Name	Attribute Code	Source Table
Address	Address	<Logical attribute>
Address 1 through Address 4	ADDRESS1 through ADDRESS4	HZ_LOCATIONS
Address Effective Date	ADDRESS_EFFECTIVE_DATE	HZ_LOCATIONS
Address Expiration Date	ADDRESS_EXPIRATION_DATE	HZ_LOCATIONS
Address Phonetic Representation	ADDRESS_LINES_PHONETIC	HZ_LOCATIONS
City	CITY	HZ_LOCATIONS
CLLI Code	CLLI_CODE	HZ_LOCATIONS
Content Source Type	CONTENT_SOURCE_TYPE	HZ_LOCATIONS
Country	COUNTRY	HZ_LOCATIONS
County	COUNTY	HZ_LOCATIONS
CUSTOM ATTRIBUTE 1 through CUSTOM ATTRIBUTE 30	CUSTOM_ATTRIBUTES1 through CUSTOM_ATTRIBUTES30	<Custom attribute>
Floor	FLOOR	HZ_LOCATIONS
House Number	HOUSE_NUMBER	HZ_LOCATIONS
Identifying Address Flag	IDENTIFYING_ADDRESS_FLAG	HZ_PARTY_SITES
Language	LANGUAGE	HZ_LOCATIONS
Mailstop	MAILSTOP	HZ_PARTY_SITES
P.O. Box Number	PO_BOX_NUMBER	HZ_LOCATIONS
Party Site Name	PARTY_SITE_NAME	HZ_PARTY_SITES
Party Site Number	PARTY_SITE_NUMBER	HZ_PARTY_SITES
Position	POSITION	HZ_LOCATIONS

Attribute Name	Attribute Code	Source Table
Postal Code	POSTAL_CODE	HZ_LOCATIONS
Postal Code Extension	POSTAL_PLUS4_CODE	HZ_LOCATIONS
Province	PROVINCE	HZ_LOCATIONS
Sales Tax Geocode	SALES_TAX_GEOCODE	HZ_LOCATIONS
Sales Tax Inside City Limits	SALES_TAX_INSIDE_CITY_LIMITS	HZ_LOCATIONS
State	STATE	HZ_LOCATIONS
Street	STREET	HZ_LOCATIONS
Street Number	STREET_NUMBER	HZ_LOCATIONS
Street Suffix	STREET_SUFFIX	HZ_LOCATIONS
Suite	SUITE	HZ_LOCATIONS
Validated Flag	VALIDATED_FLAG	HZ_LOCATIONS

Contact Attributes

The attributes for the contact logical entity are stored in the HZ_STAGED_CONTACTS table of the staged schema. The source tables for the attribute values include HZ_ORG_CONTACTS, HZ_PERSON_PROFILES, and HZ_RELATIONSHIPS. This table shows the seeded attributes and their source table from the TCA registry.

Attribute Name	Attribute Code	Source Table
Best Time to Begin Contact	BEST_TIME_CONTACT_BEGIN	HZ_PERSON_PROFILES
Best Time to End Contact	BEST_TIME_CONTACT_END	HZ_PERSON_PROFILES
Contact Name	CONTACT_NAME	<Custom attribute>
Contact Number	CONTACT_NUMBER	HZ_ORG_CONTACTS
Content Source Type	CONTENT_SOURCE_TYPE	HZ_RELATIONSHIPS
CUSTOM ATTRIBUTE 1 through CUSTOM ATTRIBUTE 30	CUSTOM_ATTRIBUTES1 through CUSTOM_ATTRIBUTES30	<Custom attribute>
Date of Birth	DATE_OF_BIRTH	HZ_PERSON_PROFILES
Date of Death	DATE_OF_DEATH	HZ_PERSON_PROFILES
Decision Maker Flag	DECISION_MAKER_FLAG	HZ_ORG_CONTACTS
Directional Flag	DIRECTIONAL_FLAG	HZ_RELATIONSHIPS
JGZZ Fiscal Code	JGZZ_FISCAL_CODE	HZ_PERSON_PROFILES
Job Title	JOB_TITLE	HZ_ORG_CONTACTS
Known As	KNOWN_AS	HZ_PERSON_PROFILES
Person Academic Title	PERSON_ACADEMIC_TITLE	HZ_PERSON_PROFILES
Person First Name	PERSON_FIRST_NAME	HZ_PERSON_PROFILES
Person Initials	PERSON_INITIALS	HZ_PERSON_PROFILES
Person Last Name	PERSON_LAST_NAME	HZ_PERSON_PROFILES
Person Middle Name	PERSON_MIDDLE_NAME	HZ_PERSON_PROFILES
Person Name	PERSON_NAME	HZ_PERSON_PROFILES
Person Name Suffix	PERSON_NAME_SUFFIX	HZ_PERSON_PROFILES

Attribute Name	Attribute Code	Source Table
Person Previous Last Name	PERSON_PREVIOUS_LAST_NAME	HZ_PERSON_PROFILES
Person Title	PERSON_TITLE	HZ_PERSON_PROFILES
Personal Identification	PERSON_IDENTIFIER	HZ_PERSON_PROFILES
Personal Identification Type	PERSON_IDEN_TYPE	HZ_PERSON_PROFILES
Phonetic Representation of Person First Name	PERSON_FIRST_NAME_PHONETIC	HZ_PERSON_PROFILES
Phonetic Representation of Person Last Name	PERSON_LAST_NAME_PHONETIC	HZ_PERSON_PROFILES
Phonetic Representation of Person Name	PERSON_NAME_PHONETIC	HZ_PERSON_PROFILES
Place of Birth	PLACE_OF_BIRTH	HZ_PERSON_PROFILES
Rank	RANK	HZ_ORG_CONTACTS
Reference Use Flag	REFERENCE_USE_FLAG	HZ_ORG_CONTACTS
Relationship Type	RELATIONSHIP_TYPE	HZ_RELATIONSHIPS
Tax Name	TAX_NAME	HZ_PERSON_PROFILES
Tax Reference	TAX_REFERENCE	HZ_PERSON_PROFILES
Title	TITLE	HZ_ORG_CONTACTS

Contact Point Attributes

The attributes for the contact point logical entity are stored in the HZ_STAGED_CONTACT_POINTS table of the staged schema. The source tables for the attribute values is HZ_CONTACT_POINTS. This table shows the seeded attributes and their source table from the TCA registry.

Attribute Name	Attribute Code	Source Table
Contact Point Type	CONTACT_POINT_TYPE	HZ_CONTACT_POINTS
CUSTOM ATTRIBUTE 1 through CUSTOM ATTRIBUTE 30	CUSTOM_ATTRIBUTES1 through CUSTOM_ATTRIBUTES30	<Custom attribute>
EDI ECE TP Location Code	EDI_ECE_TP_LOCATION_CODE	HZ_CONTACT_POINTS
EDI ID Number	EDI_ID_NUMBER	HZ_CONTACT_POINTS
EDI Payment Format	EDI_PAYMENT_FORMAT	HZ_CONTACT_POINTS
EDI Payment Method	EDI_PAYMENT_METHOD	HZ_CONTACT_POINTS
EDI Remittance Instruction	EDI_REMITTANCE_INSTRUCTION	HZ_CONTACT_POINTS
EDI Remittance Method	EDI_REMITTANCE_METHOD	HZ_CONTACT_POINTS
EDI TP Header ID	EDI_TP_HEADER_ID	HZ_CONTACT_POINTS
EDI Transaction Handling	EDI_TRANSACTION_HANDLING	HZ_CONTACT_POINTS
Email Address	EMAIL_ADDRESS	HZ_CONTACT_POINTS
Email Format	EMAIL_FORMAT	HZ_CONTACT_POINTS
Phone Number Flexible Format	FLEX_FORMAT_PHONE_NUMBER	<Logical attribute>
Last Contact Day and Time	LAST_CONTACT_DT_TIME	HZ_CONTACT_POINTS
Phone Area Code	PHONE_AREA_CODE	HZ_CONTACT_POINTS
Phone Calling Calendar	PHONE_CALLING_CALENDAR	HZ_CONTACT_POINTS
Phone Country Code	PHONE_COUNTRY_CODE	HZ_CONTACT_POINTS
Phone Extension	PHONE_EXTENSION	HZ_CONTACT_POINTS

Attribute Name	Attribute Code	Source Table
Phone Line Type	PHONE_LINE_TYPE	HZ_CONTACT_POINTS
Phone Number	PHONE_NUMBER	HZ_CONTACT_POINTS
Primary Flag	PRIMARY_FLAG	HZ_CONTACT_POINTS
Raw Phone Number	RAW_PHONE_NUMBER	HZ_CONTACT_POINTS
Phone Type	TELEPHONE_TYPE	HZ_CONTACT_POINTS
Telex Number	TELEX_NUMBER	HZ_CONTACT_POINTS
Time Zone	TIME_ZONE	HZ_CONTACT_POINTS
URL	URL	HZ_CONTACT_POINTS
Web Type	WEB_TYPE	HZ_CONTACT_POINTS

Seeded Transformation Functions

Transformation functions are preconfigured for the logical entities of party, address, contact, and contact point. You can also create your own transformation functions. For more information about transformation functions, see Transformation Functions Overview on page 2-3.

Transformation functions, seeded or not, can also contain word replacements. For more information about word replacements, see Word Replacements Overview on page 2-14.

This table shows the seeded transformation functions, the function and purpose of each, as well as examples of the transformation.

Transformation Function Name	PL/SQL Function Name	Operations	Purpose	Example Input	Example Output
CLEANSE	HZ_TRANS_PKG.CLEANSE	Capitalize all letters Remove nonalphanumeric characters Reduce all instances of white space to a single white space Remove double letters Remove vowels except initial vowels	To catch incorrect vowel usage and typing errors	D' Angello	D ANGL
CLEANSE (EMAIL)	HZ_TRANS_PKG.CLEANSED_EMAIL	Replace domain name words Capitalize all letters Replace nonalphanumeric characters with white space Reduce all instances of white space to a single white space Remove double letters Remove vowels except initial vowels	To catch incorrect vowel usage and typing errors as well as mistakes with domain names	joe.smith@oracle.co	J SMTH ORCL

Transformation Function Name	PL/SQL Function Name	Operations	Purpose	Example Input	Example Output
CLEANSE (URL)	HZ_TRANS_PKG.CLEANSSED_URL	Replace <i>http://</i> and other URL protocols Capitalize all letters Replace nonalphanumeric characters with white space Reduce all instances of white space to a single white space Replace domain names Remove double letters Remove vowels except initial vowels Keep only first five words	To catch incorrect vowel usage and typing errors as well as mistakes with domain names	http://ww.oracle.co	ORCL
CLUSTER	HZ_TRANS_PKG.CLUSTER_WORD	Capitalize all letters Remove nonalphanumeric characters Reduce all instances of white space to a single white space Remove all but first four letters of the first two words	To catch spelling errors at the end of the word and minor format errors	D' Angello	D ANGE
EXACT	HZ_TRANS_PKG.EXACT	Capitalize all letters Remove nonalphanumeric characters Reduce all instances of white space to a single white space	To catch format errors	D' Angello	D ANGELLO
EXACT (DATE)	HZ_TRANS_PKG.EXACT_DATE	Converts date format to character format	To enable character comparison for dates	01-JAN-2002	01-JAN-2002
EXACT (EMAIL)	HZ_TRANS_PKG.EXACT_EMAIL	Capitalizes all letters	To catch case errors	Joe.smith@oracle.com	JOE.SMITH@ORACLE.COM

Transformation Function Name	PL/SQL Function Name	Operations	Purpose	Example Input	Example Output
EXACT (NUMBER)	HZ_TRANS_PKG.RM_SPLCHAR	Converts numeric format to character format	To enable character comparison for numbers	100	100
EXACT (URL)	HZ_TRANS_PKG.EXACT_URL	Replace <i>http://</i> and other URL protocols Capitalize all letters	To catch case errors	<i>http://</i> WWW.oracle.com	WWW. ORACLE. COM
EXACT STRING	HZ_TRANS_PKG.EXACT_PADDED	Capitalize all letters Remove nonalphanumeric characters Reduce all instances of white space to a single white space Pads the string with # on both sides	To capture the exact string and catch format errors	D' Angello	#D ANGELLO#
REV + WR NAMES + CLEANSE	HZ_TRANS_PKG.REVERSE_WRNAMES_CLEANSE	Reverse the last word as the first word Capitalize all letters Remove nonalphanumeric characters Reduce all instances of white space to a single white space Replace words for person or organization names Remove double letters Remove vowels except initial vowels	To catch incorrect ordering of words, as well as nicknames, abbreviations, and spelling variations for person or organization names and incorrect vowel usage and typing errors	Corp D' Angello	D NGL CRPRTN

Transformation Function Name	PL/SQL Function Name	Operations	Purpose	Example Input	Example Output
REV + WR PER + CLEANSE	HZ_TRANS_PKG.REVERSE_WRPERSON_CLEANSE	Reverse the last word as the first word Capitalize all letters Remove nonalphanumeric characters Reduce all instances of white space to a single white space Replace words for person names Remove double letters Remove vowels except initial vowels	To catch incorrect ordering of words, as well as nicknames, abbreviations, and spelling variations for person names and incorrect vowel usage and typing errors	D' Angello Mike	MCHL D ANGL
REV + WR PER + CLUSTER	HZ_TRANS_PKG.REVERSE_WRPERSON_CLUSTER	Reverse the last word as the first word Capitalize all letters Remove nonalphanumeric characters Reduce all instances of white space to a single white space Replace words for person names Remove all but first four letters of the first two words	To catch incorrect ordering of words, as well as nicknames, abbreviations, and spelling variations for person names and spelling errors at the end of the word and minor format errors	Inc Mitchell	MITC INCO

Transformation Function Name	PL/SQL Function Name	Operations	Purpose	Example Input	Example Output
REV + WR NAMES + CLUSTER	HZ_TRANS_PKG.REVERSE_WRNAMES_CLUSTER	Reverse the last word as the first word Capitalize all letters Remove nonalphanumeric characters Reduce all instances of white space to a single white space Replace words for person or organization names Remove all but first four letters of the first two words	To catch incorrect ordering of words, as well as nicknames, abbreviations, and spelling variations for person or organization names and spelling errors at the end of the word and minor format errors	Inc Mitchell	MITC INCO
REVERSE	HZ_TRANS_PKG.REVERSE_NAME	Reverse the last word as the first word	To catch incorrect ordering of words	Madison Mitchell	Mitchell Madison
SOUNDEX	HZ_TRANS_PKG.SOUNDX	Group words by sound instead of exact spelling by translating words into a four-digit code consisting of one letter and three numbers	To catch spelling errors	Smith	S260
WR ADDRESS	HZ_TRANS_PKG.WRADDRESS_EXACT	Replace words for addresses	To catch abbreviations for words used in the address	W. Main St.	WEST Main STREET
WR ADDRESS + CLEANSE	HZ_TRANS_PKG.WRADDRESS_CLEANSE	Capitalize all letters Remove nonalphanumeric characters Reduce all instances of white space to a single white space Replace words for addresses Remove double letters Remove vowels except initial vowels	To catch abbreviations for words used in the address as well as incorrect vowel usage and typing errors	W. Main St.	WST MN STRT

Transformation Function Name	PL/SQL Function Name	Operations	Purpose	Example Input	Example Output
WR NAMES	HZ_TRANS_PKG. WRNAMES_EXACT	Replace words for organization and person names	To catch nicknames, abbreviations, and spelling variations for person or organization names	Oracle Corp	Oracle CORPORATI ON
WR NAMES + CLEANSE	HZ_TRANS_PKG. WRNAMES_CLEANSE	Replace words for organization and person names Capitalize all letters Remove nonalphanumeric characters Reduce all instances of white space to a single white space Remove double letters Remove vowels except initial vowels	To catch nicknames, abbreviations, and spelling variations for person or organization names as well as incorrect vowel usage and typing errors	D' Angello Corp	D NGL CRPRTN
WR NAMES + CLUSTER	HZ_TRANS_PKG. WRNAMES_CLUSTER	Capitalize all letters Remove nonalphanumeric characters Reduce all instances of white space to a single white space Replace words for organization and person names Remove all but first four letters of the first two words	To catch nicknames, abbreviations, and spelling variations for person or organization names as well as spelling errors at the end of the word and minor format errors	D' Angello Corp	D ANGE

Transformation Function Name	PL/SQL Function Name	Operations	Purpose	Example Input	Example Output
WR ORG + CLEANSE	HZ_TRANS_ PKG.WRORG_ CLEANSE	<p>Capitalize all letters</p> <p>Remove nonalphanumeric characters</p> <p>Reduce all instances of white space to a single white space</p> <p>Replace words for organization names</p> <p>Remove double letters</p> <p>Remove vowels except initial vowels</p>	To catch nicknames, abbreviations, and spelling variations for organization names as well as incorrect vowel usage and typing errors	Oracle Corp	ORCL CRPRTN
WR ORG + CLUSTER	HZ_TRANS_ PKG.WRORG_ CLUSTER	<p>Capitalize all letters</p> <p>Remove nonalphanumeric characters</p> <p>Reduce all instances of white space to a single white space</p> <p>Replace words for organization names</p> <p>Remove all but first four letters of the first two words</p>	To catch nicknames, abbreviations, and spelling variations for organization names as well as spelling errors at the end of the word and minor format errors	Mitchell Inc	MITC INCO
WR ORGANIZATIO N	HZ_TRANS_ PKG.WRORG_ EXACT	<p>Replace words for organization names</p>	To catch nicknames, abbreviations, and spelling variations for organization names	Oracle Corp	Oracle CORPORATI ON
WR PERSON	HZ_TRANS_ PKG. WRPERSON_ EXACT	<p>Capitalize all letters</p> <p>Remove nonalphanumeric characters</p> <p>Reduce all instances of white space to a single white space</p> <p>Replace words for person names</p>	To catch nicknames, abbreviations, and spelling variations for person names	Mike D' Angello	MICHAEL D ANGELLO

Transformation Function Name	PL/SQL Function Name	Operations	Purpose	Example Input	Example Output
WR PERSON + CLEANSE	HZ_TRANS_PKG.WRPERSON_CLEANSE	<p>Capitalize all letters</p> <p>Remove nonalphanumeric characters</p> <p>Reduce all instances of white space to a single white space</p> <p>Replace words for person names</p> <p>Remove double letters</p> <p>Remove vowels except initial vowels</p>	To catch nicknames, abbreviations, and spelling variations for person names as well as incorrect vowel usage and typing errors	Mike D'Angello	MCHL D ANGL
WR PERSON + CLUSTER	HZ_TRANS_PKG.WRPERSON_CLUSTER	<p>Capitalize all letters</p> <p>Remove nonalphanumeric characters</p> <p>Reduce all instances of white space to a single white space</p> <p>Replace words for person names</p> <p>Remove all but first four letters of the first two words</p>	To catch nicknames, abbreviations, and spelling variations for person names as well as spelling errors at the end of the word and minor format errors	Mike D'Angello	MICH D
WR STATE	HZ_TRANS_PKG.WRSTATE_EXACT	<p>Replace words for state</p>	To catch spelling errors and incorrect abbreviations for state	CA	California
WR STATE + CLEANSE	HZ_TRANS_PKG.WRSTATE_CLEANSE	<p>Capitalize all letters</p> <p>Remove nonalphanumeric characters</p> <p>Reduce all instances of white space to a single white space</p> <p>Replace words for state</p> <p>Remove double letters</p> <p>Remove vowels except initial vowels</p>	To catch spelling errors and incorrect abbreviations for state as well as incorrect vowel usage and typing errors	CA	CLFRN

Seeded Match Rules

DQM provides seven seeded match rules that you can use or base your custom match rules on. You can use these six seeded match rules for batch duplicate identification to identify duplicates that currently exist within your TCA registry:

- SAMPLE: IDENTICAL_ORGANIZATIONS
- SAMPLE: IDENTICAL_PERSON
- SAMPLE: ADDRESS_ORGANIZATIONS
- SAMPLE: ADDRESS_PERSONS
- SAMPLE: SIMILAR_ORGANIZATION
- SAMPLE: SIMILAR_PERSON

You must run the DQM Compile All Rules Program before you can use these seeded match rules. For more information about this program, see DQM Compile All Rules Program on page 3-15.

For more information about match rules and batch duplicate identification, see Match Rules Overview on page 3-2 and Batch Duplicate Identification Overview on page 5-2.

Other Oracle applications that implement DQM can use the same six match rules for identifying duplicates of entered or updated party information as well as the SAMPLE: SEARCH match rule for search. Only other Oracle applications that implement DQM can use match rules with the Search purpose.

SAMPLE: IDENTICAL_PERSON

The SAMPLE: IDENTICAL_PERSON match rule identifies duplicate parties of type Person.

Acquisition

This table shows the seeded attributes and transformation functions for the acquisition part of the matching process.

Attribute Name	Entity	Filter	Attribute Match	Transformation Name
Party Type	Party	Yes	Match All Attributes	EXACT
Tax Reference	Party	No	Match All Attributes	EXACT
Phone Number Flexible Format	Contact Point	No	Match All Attributes	EXACT
Email Address	Contact Point	No	Match All Attributes	CLEANSE (EMAIL)

Scoring

This table shows the seeded thresholds for the scoring part of the matching process.

Threshold	Value
Match Threshold	0
Override Threshold	
Automatic Merge Threshold	

This table shows the seeded attributes and transformation functions for the scoring part of the matching process.

Attribute Name	Entity	Score	Transformation Name	Weight (%)	Type	Similarity (%)
Tax Reference	Party	20	EXACT	100	Exact	
Phone Number Flexible Format	Contact Point	20	EXACT	100	Exact	
Email Address	Contact Point	20	EXACT (EMAIL)	100	Exact	
			CLEANSE (EMAIL)	70	Exact	

SAMPLE: IDENTICAL_ORGANIZATIONS

The SAMPLE: IDENTICAL_ORGANIZATION match rule identifies duplicate parties of type Organization.

Acquisition

This table shows the seeded attributes and transformation functions for the acquisition part of the matching process.

Attribute Name	Entity	Filter	Attribute Match	Transformation Name
Tax Reference	Party	No	Match All Attributes	EXACT
DUNS Number	Party	No	Match All Attributes	EXACT
Party Type	Party	Yes	Match All Attributes	EXACT

Scoring

This table shows the seeded thresholds for the scoring part of the matching process.

Threshold	Value
Match Threshold	0
Override Threshold	
Automatic Merge Threshold	

This table shows the seeded attributes and transformation functions for the scoring part of the matching process.

Attribute Name	Entity	Score	Transformation Name	Weight (%)	Type	Similarity (%)
Tax Reference	Party	20	EXACT	100	Exact	
DUNS Number	Party	20	EXACT	100	Exact	

SAMPLE: ADDRESS_ORGANIZATIONS

The SAMPLE: ADDRESS_ORGANIZATIONS match rule identifies duplicate parties of type Organization based on party name and address.

Acquisition

This table shows the seeded attributes and transformation functions for the acquisition part of the matching process.

Attribute Name	Entity	Filter	Attribute Match	Transformation Name
Party Name	Party	No	Match All Attributes	WR NAMES + CLEANSE
				WR NAMES + CLUSTER
				REV + WR NAMES + CLEANSE
Postal Code	Address	No	Match All Attributes	EXACT
Party Type	Party	Yes	Match All Attributes	EXACT

Scoring

This table shows the seeded thresholds for the scoring part of the matching process.

Threshold	Value
Match Threshold	55
Override Threshold	
Automatic Merge Threshold	

This table shows the seeded attributes and transformation functions for the scoring part of the matching process.

Attribute Name	Entity	Score	Transformation Name	Weight (%)	Type	Similarity (%)
Party Name	Party	50	EXACT STRING	100	Exact	
			WR NAMES	90	Exact	
			WR NAMES + CLEANSE	70	Similarity	75
Address	Address	50	WR ADDRESS	100	Exact	
			WR ADDRESS + CLEANSE	70	Exact	
Contact Name	Contact	50	WR PERSON	100	Exact	
			WR PERSON + CLEANSE	70	Exact	
DUNS Number	Party	70	EXACT	100	Exact	
Tax Reference	Party	70	EXACT	100	Exact	

SAMPLE: ADDRESS_PERSONS

The SAMPLE: ADDRESS_PERSONS match rule identifies duplicate parties of type Person based on party name and address.

Acquisition

This table shows the seeded attributes and transformation functions for the acquisition part of the matching process.

Attribute Name	Entity	Filter	Attribute Match	Transformation Name
Party Name	Party	No	Match All Attributes	WR NAMES + CLEANSE
				WR NAMES + CLUSTER
				REV + WR NAMES + CLEANSE
Postal Code	Address	No	Match All Attributes	EXACT
Party Type	Party	Yes	Match All Attributes	EXACT

Scoring

This table shows the seeded thresholds for the scoring part of the matching process.

Threshold	Value
Match Threshold	55
Override Threshold	
Automatic Merge Threshold	

This table shows the seeded attributes and transformation functions for the scoring part of the matching process.

Attribute Name	Entity	Score	Transformation Name	Weight (%)	Type	Similarity (%)
Party Name	Party	50	EXACT STRING	100	Exact	
			WR NAMES	90	Exact	
			WR NAMES + CLEANSE	70	Exact	
Email Address	Contact Point	50	EXACT (EMAIL)	100	Exact	
			CLEANSE (EMAIL)	70	Exact	
Phone Number Flexible Format	Contact Point	50	EXACT	100	Exact	
Address	Address	50	WR ADDRESS + CLEANSE	70	Exact	
			WR ADDRESS	100	Exact	

SAMPLE: SIMILAR_ORGANIZATION

The SAMPLE: SIMILAR_ORGANIZATION match rule identifies duplicate parties of type Organization that have similar names, addresses, contacts, or contact points.

Acquisition

This table shows the seeded attributes and transformation functions for the acquisition part of the matching process.

Attribute Name	Entity	Filter	Attribute Match	Transformation Name
Party Name	Party	No	Match Any Attributes	WR NAMES + CLEANSE
				WR NAMES + CLUSTER
				REV + WR NAMES + CLEANSE
				REV + WR NAMES + CLUSTER
Address	Address	No	Match Any Attribute	WR ADDRESS + CLEANSE
Postal Code	Address	No	Match Any Attributes	EXACT
DUNS Number	Party	No	Match Any Attributes	EXACT
Tax Reference	Party	No	Match Any Attributes	EXACT

Scoring

This table shows the seeded thresholds for the scoring part of the matching process.

Threshold	Value
Match Threshold	55
Override Threshold	
Automatic Merge Threshold	

This table shows the seeded attributes and transformation functions for the scoring part of the matching process.

Attribute Name	Entity	Score	Transformation Name	Weight (%)	Type	Similarity (%)
Party Name	Party	50	EXACT STRING	100	Exact	
			WR NAMES	90	Exact	
			WR NAMES + CLEANSE	70	Exact	
			REV + WR NAMES + CLUSTER	70	Exact	
Address	Address	30	EXACT	100	Exact	
			WR ADDRESS + CLEANSE	80	Exact	
Contact Name	Contact	30	WR PERSON	90	Exact	
			WR PERSON + CLEANSE	70	Exact	
			REV + WR PER + CLEANSE	70	Exact	
			EXACT STRING	100	Exact	
DUNS Number	Party	70	Exact	100	Exact	
Tax Reference	Party	70	Exact	100	Exact	
Phone Number Flexible Format	Contact Point	30	Exact	100	Exact	
SIC Code	Party	20	Exact	100	Exact	
Postal Code	Address	20	Exact	100	Exact	

SAMPLE: SIMILAR_PERSON

The SAMPLE: SIMILAR_PERSON match rule identifies duplicate parties of type Person that have similar names, address, or contact points.

Acquisition

This table shows the seeded attributes and transformation functions for the acquisition part of the matching process.

Attribute Name	Entity	Filter	Attribute Match	Transformation Name
Party Name	Party	No	Match All Attributes	WR NAMES + CLEANSE
				WR NAMES + CLUSTER
				REV + WR NAMES + CLEANSE
				REV + WR NAMES + CLUSTER
Address	Address	No	Match All Attribute	WR ADDRESS + CLEANSE
Postal Code	Address	No	Match All Attributes	EXACT
Email Address	Contact Point	No	Match All Attributes	CLEANSE (EMAIL)
Tax Reference	Party	No	Match All Attributes	EXACT

Scoring

This table shows the seeded thresholds for the scoring part of the matching process.

Threshold	Value
Match Threshold	55
Override Threshold	
Automatic Merge Threshold	

This table shows the seeded attributes and transformation functions for the scoring part of the matching process.

Attribute Name	Entity	Score	Transformation Name	Weight (%)	Type	Similarity (%)
Party Name	Party	50	EXACT STRING	100	Exact	
			WR NAMES	90	Exact	
			REVERSE	80	Exact	
Email Address	Contact Point	50	EXACT (EMAIL)	100	Exact	
			CLEANSE (EMAIL)	70	Exact	
Phone Number Flexible Format	Contact Point	50	EXACT	100	Exact	
Address	Address	30	WR ADDRESS + CLEANSE	70	Exact	
			WR ADDRESS	100	Exact	
Postal Code	Address	20	Exact	100	Exact	
Tax Reference	Party	70	Exact	100	Exact	

SAMPLE: SEARCH

The SAMPLE: SEARCH match rule is for other Oracle applications that implement DQM to create search interface based on commonly used attributes.

Acquisition

This table shows the seeded attributes and transformation functions for the acquisition part of the matching process.

Attribute Name	Entity	Filter	Attribute Match	Transformation Name
Party Name	Party	No	Match Any Attributes	WR NAMES + CLEANSE
				WR NAMES + CLUSTER
				REV + WR NAMES + CLEANSE
Contact Name	Contact	No	Match Any Attribute	WR PERSON + CLEANSE
				WR PERSON + CLUSTER
				REV + WR PER + CLEANSE
Address	Address	No	Match Any Attribute	WR ADDRESS + CLEANSE
City	Address	No	Match Any Attribute	CLEANSE
State	Address	No	Match Any Attributes	WR STATE + CLEANSE
Phone Number Flexible Format	Contact Point	No	Match Any Attribute	EXACT
Email Address	Contact Point	No	Match Any Attributes	CLEANSE (EMAIL)
Concatenated Party Names	Party	No	Match Any Attributes	WR NAMES + CLEANSE

Scoring

This table shows the seeded thresholds for the scoring part of the matching process.

Threshold	Value
Match Threshold	300
Override Threshold	
Automatic Merge Threshold	

This table shows the seeded attributes and transformation functions for the scoring part of the matching process.

Attribute Name	Entity	Score	Transformation Name	Weight (%)	Type	Similarity (%)
Party Name	Party	50	EXACT STRING	100	Exact	
			WR NAMES	90	Exact	
			WR NAMES + CLEANSE	70	Exact	
			WR NAMES + CLUSTER	50	Similarity	80
Contact Name	Contact	50	WR PERSON	100	Exact	
			WR PERSON + CLEANSE	70	Exact	
			REVERSE	90	Exact	
			REV + WR PER + CLEANSE	60	Exact	
Address	Address	50	WR ADDRESS	100	Exact	
			WR ADDRESS + CLEANSE	70	Exact	
City	Address	30	EXACT	100	Exact	
State	Address	20	WR STATE	100	Exact	
Phone Number Flexible Format	Contact Point	80	EXACT	100	Exact	
Email Address	Contact Point	80	EXACT (EMAIL)	100	Exact	

SAMPLE: SEARCH

Attribute Name	Entity	Score	Transformation Name	Weight (%)	Type	Similarity (%)
Concatenated Party Names	Party	40	WR NAMES	100	Exact	
			WR NAMES + CLEANSE	70	Exact	

Standard Navigation Paths

This appendix describes how to navigate to each window in Oracle Trading Community Architecture Data Quality Management.

Standard Navigation Paths

Although your system administrator may have customized your navigator, typical navigation paths are shown in this table. Access all of these windows through the Trading Community Manager responsibility.

Window Name	Navigation Path
Attributes and Transformation Functions	Data Quality Management > Setup > Attributes and Transformation Functions
Copy Match Rule	Data Quality Management > Setup > Match Rules > Copy
Duplicate Identification: Batch Review	Data Quality Management > Duplicate Identification > Batch Review
Match Details	Data Quality Management > Duplicate Identification > Batch Review > View Match Details
Match Rules	Data Quality Management > Setup > Match Rules
Merge Batch	Data Quality Management > Merge Parties Data Quality Management > Duplicate Identification > Batch Review > Create Merge Set > Go to Party Merge Data Quality Management > Duplicate Identification > Batch Review > Create Merge Set > Submit Party Merge Batch
Review Party Merge Batches	Data Quality Management > Duplicate Identification > Batch Review > Create Merge Set Data Quality Management > Duplicate Identification > Batch Review > View Merge Set
Submit Duplicate Identification Batch	Data Quality Management > Duplicate Identification > Batch Definition
Word Replacements	Data Quality Management > Setup > Word Replacements (DQM)

Profile Options

This appendix lists the profile options that affect the operation of Oracle Trading Community Architecture Data Quality Management. This appendix includes a brief description of each profile option that you or your system administrator can set at the site, application, responsibility, or user levels.

Profile Options

During implementation, your system administrator sets a value for each user profile option to specify how Data Quality Management controls access to and processes data.

See also: Overview of Setting User Profiles, *Oracle Applications System Administrator's Guide*

Profile Options Summary

This table indicates whether you can view or update profile options and at which levels your system administrator can update these profile options: the user, responsibility, application, or site levels.

A *Required* profile option requires you to provide a value. An *Optional* profile option already provides a default value which you can change.

The key for this table is:

- **Update** - You can update the profile option.
- **View Only** - You can view the profile option but cannot change it.
- **No Access** - You cannot view or change the profile option value.

Profile Options	Value	Default	User Access	System Administrator Access			
				User	Responsibility	Application	Site
DQM Match Rule ID for Batch Duplicate Identification	Required	No Default	Update	Update	Update	Update	Update
HZ: DQM Index Creation Memory	Required	No Default	Update	Update	Update	Update	Update

Profile Options in Data Quality Management

This section lists the profile options in Data Quality Management.

DQM Match Rule ID for Batch Duplicate Identification

Use the DQM Match Rule ID for Batch Duplicate Identification profile option to store the default match rule to use for identifying duplicates that currently exist in the TCA registry. This match rule defaults in the Match Rule field of the Submit Duplicate Identification Batch window.

HZ: DQM Index Creation Memory

Use the HZ: DQM Index Creation Memory profile option to override the default amount of memory for the DQM Index Creation Program to use to generate *interMedia* indexes. Enter the amount of memory in megabytes.

DQM Match Rule ID for Online Duplicate Identification

The DQM Match Rule ID for Online Duplicate Identification profile option is used only by other Oracle applications that implement DQM to store the default match rule to use for finding and duplicates of entered or updated party information and preventing duplicates from entering the registry.

DQM Match Rule ID for Search UI

The DQM Match Rule ID for Search UI profile option is used only by other Oracle applications that implement DQM to store the default match rule to use for matching search criteria.

HZ: DQM Index Creation Workers

This profile option is not currently used.

HZ: Duplicate Allowed

This profile option is not currently used.

Glossary

Acquisition

Acquisition is the part of the 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.

Automatic Merge Threshold

See *Threshold*.

Compile

To generate the PL/SQL code for the match rule. This process also saves and locks the match rule.

Exact Search

An exact search requires an exact character match, as compared to a search that uses the similarity algorithm.

Fuzzy Match

Fuzzy match finds data that has only some similarity to the search criteria and compensates for errors in data entry and phonetics.

Input Record

The input record is used as the basis for comparison or matching. Input records can be compared to other records in the staged schema to find existing duplicates in the TCA registry. An input record can also be search criteria or entered party information that other Oracle applications which implement DQM compare to the staged schema to find matches or potential duplicates in the registry.

***interMedia* Index**

Oracle *interMedia* lets you perform a variety of text data searches in over 39 languages. For 16 of those languages, *interMedia* text search also provides: base letter conversion, alternate spelling, fuzzy matching, language specific lexing, sample stop list, and stemming.

In DQM, *interMedia* indexes are generated for the staged schema to use in the acquisition phase of matching. These indexes quickly narrow down the number of records to evaluate for scoring. Searches against the staged schema are actually done through the indexes.

Logical Entity

A logical entity is group of related attributes. The four logical entities in DQM are party, address, contact, and contact point.

Match Rule

A match rule is 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.

Match Threshold

See *Threshold*.

Organization Person

An organization person is one who acts on behalf of or in the context of an organization.

Override Threshold

See *Threshold*.

Party

A party is a person, organization, or collection of parties that can enter into relationships with other parties.

Score

Scores are assigned in integer form to attributes in the Match Rules window. Scores are not limited to the range of 1 to 100 and are multiplied by the weight percentage to calculate the match score.

Scoring

Scoring is the process of assigning weighting factors to match results and grouping and ranking the match results.

Similarity Algorithm

The similarity algorithm computes the edit distance between two strings, or groups of text, and assigns a percentage value to the result. The calculation used to determine the similarity percentage is:

1. Determine the edit distance, or the number of changes required to make the longer string match the shorter string.

For example, for Smythe and Smith, the edit distance is two.

2. Subtract the edit distance from the number of characters in the longest string.

Following the example above: $6 - 2 = 4$.

3. Divide the amount calculated in step 2 by the number of characters in the longest string.

Continuing the example: $4/6 = 0.6666$

4. Express the result as an integer.

In this example the result would be a similarity score of 67.

If two strings are identical, then the similarity percentage equals 100. If no characters in the two strings are the same, then the similarity percentage is zero.

Soundex

Soundex is an indexing system that can group words by sound instead of exact spelling. Soundex translates words into a four-digit code consisting of one letter and three numbers.

Staged Schema

The staged schema is a mirror of the TCA registry but contains data from the registry entities that is transformed and standardized based on transformation functions for better matching. Data in the staged schema is extracted into an *interMedia* index for high performance searching.

Standardize

Standardizing is a data manipulation technique that improves the quality of matching. Standardizing in Data Quality Management involves transforming words or phrases into the same format.

TCA Registry

The TCA registry is the central repository of party information for all Oracle applications. The party information includes details about organizations and people, the relationships among the parties, and the places where the parties do business.

Threshold

Automatic merge, match, and override thresholds are values that are used in matching. A record with a score that exceeds the match threshold is selected as a match for the input record. A record with a score that exceeds the automatic merge threshold by default selected for party merge. Other Oracle applications that implement DQM can prevent the input record from entering the TCA registry if a duplicate is found that exceeds the override threshold.

Trading Community Architecture (TCA)

Oracle Trading Community Architecture is a model that provides a virtual representation of the community that business is conducted in. This model includes parties and related party entities.

Transform

To apply a transformation function to the value of an attribute to standardize the value for matching purposes.

Transformation Function

Transformation functions are seeded or user-defined rules that transform and standardize TCA attribute values into representations that can assist in the identification of potential matches.

Uncompiled

A possible match rule status. The others are New and Compiled. An uncompiled match rule does not have any associated PL/SQL code. You cannot use or copy an uncompiled match rule.

Weight

A percentage that is assigned to a transformation function and is used to factor the score of an attribute.

Word Replacement

A word replacement is a word mapping that is used to create synonyms which are treated as equivalents for searching and matching. Base words in the party, account, address, or organization contact record are replaced with a normalized word during search to provide consistency in search results.

Work Unit

A work unit consists of all records from the staged schema with attribute values that match the attribute values of the input record.

Index

A

Acquisition

- defining in a match rule, 3-6
- definition, Glossary-1
- in match rules, 3-2
- prerequisite, 3-7
- tabbed region, 3-6
- without match rule scoring, 3-7
- work unit, 3-7

Attributes

- creating custom, 2-8, 2-9
- custom procedures, 2-6
- defining, 2-5, 2-6
- definition, Glossary-1
- description, 1-3
- overview, 2-2
- seeded, 2-2, A-2
- seeded for address logical entity, A-11
- seeded for contact logical entity, A-13
- seeded for contact point logical entity, A-15
- seeded for party logical entity, A-3
- tips, 2-13

Attributes and Transformation Functions window

- creating a custom attribute, 2-9
- creating custom attributes, 2-8
- creating custom transformation functions, 2-11
- standard navigation path, B-2
- transformations region, 2-4
- using, 2-5, 2-6

Automatic merge threshold

- definition, Glossary-1
- entering its value, 3-13
- in the matching process, 3-4

B

Batch duplicate identification

- defining and submitting duplicate identification batches, 5-5
- overview, 5-2
- prerequisites, 5-4
- process, 5-2
- reviewing duplicates and creating merge batches, 5-8
- seeded match rules, A-25
- submitting merge batches, 5-13

C

Compile

- button, 3-9, 3-23, 3-25
- copying match rule, 3-22
- definition, Glossary-1
- locking feature, 3-9
- status definition, 3-5
- using the Unlock button to uncompile, 3-24

Copy Match Rule window

- copying match rules, 3-22
- standard navigation path, B-2

D

Data Quality Management

See DQM

DQM

- batch duplicate identification, 5-1
- interMedia* indexes, 4-1
- major features, 1-3

- matching process, 3-3
- overview, 1-2
- process, 1-9
- profile options, C-1
- seed data, A-1
- setting up and using, 1-11
- staged schema, 4-1
- standard navigation paths, B-1
- DQM Batch Duplicate Identification Program
 - description, 5-5
 - reviewing results, 5-8
 - using match rules, 3-2
- DQM Compile All Rules Program
 - description, 3-15
- DQM Index Creation Program
 - description, 4-4
- DQM Index Optimization Program
 - description, 4-6
 - prerequisites, 4-6
- DQM Match Rule ID for Batch Duplicate Identification profile option
 - description, C-3
- DQM Match Rule ID for Online Duplicate Identification profile option
 - description, C-3
- DQM Match Rule ID for Search UI profile option
 - description, C-3
- DQM Staging Program
 - description, 4-4
 - parameters, 4-5
 - prerequisites, 4-5
 - tip, 2-13
 - updating transformation functions, 2-4, 2-13
 - updating word replacement list, 2-14
 - updating word replacement lists, 2-17
- DQM Synchronization Program
 - description, 4-7
 - prerequisites, 4-7
- Duplicate identification batches
 - defining and submitting, 5-5
 - reviewing potential duplicates, 5-8
- Duplicate Identification: Batch Review window
 - reviewing duplicates and creating merge batches, 5-8
 - standard navigation path, B-2

- Duplicate parties
 - example of identifying, 1-6
 - example of preventing, 1-8
 - identifying in TCA registry, 1-5

E

- Exact match, 3-13
- Exact search
 - definition, Glossary-1

F

- Fuzzy match
 - definition, Glossary-1

H

- HZ: DQM Index Creation Memory profile
 - option, C-3
- HZ: DQM Index Creation Workers profile
 - option, C-3
- HZ: Duplicate Allowed profile option, C-3

I

- Input record
 - definition, Glossary-2
- interMedia* indexes
 - creating and updating, 4-4
 - definition, Glossary-2
 - description, 1-5
 - optimizing, 4-6
 - overview, 4-3
 - synchronizing with TCA registry, 4-7

L

- Logical attributes
 - description, A-2
- Logical entities
 - definition, Glossary-2
 - description, 1-3

M

Match Details window

- reviewing reasons for duplicate party match, 5-8
- standard navigation path, B-2

Match rules

- compile, 3-14
- compiled, 3-5
- defining acquisition component, 3-6
- defining scoring component, 3-10
- definition, Glossary-2
- description, 1-4
- matching process, 3-3
- overview, 3-2
- prerequisite, 3-11
- seeded, A-25
- setting up, 3-5
- steps to define scoring component, 3-12
- transformation functions, 2-3
- uncompiled., 3-5

Match Rules window

- Acquisition tabbed region, 3-6
- standard navigation path, B-2

Match threshold

- definition, Glossary-2

Matching

- exact, 3-13
- similarity, 3-13

Merge Batch window

- standard navigation path, B-2

Merge batches

- creating, 5-8
- submitting to Party Merge, 5-13

Merge-from parties

- description, 5-8

Merge-to parties

- description, 5-8

O

Online duplicate identification

- seeded match rules, A-25

Oracle Trading Community Architecture

- See TCA

Organization person

- definition, Glossary-2

Override threshold

- definition, Glossary-2

P

Parties

- definition, Glossary-3
- description, 1-2
- example of identifying duplicates, 1-6
- example of preventing duplicates, 1-8
- example of searching for, 1-6
- identifying duplicates in TCA registry, 5-2
- Party Merge, 5-13
- TCA registry, 1-2

Party Merge

- description, 5-13

Profile options

- DQM Match Rule ID for Batch Duplicate Identification profile option, C-3
- DQM Match Rule ID for Online Duplicate Identification profile option, C-3
- DQM Match Rule ID for Search UI profile option, C-3
- HZ: DQM Index Creation Memory profile option, C-3
- HZ: DQM Index Creation Workers, C-3
- HZ: Duplicate Allowed, C-3
- overview, C-2
- settings, C-2

Programs

- DQM Batch Duplicate Identification Program, 5-5
- DQM Compile All Rules Program, 3-15
- DQM Index Creation Program, 4-4
- DQM Index Optimization Program, 4-6
- DQM Staging Program, 4-4
- DQM Synchronization Program, 4-7

R

Review Party Merge Batches window

- standard navigation path, B-2
- submitting merge batches, 5-13

S

- Scores
 - definition, Glossary-3
- Scoring
 - definition, Glossary-3
- Search for parties
 - example of, 1-6
- Seed data
 - attributes, A-2
 - match rules, A-25
 - transformation functions, A-17
- Similarity algorithm
 - definition, Glossary-3
- Soundex
 - definition, Glossary-3
- srs.arhdqmstw, 4-4
- Staged schema
 - attributes, 2-3
 - creating and updating, 4-4
 - definition, Glossary-4
 - description, 1-5
 - installing seeded transformation functions, 2-4
 - mapping to TCA registry tables, 4-2
 - overview, 4-2
 - synchronizing with TCA registry, 4-7
- Standard navigation paths, B-2
- Standard Request Submission windows
 - DQM Compile All Rules Program, 3-15
 - DQM Index Optimization Program, 4-6
 - DQM Staging Program, 4-4
 - DQM Synchronization Program, 4-7
- Standardize
 - definition, Glossary-4
- Submit Duplicate Identification Batch window
 - defining and submitting duplicate identification batches, 5-5
 - standard navigation path, B-2

T

- TCA
 - definition, Glossary-4
 - description, 1-2
- TCA registry

- definition, Glossary-4
 - description, 1-2
 - identifying duplicate parties within, 5-2
- Thresholds
- automatic merge, 3-13
 - definition, Glossary-4
 - match, 3-13
- Trading Community Architecture
- See TCA
- Transform
- definition, Glossary-4
- Transformation functions, 2-3
- creating custom, 2-11
 - deactivating, 2-5
 - defining, 2-5, 2-6
 - definition, Glossary-4
 - description, 1-4
 - installing in staged schema, 2-4
 - logical entities, 2-5
 - lookups, 2-13
 - seeded, 2-3, 2-5, A-17
 - tips, 2-13

U

- Uncompiled
 - definition, Glossary-5

W

- Weight
 - definition, Glossary-5
- Windows
 - Copy Match Rule window, 3-22
 - Duplicate Identification: Batch Review window, 5-8
 - Match Details window, 5-8
 - Review Party Merge Batches window, 5-13
 - standard navigation paths, B-2
 - Standard Request Submission windows, 3-15, 4-4, 4-6, 4-7
 - Submit Duplicate Identification Batch window, 5-5
- Word replacements
 - creating a list of word pairs, 2-16

- defining a list, 2-17
- definition, Glossary-5
- description, 1-4
- overview, 2-14
- seeded pairs, 2-15
- tips, 2-18

Word Replacements window

- standard navigation path, B-2

Work units

- definition, Glossary-5

Workers

- description, 4-4

