# Oracle Financial Services Transaction Filtering

## Matching Guide

**Release 8.1.2.2.0**

**October 2022**

**E96622-05**

ORACLE
Financial Services

ORACLE

OFS Transaction Filtering Matching Guide

# Document Control

Table 1 lists the document control of this guide:

**Table 1: Document Control**

| Revision Number | Revision Date | Description |
|---|---|---|
| 8.1.2.2.0 | October 2022 | No content updates for this release. |
| 8.1.2.0.0 | July 2022 | The first publication of this guide. |

## Content


**1 About This Guide ..... 4**

1.1 Who Should Use this Guide ..... 4

1.2 How this Guide is Organized ..... 4

1.3 Where to Find More Information ..... 4

1.4 Conventions Used in this Guide ..... 5

**2 About Transaction Filtering ..... 6**

2.1 Overview of Transaction Filtering ..... 6

**3 Introduction to Matching ..... 7**

3.1 General Matching Strategy ..... 7

*3.1.1 Identifier Preparation ..... 7*

*3.1.2 Clustering ..... 7*

*3.1.3 Matching ..... 7*

3.2 Configuring Oracle Financial Services TF for different scenarios ..... 8

**4 Matching Webservices ..... 9**

4.1 Overview ..... 9

4.2 Name and Address Matching ..... 10

4.3 Country and City Matching ..... 10

4.4 Identifier Matching ..... 11

4.5 Narrative Matching ..... 11

4.6 Trade Goods Matching ..... 11

4.7 Trade Port Matching ..... 11

**5 SWIFT Message Tags Screening ..... 12**

5.1 Name and Address Screening ..... 12

5.2 Country and City Screening ..... 12

5.3 Identifier Screening ..... 13

5.4 Narrative Screening ..... 14

**6 Name and Address Matching ..... 15**

6.1 Identifier preparation ..... 15

*6.1.1 Name Normalization ..... 16*

*6.1.2 City and country identifiers ..... 16*

6.2 Clustering ..... 17
*6.2.1 Family Name Cluster (dnClusterFamilyName)* ..... *17*
*6.2.2 Individual Full Name Metaphone Pairs Cluster (dnClusterFullNameMeta)* ..... *19*
*6.2.3 Individual Given Names Cluster (dnClusterGivenNames)* ..... *21*
*6.2.4 Individual Full Name Trim Pairs Cluster (dnClusterFullNameTrim)* ..... *22*
*6.2.5 Individual Initials (dnClusterInitials)* ..... *25*
*6.2.6 Original Script Name (dnClusterOriginalScript)* ..... *26*
*6.2.7 Entity Name Tokens (dnClusterNameTokens)* ..... *26*
*6.2.8 Entity Name Meta (dnClusterLongName)* ..... *27*
*6.2.9 Entity Name Trim (dnClusterShortName)* ..... *27*
*6.2.10 Entity Start End Name Tokens (dnClusterStartEndNameTokens)* ..... *28*
6.3 Matching ..... 28
*6.3.1 Match Rules* ..... *29*
*6.3.2 Individual Name Matching Rules* ..... *31*
*6.3.3 Loose Individual Name Matching Rules* ..... *41*
*6.3.4 Aircraft Matching Rules* ..... *43*
*6.3.5 Vessel Matching Rules* ..... *44*
*6.3.6 Entity Matching Rules* ..... *45*
**7 Identifier Matching ..... 73**
7.1 Identifier ..... 73
7.2 Cluster ..... 73
7.3 Match Rules ..... 73
**8 Country and City Matching ..... 76**
8.1 Identifiers ..... 76
8.2 Clusters ..... 76
8.3 Match Rules ..... 77
**9 Narrative Matching ..... 79**
9.1 Identifiers ..... 79
9.2 Clusters ..... 80
9.3 Match Rules ..... 80
9.4 Enabling a Match Rule and Web Service ..... 83

**10 Trade Goods Matching ..... 90**
10.1 Identifiers ..... 90
10.2 Clusters ..... 90
10.3 Match Rules ..... 90
**11 Trade Port Matching ..... 94**
11.1 Identifiers ..... 94
11.2 Clusters ..... 94
11.3 Match Rules ..... 95
**12 OFSAA Support Contact Details ..... 97**
**13 Send Us Your Comments ..... 98**

# 1 About This Guide

This guide provides information on webservices matching, individual matching, vessel matching, aircraft matching, and entity matching

This chapter focuses on the following topics:

- Who Should Use this Guide
- How this Guide is Organized
- Where to Find More Information
- Conventions Used in this Guide

## 1.1 Who Should Use this Guide

The *Transaction Filtering Matching Guide* is designed for the following users:

- **Analyst**: This user works on the transactions within the application frequently. This user's specific role determines what they can view and perform within the application
- **Supervisor**: This user works on the transactions within the application on a daily basis and is typically a higher level Analyst or Compliance Officer
- **Admin**: This user performs the activities related to administration.

## 1.2 How this Guide is Organized

The *Transaction Filtering Matching Guide* includes the following chapters:

- Chapter 2, About Transaction Filtering, provides an overview of Oracle Financial Services Transaction Filtering, how it works, and what it does.
- Chapter 3, Introduction to Matching, provides a brief introduction to Matching and how it is used in Transaction Filtering .
- Chapter 4, Matching Webservices, explains how to match webservices such as name and address, country and city, Identifier, and narrative fields.
- Chapter 5, SWIFT Message Tags Screening, explains how to match SWIFT message tags for different SWIFT message types.
- Chapter 6, Name and Address Matching, explains how to match individuals to different watchlists.
- Chapter 6, Identifier Matching, explains how to match vessels to different watchlists.
- Chapter 7, Country and City Matching, explains how to match aircrafts to different watchlists.
- Chapter 9, Narrative Matching, explains how to match entities to different watchlists.
- Chapter 10, Trade Goods Matching,This chapter discusses This matching webservice is used to extract the trade goods name and match the name against the prohibited goods list.
- Chapter 11, Trade Port Matching, This chapter discusses the matching webservice that is used to extract the trade port name and match it to the country while screening.

## 1.3 Where to Find More Information

For more information about Oracle Financial Services Transaction Filtering, refer to the following documents:

- Oracle Financial Services Sanctions Installation Guide
- Oracle Financial Services Transaction Filtering Administration Guide
- Oracle Financial Services Transaction Filtering User Guide
- Oracle Financial Services Transaction Filtering Reporting Guide
- Oracle Financial Services Transaction Filtering Release Notes
- Oracle Financial Services Technical Integration Guide

These documents are available at the following link:

http://docs.oracle.com/cd/E60570_01/homepage.htm

To find more information about Oracle Financial Services Transaction Filtering and our complete product line, visit our Web site www.oracle.com/financialservices.

## 1.4 Conventions Used in this Guide

Table 1 provides the conventions used in this guide.

**Table 1: Conventions Used in this Guide**

| Conventions | Description |
|---|---|
| *Italics* | • Names of books, chapters, and sections as references<br>• Emphasis |
| **Bold** | • The object of an action (menu names, field names, options, button names) in a step-by-step procedure<br>• Commands typed at a prompt<br>• User input |
| `Monospace` | • Directories and subdirectories<br>• File names and extensions<br>• Process names<br>• Code sample, including keywords and variables within the text and as separate paragraphs, and user-defined program elements within the text. |
| Asterisk | Mandatory fields in User Interface |
| `<Variable>` | Substitute input value |

# 2 About Transaction Filtering

This chapter gives an overview of Transaction Filtering. For any information related to features, workflow, matching logic, SWIFT message formats, and user roles and actions, see Oracle Financial Services Transaction Filtering User Guide.

## 2.1 Overview of Transaction Filtering

Oracle Financial Services Transaction Filtering is a real-time blacklist filtering system that identifies blacklisted, restricted and sanctioned individuals, entities, cities and countries in a financial transaction processed through the Solution. The solution can interface with any clearing systems, payment system or any source system. The solution accepts messages from the source systems in real time and scans them against different watch lists maintained within the system to identify existence of any blacklisted data present within the message.

# 3 Introduction to Matching

Oracle Financial Services Transaction Filtering provides a flexible and customizable strategy for matching customer records to watch list records. **Sanctions screening** typically requires the business to employ tightly-defined, zero tolerance matching policies which will identify every possible match against a sanctions list.

Oracle Financial Services Transaction Filtering therefore employs a range of clustering strategies and matching rules. These can be enabled and disabled as needed, to tune the behavior of Oracle Financial Services Transaction Filtering to your requirements.

In general, the looser the match rule, the more likely it is to raise false positives. It is not possible to eliminate all false positives, especially if there is a requirement to identify **all** true matches. Tuning the matching strategy is therefore a trade-off between the proportion of true matches which are not detected and the work required to manually eliminate false positives. This will be evident in the examples in this document.

## 3.1 General Matching Strategy

This section provides a brief description of the general strategy used in Oracle Financial Services Transaction Filtering. It consists of three main components: identifier preparation, clustering and matching.

### 3.1.1 Identifier Preparation

There are some differences between the structure of data sets that always need to be normalized before clustering and matching, so that the matching process does not need to repeat the configuration of transformations on each comparison.

Identifier preparation is used to ensure that the records conform to a pre-defined data structure which can be used by the rest of the matching process, and also to eliminate common forms of variance between the records (such as spelling variants of given names and abbreviations of frequently-used tokens).

### 3.1.2 Clustering

Clustering is used to minimize the work that must be performed by the final stage of matching. It works by splitting the working and reference data into wide tranches (clusters), based on similarities in significant data fields. Only subsets of the data which share similar characteristics, and will therefore be placed in the same cluster, will be compared on a record-by-record basis later in the matching process.

If very wide clusters are used, there will be a large number of records in each cluster. This means that there is a reduced risk that true matches will be missed, but also that a greater amount of processing power is required to compare all the clustered records by brute force. A tighter clustering strategy will result in smaller clusters, with fewer records per cluster. This results in reduced processing requirements for row-by-row comparisons, but increases the likelihood that some true matches will not be detected.

### 3.1.3 Matching

Once the working and watch list records have been divided into clusters, the rows within each cluster are compared to one another according to the match rules defined for the matching processor. Each match rule defines a set of criteria, specified as comparisons, that the pair of records must satisfy in order to qualify as a match under that rule. The rules are applied as a decision table, so if a pair of

records qualifies as a match under a rule higher in the table, it will not be compared using any rules below that. All rules are configured to operate on a case-insensitive basis. Unless stated otherwise, all noise and whitespace characters are removed or normalized before matching.

## 3.2 Configuring Oracle Financial Services TF for different scenarios

As previously mentioned, Oracle Financial Services Transaction Filtering includes clusters and matching rules that are suited to various screening requirements. Tuning Oracle Financial Services Transaction Filtering to match your policies should be undertaken carefully and under the supervision of a risk and compliance expert, with knowledge of your business requirements and the relevant legislation.

The following general points may be useful when tuning the behavior of Oracle Financial Services Transaction Filtering:

- Some organizations use a zero-tolerance policy for individual name matches. Such a policy typically requires that all potential name matches must be manually reviewed, irrespective of the rest of the data associated with the record. If such a policy is in place, you must enable the conflict rules in individual matching
- In general, using a loose clustering strategy will result in relatively few clusters, each containing many records. This reduces the potential for missing true matches, and increases the chance of false positives. It is also expensive in terms of processing requirements, as every record in the cluster must be directly compared with every other record in that cluster.
- Conversely, a tight clustering strategy will result in a relatively large number of clusters, each containing fewer records. This increases the potential for missing true matches, and decreases the chance of false positives. It also reduces the overall cost of the processing requirements, as relatively few direct comparisons are required.

# 4 Matching Webservices

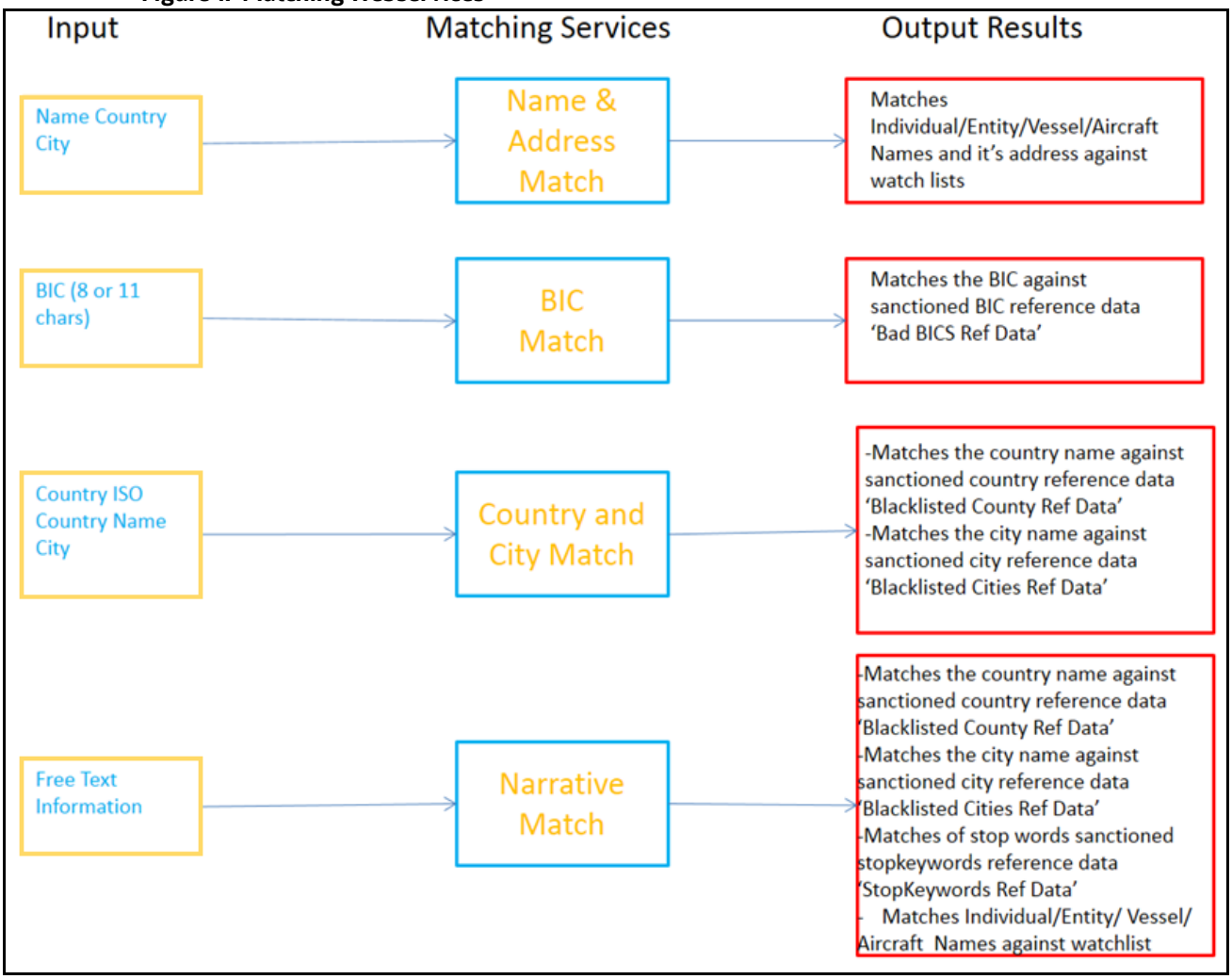
This chapter gives an overview of Matching Webservices for Transaction Filtering. This chapter discusses the following topics:


- Overview
- Name and Address Matching
- Country and City Matching
- Identifier Matching
- Narrative Matching
- Trade Goods Matching
- Trade Port Matching


## 4.1 Overview

This section explains the various Matching Webservices that are used for Transaction Filtering. The following diagram depicts the input and output for these Webservices.

**Figure 1: Matching Webservices**




## 4.2 Name and Address Matching

This matching webservice is used for the party name and address matching, such as orderer and beneficiary. Party can be an individual, entity, vessel, or aircraft. The webservice takes the party name, country (if available in the message) and city (if available in the message) as an input and matches them against the watchlist records.

For information on the reference data for watchlists, see Oracle Financial Services Transaction Filtering Administration Guide.

## 4.3 Country and City Matching

This matching webservice is used for country and city matching against the sanctioned country and city list. The sanctioned country and city reference data names are *Blacklisted Country Ref Data* and *Blacklisted Cities Ref Data* respectiveldentifierly. The blacklisted city information is extracted from the party address field of the input message and the blacklisted country extraction happens from different fields such as *Country from party address, Country ISO code from BIC, Country ISO from IBAN,* and *Country confirmation from party location*.

For information on the reference data for watchlists, see Oracle Financial Services Transaction Filtering Administration Guide.

## 4.4 Identifier Matching

This matching webservice is used for Identifier matching against the sanctioned Identifier list such as Bad BICs Ref Data.This solution supports more legal identifiers matching. For example, LEI, passport, and so on.

For information on the reference data for watchlists, see Oracle Financial Services Transaction Filtering Administration Guide

## 4.5 Narrative Matching

This matching webservice is used for free text or narrative field screening such as remittance information, reference fields, and sender to receiver information. This particular webservice screens individual, entity, vessel and aircarft names (if present) against different records such as watchlist records, country against sanctioned country reference data, city against sanctioned city reference data, BIC against sanctioned BIC reference data and key words against Stopkeywords reference data or StopKeywords Ref Data.

For information on the reference data for watchlists, see Oracle Financial Services Transaction Filtering Administration Guide.

## 4.6 Trade Goods Matching

This matching webservice is used to extract the trade goods name and match it name against the prohibited goods list. This list provides country-wise data.

For information on the reference data for watchlists, see Oracle Financial Services Transaction Filtering Administration Guide.

## 4.7 Trade Port Matching

This matching webservice is used to extract the trade port name and match it to the country while screening. This is done because two countries may not have the same port name, so this avoids duplication. The port name is also matched against the sanctioned port reference data.

For information on the reference data for watchlists, see Oracle Financial Services Transaction Filtering Administration Guide.

# 5 SWIFT Message Tags Screening

This chapter gives an overview of SWIFT tags used for screening under each matching webservices. This section covers four SWIFT message types: MT101, MT103, MT202, MT202COV.

This configuration is only given for reference purposes. You can configure other message types in a similar manner.

This chapter discusses the following topics:

- Name and Address Screening
- Country and City Screening
- Identifier Screening
- Narrative Screening

## 5.1 Name and Address Screening

This section explains the Name and Address SWIFT tags that are screened for various Message Types. Table 2 describes the Name and Address Screening details.

**Table 2: Name and Address Screening**

| Message Type | Tags/ Fields |
|---|---|
| MT101 | 50F, 50H,56D,57D, 59-No letter option, 59F |
| MT103 | 50F, 50K, 53D, 52D, 54D, 55D,56D, ,57D,59-No letter option, 59F |
| MT202 | 52D,53D,54D,56D,57D,58D |
| MT202COV | 52D,53D,54D,56D,57D,58D,50F, 50K,59-No letter option, 59F |

## 5.2 Country and City Screening

This section explains the Country and City SWIFT tags that are screened for various Message Types.

Table 2 describes the Country and City Screening details.

**Table 3: Country and City Screening**

| Message Type | Tags/ Fields | Tags/ Fields | Tags/ Fields |
|---|---|---|---|
| MT101 | Extract ISO country code from BIC present in the following tag:<br>50C (4!a2!a2!c[3!c]), Block1 BIC, Block 2 BIC, 50G,52A,51A,56A,57A, 59A | 50F, 50H,56D,57D, 59-No letter option, 59F | Extract country code from IBAN which might be present in the following tags.<br>50F, 50H, 50G, 59-No letter option, 59F, 59A,,25A |
| MT103 | Extract ISO country code from BIC present in the following tag:<br>Block1 BIC, Block 2 BIC, 50A, 51A , 52A,53A,55A, 54A,56A,57A,59A | 50F, 50K, 53D, 52D, 54D, 55D,56D, ,57D,59-No letter option, 59F, 53B, 54B, 55B,57B | 50A, 50F, 50K, 59-No letter option, 59F, 59A |
| MT202 | Extract ISO country code from BIC present in the following tag:<br>Block1 BIC, Block 2 BIC, 52A,53A, 54A,56A,57A,58A | 52D,53D,54D,56D,57D, 58D, 53B, 54B, 57B | N/A |
| MT202COV | Extract ISO country code from BIC present in the following tag:<br>Block1 BIC, Block 2 BIC, 52A,53A, 54A,56A,57A,58A,50A, 59A | 52D,53D,54D,56D,57D, 58D,50F, 50K,59-No letter option, 59F, 53B, 54B, 57B | 50A, 50F, 50K, 59-No letter option, 59F, 59A |

Block 1 and Block 2 of SWIFT messages are included by default. They can be enabled or disabled based on your requirement. For information on how to enable or disable a tag for screening, see Oracle Financial Services Transaction Filtering Administration Guide.

## 5.3 Identifier Screening

This section explains the BIC SWIFT tags that are screened for various Message Types.

Table 4 describes the BIC Screening details.

**Table 4: BIC Screening**

| Message Type | Tags/ Fields |
|---|---|
| MT101 | 50C (4!a2!a2!c[3!c]), Block1 BIC, Block 2 BIC, 50G,52A,51A,56A,57A,59A |
| MT103 | Block1 BIC, Block 2 BIC, 50A, 51A , 52A,53A,55A, 54A,56A,57A,59A |
| MT202 | Block1 BIC, Block 2 BIC, 52A,53A, 54A,56A,57A,58A |
| MT202COV | Block1 BIC, Block 2 BIC, 50A, 51A , 52A,53A, 54A,56A,57A,59A |

Block 1 and Block 2 of SWIFT messages are included by default. They can be enabled or disabled based on your requirement. For information on how to enable or disable a tag for screening, see Oracle Financial Services Transaction Filtering Administration Guide.

## 5.4 Narrative Screening

This section explains the BIC SWIFT tags that are screened for various Message Types.

Table 5 describes the Narrative Screening details.

**Table 5: Narrative Screening**

| Message Type | Tags/ Fields |
|---|---|
| MT101 | 20, 21R, 21,25,21F, 70, 77B |
| MT103 | 20,23E, 70,72, 77B |
| MT202 | 72,20, 21 |
| MT202COV | 20,21,72,70 |

# 6 Name and Address Matching

This section details the default configuration of Name and Address screening against sanctions lists or watch lists.

The matching strategy for entities in Oracle Watchlist Screening raises a possible match if there is an exact match or a fuzzy name match to a normal (non-acronym) entity name, or if there is an exact match to an acronym entity name.

The fuzzy entity name matching algorithms include some of the following techniques:

- Standardizing entity names (for example, different forms of company name suffixes are standardized to a common form)
- Ignoring insignificant name tokens
- Typo tolerance
- Allowance for missing name tokens
- Allowance for different tokenization of the name

## 6.1 Identifier preparation

The following identifiers are prepared for use in the individual and Entity matching process:

For Identifier preparation, Vessel and Aircraft come under Entity.

Table 6 describes the Individual and Entity Identifier Preparation details.

**Table 6: Individual and Entity Identifier Preparation**

| Identifier Description | Standard prepared attribute name | Summary of preparation logic |
|---|---|---|
| Individual Given Names | dnGivenNames | A space-separated list of the first and middle names of the individual, after normalization (see the name normalization section, below). |
| Individual Family Name | dnFamilyName | A normalized version of the family name (see the name normalization section, below). |
| Individual Full Name | dnFullName | A concatenation of the given names and family name, separated using spaces. |
| Original Script Name | dnOriginalScriptName | A whitespace normalized version of the original script name. |
| dnCity | dnCity | A pipe-separated list of cities associated with the individual data. |
| dnAddressCountryCode | dnAddressCountryCode | A space separated list of standard 2-character country codes. |
| dnEntityName | dnEntityName | The original entity name, after Name Normalization. |

The following sections describe the data preparation strategy for each of these identifiers.:

### 6.1.1 Name Normalization

The Individual, entity, vessel, and aircraft names are normalized using the following logic:

1. Standardization of accented characters.
2. Replacement of non-alpha (A-Z or a-z) characters with spaces.

If data is matched in the original language against original script names in the watch lists, then the appropriate character ranges must be removed from the Name Noise Characters Reference Data so that they are not replaced.

If transliteration of data is done before matching, then transliteration must also be done before name normalization.

3. Normalization of whitespace.
4. Conversion to upper case.

Note that the purpose of these transformations is not to create the most 'correct' name. For example, hyphens may be used in names in a number of ways, such as in a double-barreled surname, or as an alternative for a space when a surname has a qualifier (common in the World-Check data file).

In the former case, one might ideally want to preserve the hyphen, and in the latter case replace it with a space. In general, however, additional spaces in names will not cause names to miss matching, whereas different characters could.

Table 7 describes the Name Normalization example.

**Table 7: Name Normalization**

| Input data Forename | Input data Surname | Identifiers dnGivenNames | Identifiers dnFamilyName | Identifiers dnFullName |
|---|---|---|---|---|
| Carmelo | Raschellà | CARMELO | RASCHELLA | CARMELO RASCHELLA |
| Darwen | MANN`A | DARWEN | MANN A | DARWEN MANN A |
| Badr bin Saud bin Harib | AL-BUSAIDI | BADR BIN SAUD BIN HARIB | AL BUSAIDI | BADR BIN SAUD BIN HARIB AL BUSAIDI |
| A. Arnaldo G. | TAVEIRA | A ARNALDO G | TAVEIRA | A ARNALDO G TAVEIRA |
| Jose Mardônio | DA COSTA** | JOSE MARDONIO | DA COSTA | JOSE MARDONIO DA COSTA |

### 6.1.2 City and country identifiers

City and country values are derived from the source data wherever possible. There may be multiple possible cities or countries associated with an individual, perhaps because an individual resides in more than one country, has dual nationality, or resides in a different country from his/her nationality.

Country values are prepared as a space-separated list of two-character country codes in the `dnAllCountryCodes` attribute.

City values (which may contain spaces, for example, 'New York') are prepared as a pipe-separated list of cities in the `dnCity` attribute.

## 6.2 Clustering

Oracle Financial Services Transaction Filtering provides clusters for matching individuals and entities to watch lists during Sanctions screening. These clusters can be activated or deactivated, as required, and different cluster limits can be configured.

Table 8 describes the cluster list.

**Table 8: Clustering**

| Cluster Method |
|---|
| Individual Family Name |
| Individual Full Name Metaphone |
| Individual Given Names |
| Individual Full Name Trim |
| Individual  Initials |
| Original Script Name |
| Entity Name Tokens |
| Entity Name Meta |
| Entity Name Trim |
| Entity Start End Name Tokens |

This table shows the default configuration of Real-Time screening processes, but these may be customized independently of one another.

The data used to create the clusters is created before matching by the preparation process. In all cases, the clusters use the prepared and normalized name attributes `dnGivenNames`, `dnFamilyName` , `dnFullName`, `dnEntityName`, and `dnOriginalScriptName`. For further information see Name Normalization.

### 6.2.1 Family Name Cluster (dnClusterFamilyName)

The **Family Name** cluster provides a backup to the full name clusters. This is especially important where the given name data is incomplete, making it difficult to form a complete cluster key for two names. For example, the following three example records do not share any Full Name cluster keys, due to the initials in the second record and the spacing and spelling variations seen throughout:

Table 9 describes the Family Name Cluster details.

**Table 9: Family Name Cluster**

| dnFullName | Name tokens and trimmed values | | Identifiers dnFamilyName | dnClusterFullNameTrim |
|---|---|---|---|---|
| STEPHEN JEQE NKOMO | JEQE | JEQ | JEQNKO JEQSTE NKOSTE | JEQNKO\|JEQSTE\|NKOSTE |
| | NKOMO | NKO | | |
| | STEPHEN | STE | | |
| S J NKOMO | S | S | NKO | NKO |
| | NKOMO | NKO | | |
| | J | J | | |
| STEPHEN JEKE N KOMO | JEKE | JEK | JEKKOM JEKSTE KOMSTE | JEKKOM\|JEKSTE\|KOMSTE |
| | KOMO | KOM | | |
| | N | N | | |
| | STEPHEN | STE | | |

Clustering only on the family name circumvents this issue, but results in large clusters and a concomitant increase in the processing required to cross-check all the records.

The **Family Name** cluster builder counters spacing and punctuation differences by generating Metaphone keys for all tokens of the family name, AND the whole of the family name after all white space is trimmed. This is to ensure that family names such as those in the last two records in the example table below are all clustered together despite the spacing differences.

The default logic of the cluster builder is as follows:

1. Trim all white space from the normalized family name
2. Apply the **Metaphone** transformation to the result, outputting a key with a length of up to 4 characters
3. Strip common name qualifiers from the normalized family name, e.g. Abd, Al.
4. Split the family name into several name tokens, using a space delimiter.

Many other punctuation and noise characters are normalized to spaces before generating the cluster. For further information see Name Normalization.

5. Apply the **Metaphone** transformation to each name token, outputting a key with a length of up to 4 characters. If there were no tokens remaining after stripping common name qualifiers then apply the Metaphone transformation to the each name token of the original normalized family name.
6. Concatenate all the generated Metaphone keys
7. Deduplicate the list of keys

Table 10 describes the Metaphone Transormations for Family Name Cluster example.

**Table 10: Metaphone Transormations for Family Name Cluster**

| dnFamilyName | Tokens derived from dnFamilyName | Metaphone transformations | dnClusterFamilyName |
|---|---|---|---|
| ZHONG | ZHONG | JNK | JNK |
| XIAOJIAN | XIAOJIAN | SJN | SJN |
| ABACHE | ABACHE | APX | APX |
| ABANDA | ABANDA | APNT | APNT |
| ABD AL HAFIZ | HAFIZ ABDALHAFIZ | HFS APTL | HFS\|APTL |
| AL BUTHE | BUTHE ALBUTHE | P0 ALP0 | P0\|ALP0 |
| AL | AL | AL | AL |
| SOLEIMAN HAMAD | SOLEIMAN HAMAD SOLEIMANHAMAD | SLMN HMT SLMN | SLMN\|HMT |
| GOODRIDGE | GOODRIDGE | KTRJ | KTRJ |
| GOODRICH SR | GOODRICH SR GOODRICHSR | KTRX SR KTRK | KTRX\|SR\|KTRK |
| NKOMO | NKOMO | NKM | NKM |
| N KOMO | N KOMO NKOMO | N KM NKM | N\|KM\|NKM |

### 6.2.2 Individual Full Name Metaphone Pairs Cluster (dnClusterFullNameMeta)

The **Full Name Metaphone Pairs** cluster uses the normalized full name for the individual to generate a cluster key for every pair of names within the full name. The default logic of this is as follows:

1. Split the normalized full name into several name tokens, using space as a delimiter.

Many other punctuation and noise characters are normalized to spaces before generating the cluster. For further information see Name Normalization.

2. Sort the name tokens alphabetically.
3. Apply the **Metaphone** transformation (the standard double-metaphone algorithm) to each name token, outputting a key with a length of up to three characters.
4. Concatenate the Metaphone values, generating a final key value for each distinct pair of tokens.
5. Deduplicate the list of keys.

Table 11 describes the Full Name Metaphone Pairs Cluster example.

**Table 11: Full Name Metaphone Pairs Cluster**

| dnFullName | Name tokens and Metaphone values | | Distinct Cluster Keys | dnClusterFull-NameMeta |
|---|---|---|---|---|
| XIAO JIAN ZHONG | JIAN | JN | JNS JNJNK SJNK | JNS\|JNJNK\|SJNK |
| | XIAO | S | | |
| | ZHONG | JNK | | |
| ZHONG XIAOJIAN | XIAOJIAN | SJN | SJNJNK | SJNJNK |
| | ZHONG | JNK | | |
| MOHAMMED SANI ABACHE | ABACHE | ABX | APXMHM APXSN MHMSN | APXMHM\|APXSN \| MHMSN |
| | MOHAMMED | MHMT | | |
| | SANI | SN | | |
| JOSEPH TSANGA ABANDA | ABANDA | APNT | APNJSF APNTSN JSFTSN | APNJSF\|APNTSN \|JSFTSN |
| | JOSEPH | JSF | | |
| | TSANGA | TSNK | | |
| ABD AL WAHAB ABD AL HAFIZ | ABD | APT | APTAPT APTAL APTHFS APTAHP ALAL ALHFS ALAHP HFSAHP | APTAPT\|APTAL\|APTHFS \|APTAHP\|ALAL\|ALHFS \|ALAHP\|HFSAHP |
| | ABD | APT | | |
| | AL | AL | | |
| | AL | AL | | |
| | HAFIZ | HFS | | |
| | WAHAB | AHP | | |
| SULIMAN HAMD SULEIMAN AL BUTHE | AL | AL | ALP0 ALHMT ALSLM P0HMT P0SLM HMTSLM SLMSLM | ALP0\|ALHMT\|ALSLM\| P0HMT\|P0SLM\| HMTSLM \|SLMSLM |
| | BUTHE | P0 | | |
| | HAMD | HMT | | |
| | SULEIMAN | SLMN | | |
| | SULIMAN | SLMN | | |
| AL BUTHE SOLEIMAN HAMAD | AL | AL | ALP0 ALHMT ALSLM P0HMT P0SLM HMTSLM | ALP0\|ALHMT\|ALSLM\| P0HMT\|P0SLM\| HMTSLM |
| | BUTHE | P0 | | |
| | HAMAD | HMT | | |
| | SOLEIMAN | SLMN | | |
| REGINALD B GOODRIDGE | B | P | KTRRJN **NOTE:** Initials are ignored by default when generating cluster keys | KTRRJN |
| | GOODRIDGE | KTRJ | | |
| | REGINALD | RJNLT | | |

| dnFullName | Name tokens and Metaphone values | | Distinct Cluster Keys | dnClusterFull-NameMeta |
|---|---|---|---|---|
| REGINALD B SR GOODRICH | B | P | KTRRJN KTRSR RJNSR<br>**NOTE:** Initials are ignored by default when generating cluster keys | KTRRJN\|KTRSR\|RJNSR |
| | GOODRIDGE | KTRJ | | |
| | REGINALD | RJNLT | | |
| | SR | SR | | |
| STEPHEN JEQE NKOMO | JEQE | JK | JKNKM JKSTF NKMSTF | JKNKM\|JKSTF\|NKMSTF |
| | NKOMO | NKM | | |
| | STEPHEN | STFN | | |
| S J NKOMO | J | J | NKM<br>Initials are ignored by default when generating cluster keys | NKM |
| | NKOMO | NKM | | |
| | S | S | | |
| STEPHEN JEKE N KOMO | JEKE | JK | JKKM JKSTF KMSTF | JKKM\|JKSTF\|KMSTF |
| | KOMO | KM | | |
| | N | N | | |
| | STEPHEN | STFN | | |

## 6.2.3 Individual Given Names Cluster (dnClusterGivenNames)

The **Given Names** cluster provides a further backup to the remaining clusters, especially to deal with cases where names are not necessarily well-structured into family and given names.

Depending on the quality and culture of the name information, this cluster will often not be required. You can test the number of additional alerts identified by the cluster by running matching with this cluster disabled, and then running with it enabled. Comparing the new relationships against the old will highlight the relationships identified by using this cluster.

The default logic of the cluster builder is as follows:

1. Split the normalized full name into several name tokens, using space as a delimiter.

Many other punctuation and noise characters are normalized to spaces before generating the cluster. For further information see Name Normalization.

2. Standardize the normalized given names before clustering. This ensures, for example, that names such as 'William' and 'Bill' will be clustered together, although their raw Metaphone values are not the same. A space delimiter is used to split the name before standardizing.
3. Apply the **Metaphone** transformation to the whole of the given names value after token standardization, outputting a key with a length of up to 4 characters.

Table 12 describes the Given Names Cluster example.

**Table 12: Given Names Cluster**

| dnGivenNames | Metaphone values | dnClusterGivenNames |
|---|---|---|
| XIAO JIAN | SJN | SJN |
| ZHONG | JNK | JNK |
| MOHAMMED SANI | MHMT | MHMT |
| JOSEPH TSANGA | JSFT | JSFT |
| ABD AL WAHAB | APTL | APTL |
| SULIMAN HAMD SULEIMAN | SLMN | SLMN |
| AL BUTHE | ALP0 | ALP0 |
| REGINALD B | RJNL | RJNL |
| STEPHEN JEQE | STFN | STFN |
| S J | SJ | SJ |
| STEPHEN JEKE | STFN | STFN |

## 6.2.4 Individual Full Name Trim Pairs Cluster (dnClusterFullNameTrim)

On occasion, two names which are close matches may not generate a common cluster key using the **Full Name Metaphone Pairs** cluster.

Table 13 describes the Full Name Trim Pairs Cluster.

**Table 13: Full Name Trim Pairs Cluster**

| dnFullName | Name tokens and Metaphone values | | Distinct Cluster Keys | dnClusterFullNameMeta |
|---|---|---|---|---|
| XIAO JIAN ZHONG | JIAN | JN | JNS JNJNK SJNK | JNS\|JNJNK\|SJNK |
| | XIAO | S | | |
| | ZHONG | JNK | | |
| ZHONG XIAOJIAN | XIAOJIAN | SJN | SJNJNK | SJNJNK |
| | ZHONG | JNK | | |

These two records are a possible name match. However, the **Full Name Metaphone Pairs** cluster does not produce a common cluster key for the pair because the tokens ‘Xiao’ and ‘Xiaojian’ yield different three character Metaphone keys.

In order to match these cases efficiently, a **Full Name Trim Pairs** cluster is prepared in a similar way to the primary cluster, but without applying a Metaphone transformation. This allows for typos and spacing differences in the names, but is ‘left-biased’; that is, it demands that the first few characters of the names match.

The logic of the cluster is as follows:

1. Split the normalized full name into name tokens, using space as a delimiter.
2. Sort the name tokens alphabetically.
3. Apply the **Trim Characters** transformation to each name token, outputting a key with a length of (up to) 3 characters.
4. Concatenate the trimmed values, generating a final key value for each distinct pair of tokens.
5. Deduplicate the list of keys.

Table 14 describes the Trim Characters for Full Name Trim Pairs Cluster.

**Table 14: Trim Characters for Full Name Trim Pairs Cluster**

| dnFullName | Name tokens and trimmed values | | Cluster Keys | dnClusterFullNameTrim |
|---|---|---|---|---|
| XIAO JIAN ZHONG | JIAN | JIA | JIAXIA JIAZHO XIAZHO | JIAXIA\|JIAZHO\|XIAZHO |
| | XIAO | XIA | | |
| | ZHONG | ZHO | | |
| ZHONG XIAOJIAN | XIAOJIAN | XIA | XIAZHO | XIAZHO |
| | ZHONG | ZHO | | |
| MOHAMMED SANI ABACHE | ABACHE | ABA | ABAMOH ABASAN MOHSAN | ABAMOH\|ABASAN\|MOHSAN |
| | MOHAMMED | MOH | | |
| | SANI | SAN | | |
| JOSEPH TSANGA ABANDA | ABANDA | ABA | ABAJOS ABATSA JOSTSA | ABAJOS\|ABATSA\|JOSTSA |
| | JOSEPH | JOS | | |
| | TSANGA | TSA | | |
| ABD AL WAHAB ABD AL HAFIZ | ABD | ABD | ABDABD ABDAL ABDHAF ABDWAH ALAL ALHAF ALWAH HAFWAH | ABDABD\|ABDAL\|ABDHAF\|ABDWAH\|ALAL\|ALHAF\|ALWAH\|HAFWAH |
| | ABD | ABD | | |
| | AL | AL | | |
| | AL | AL | | |
| | HAFIZ | HAF | | |
| | WAHAB | WAH | | |
| SULIMAN HAMD SULEIMAN AL BUTHE | AL | AL | ALBUT ALHAM ALSUL ALSUL BUTHAM BUTSUL HAMSUL SULSUL | ALBUT\|ALHAM\|ALSUL\|BUTHAM\|BUTSUL\|HAMSUL\|SULSUL |
| | BUTHE | BUT | | |
| | HAMD | HAM | | |
| | SULEIMAN | SUL | | |
| | SULIMAN | SUL | | |
| AL BUTHE SOLEIMAN HAMAD | AL | AL | ALBUT ALHAM ALSOL BUTHAM BUTSOL HAMSOL | ALBUT\|ALHAM\|ALSOL\|BUTHAM\|BUTSOL \|HAMSOL |
| | BUTHE | BUT | | |
| | HAMAD | HAM | | |
| | SOLEIMAN | | | |
| REGINALD B GOODRIDGE | B | B | GOOREG **NOTE:** Initials are ignored by default when generating cluster keys | GOOREG |
| | GOODRIDGE | GOO | | |
| | REGINALD | REG | | |

| dnFullName | Name tokens and trimmed values | | Cluster Keys | dnClusterFullNa-meTrim |
|---|---|---|---|---|
| REGINALD B SR GOODRICH | B | B | GOOREG GOOSR REGSR | GOOREG\|GOOSR\|REGSR |
| | GOODRICH | GOO | | |
| | REGINALD | REG | | |
| | SR | SR | | |
| STEPHEN JEQE NKOMO | JEQE | JEQ | JEQNKO JEQSTE NKOSTE | JEQNKO\|JEQSTE\|NKOSTE |
| | NKOMO | NKO | | |
| | STEPHEN | STE | | |
| S J NKOMO | S | S | NKO<br>Initials are ignored by default when generating cluster keys | NKO |
| | NKOMO | NKO | | |
| | J | J | | |
| STEPHEN JEKE N KOMO | JEKE | JEK | JEKKOM JEKSTE KOMSTE<br>Initials are ignored by default when generating cluster keys | JEKKOM\|JEKSTE\|KOMSTE |
| | KOMO | KOM | | |
| | N | N | | |
| | STEPHE | STE | | |

## 6.2.5 Individual Initials (dnClusterInitials)

The **First Initial Last Name** cluster provides a clustering method to group together names that share the same first name initial and last name, and allows some variation for transposed names. The default logic of the cluster builder is as follows:

1. Split the normalized given names into several name tokens, using a space character as the delimiter.
2. Split the normalized family name into several name tokens, using a space character as the delimiter.
3. Generate the cluster key value as follows:
   - If there are two or more characters in the last token of the family name, then concatenate the first character of the given name with the last token of the family name.
   - If the last token of the family name is a single initial, then concatenate that character with the first token of the given name
4. Trim the cluster key to a maximum of 12 characters.

Table 15 describes the First and Last Name Cluster.

**Table 15: First and Last Name Cluster**

| dnGivenNames | dnFamilyName | dnClusterFirstLast |
|---|---|---|
| MARTIN | JONES | MJONES |
| MARTIN PETER | JONES | MJONES |
| MARTIN | MORGAN JONES | MJONES |
| JONES | M | MJONES |

## 6.2.6 Original Script Name (dnClusterOriginalScript)

The **Original Script Name** cluster provides a clustering method for matching names represented in non-Latin writing systems. The cluster builder generates a key for each token in the name.

A single cluster value of "Myanmar" is generated for original script names written in the Burmese alphabet irrespective of the name. This is needed because token splitting is not possible for the Myanmar writing system as it does not use a space character between words. As a result, all original script names in Burmese script will be compared during matching. This should not cause performance issues during screening providing there are a low number of customer records using this writing system.

The default logic of the cluster builder is as follows:

1. Split the original script name into several name tokens, using a space character as the delimiter.
2. Trim each name token to a maximum of 5 characters.
3. Concatenate all of the trimmed token values with a pipe separator
4. Deduplicate the list of keys.

Table 16 describes the Original Script Name Cluster.

**Table 16: Original Script Name Cluster**

| dnOriginalScriptName | dnClusterOriginalScript |
|---|---|
| Іван Антонавіч Шчурок | Іван\|Антон\|Шчуро |
| 林 紹 藏 | 林\|紹\|藏 |
| သင် သင် ဒေဝ | Myanmar |
| محمد محمد منشوري | منشور\|محمد |

## 6.2.7 Entity Name Tokens (dnClusterNameTokens)

This cluster uses the standardized entity name to generate cluster keys. The default logic is as follows:

1. Remove initials.
2. Remove common name tokens, such as Limited, or Corporation.
3. Normalize whitespace.

4. Convert space characters to pipe characters.

Table 17 describes the Entity Name Tokens Cluster.

**Table 17: Entity Name Tokens Cluster**

| dnEntityName | Name with initials and common name tokens stripped | dnClusterNameTokens |
|---|---|---|
| ANGLO<br>CARIBBEAN CO<br>LTD | ANGLO CARIBBEAN | ANGLO\|CARIBBEAN |
| GUAMATUR S A | GUAMATUR | GUAMATUR |

## 6.2.8 Entity Name Meta (dnClusterLongName)

This cluster uses the standardized entity name to generate cluster keys. The default logic is as follows:

1. Remove initials.
2. Remove common name tokens, such as Limited, or Corporation.
3. Normalize whitespace.
4. Remove common business words, such as Company, or Association.
5. Transliterate any non-Latin characters into Latin.
6. Apply the Metaphone transformation (the standard double-Metaphone algorithm) outputting a key with a length of up to eight characters.

Table 18 describes the Name Metaphone Cluster.

**Table 18: Name Metaphone Cluster**

| dnEntityName | Name with initials, common name tokens and common business words stripped | dnClusterLongName |
|---|---|---|
| HAVANA<br>INTERNATIONAL<br>BANK LTD | HAVANA BANK | HFNPNK |
| CIMEX S A | CIMEX | SMKS |
| LA EMPRESA<br>CUBANA DE<br>FLETES | EMPRESA CUBANA FLETES | AMPRSKPN |

## 6.2.9 Entity Name Trim (dnClusterShortName)

This cluster uses the standardized entity name to generate cluster keys. The default logic is as follows:

1. Remove all whitespace.
2. Left-trim the value to a maximum of 4 characters.

Table 19 describes the Name Trimmed Cluster.

**Table 19: Name Trimmed Cluster**

| dnEntityName | dnClusterShortName |
|---|---|
| HAVANA INTERNATIONAL BANK LTD | HAVA |
| CIMEX S A | CIME |
| LA EMPRESA CUBANA DE FLETES | LAEM |

### 6.2.10 Entity Start End Name Tokens (dnClusterStartEndNameTokens)

This clustering method is designed as a looser version of the Entity Name Tokens cluster and allows for variation in entity names by creating clusters for the first five and last five characters of each name token.

The default logic is as follows:

1. Remove initials.
2. Remove common name tokens, such as Limited, or Corporation.
3. Normalize whitespace.
4. For each token that is longer than five characters, replace with two new tokens that are:
   - The first five characters of the token
   - The last five characters of the token

Table 20 describes the Start/End Name Tokens Cluster.

**Table 20: Start/End Name Tokens Cluster**

| dnEntityName | Name with initials and common name tokens stripped | dnClusterStartEndNameTokens |
|---|---|---|
| HAVANA INTERNATIONAL BANK LTD | HAVANA INTERNATIONAL BANK | HAVAN\|AVANA\|INTER\|IONAL\|B ANK |
| CIMEX S A | CIMEX | CIMEX |
| LA EMPRESA CUBANA DE FLETES | LA EMPRESA CUBANA FLETES | LA\|EMPRE\|PRESA\|CUBAN\|UBA NA\|FLETE\|LETES |

## 6.3 Matching

Individual and entity matching is centered on individual and entity names respectively. Other items of data, such as associated countries and cities, are used to strengthen a possible match. Match rule groups are places in the following order:

- Individual name match groups
- Aircraft name match groups
- Vessels name match groups
- Entity name match groups

The following general notes describe the approach to matching:

- Matches are ranked according to how well the name matches. An exact name match rates as a match at the highest level, with the lowest level being represented by two loosely possible name matches with a different name structure. Further ranking is imposed by how well additional information (such as city or country information, and date of birth information) matches between the records.
- Oracle Financial Services Transaction Filtering allows for various levels of name match, including, but not limited to:
  - Name variation recognition. This is carried out by name standardization. For example, all variations of Mohammed (Muhamad, Mohammad, Mohamed and so on) are substituted with 'Mohammed' when matching. This is particularly used for given names, though also applied when matching whole names. For example, more than 20 variations of the name 'Mohammed' are recognized and considered to be the same name.
  - Allowances for name abbreviation and initials. For example, 'Pete' is a possible match to 'Peter', and 'J' is a possible match to 'John'.
  - Allowances for typographical errors and transliteration differences. For example, 'Abdool' is a possible match to 'Abdul', even if the variants are not standardized.
  - Allowances for names being out of order or structured differently. For example, 'Mohammed Abbas Al-Tikriti' can be matched with 'Mohammed Al-Tikriti Abbas'.
  - Allowance for additional names. For example, 'Juan Carlos Ferreira' can be matched with 'Juan Ferreira'.
  - Allowance for names being split differently. For example, 'Xiao Jian' is a match to 'Xiaojian'.
- Oracle Financial Services Transaction Filtering attempts to prevent false positives by various means, including, but not limited to, the following methods:
  - Backing up typo tolerance with Metaphone matching. For example, 'Mary' and 'Mark' are not considered a match, although they are only one character different.
  - Backing up typo tolerance with consideration of the percentage of characters that are different. For example, the initials 'A' and 'E' are not considered a match, even though they are only one character different.
  - Considering the different significance and commonality of name tokens. For example, if name qualifiers such as 'Al' are shared between two Arabic names, this is not as significant as if an uncommon name such as 'Abbas' is shared.

It may be advisable to tune the set of match rules that are activated. In particular, you may wish to activate or deactivate some of the lower match rules in the list, which lead to the weakest name matches. Factors affecting the usefulness of these rules include:

- the policies of the organization
- the quality of the transaction data
- the provenance of the transaction data

For example, Asian and Arabic names may be subject to more typographical and name ordering issues than other names. Where the data contains many of these names, the lower strength rules may identify more possible matches. The organization may want to review some or all of these as a matter of policy, or it may consider the matches too weak to review.

The required rules are easily activated or deactivated as needed in Oracle Financial Services Transaction Filtering.

### 6.3.1 Match Rules

There are several different types of match rule involved in the name and address screening:

- The name matching rules: These are organized by the level of name match, with the strongest name matching rules placed at the top of the decision table.

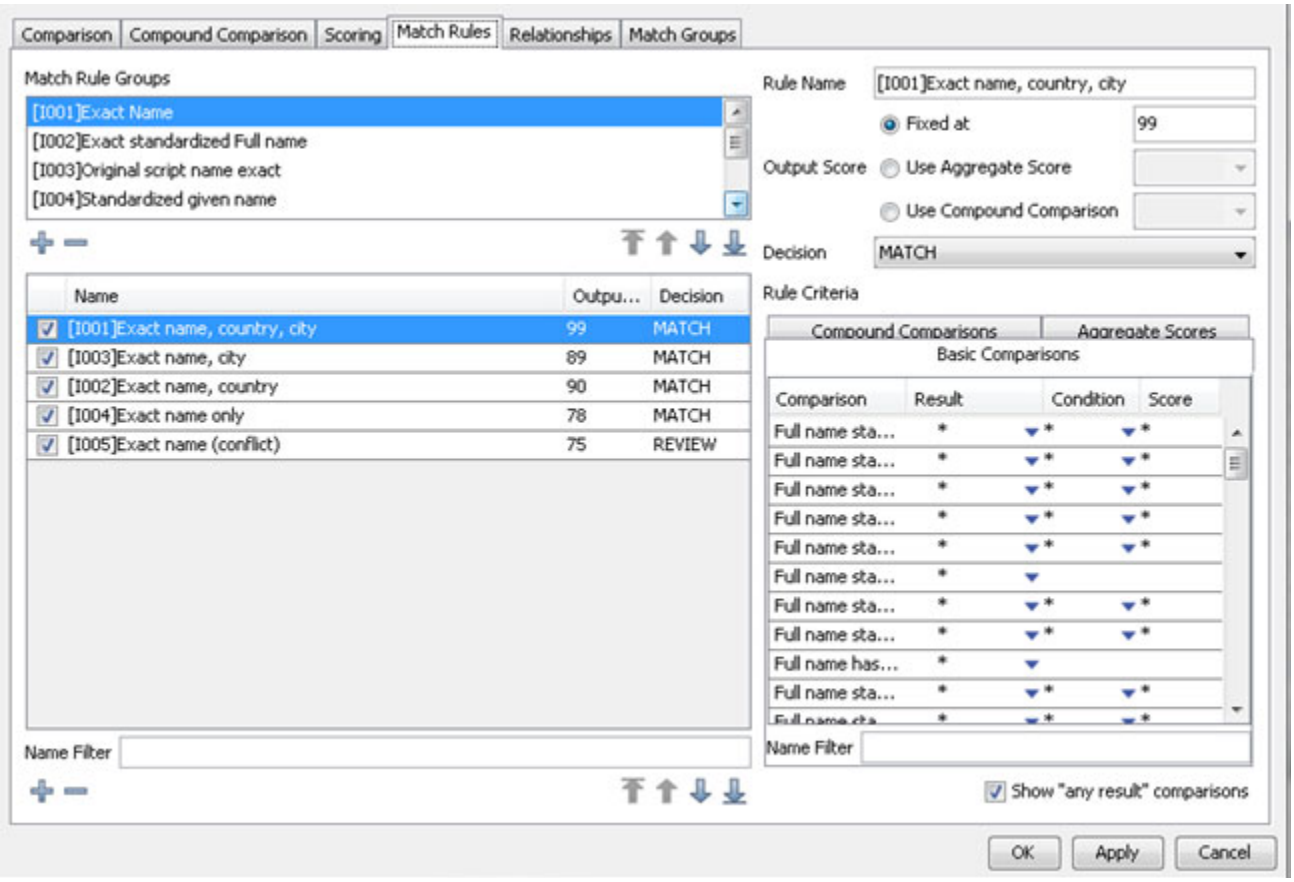
This means that the match rules are not ordered by strength across all identifiers. For example, a weaker name match that is strengthened by matches on city and country is likely to be a stronger overall match than a strong name with strongly contradictory data in the other fields.

The last rule in each set is a 'conflict' rule, and in many cases will be disabled by default. These rules allow records which fulfill the specified level of name match but have conflicting supporting data fields indicating that a true match is unlikely.

- The loose name matching rules: These are also based around name matching, but identify looser matches and are not enabled by default. These rules are likely to result in a large number of false positive matches and are most likely to be of use when screening against sanctions lists, where it is important that no true matches are missed.

For the sake of clarity, match rules are divided into groups. As each group is selected, the match rules it contains are displayed in the window below:

**Figure 2: Match Rules Displayed**




The priority of the groups can be changed using the arrows below the **Match Rules Group** list. When a group is highlighted:

- Click  to move it up one place in the list.

- Click  to move it down one place in the list.
- Click  to move it to the top of the list.
- Click  to move it to the bottom of the list.

The remainder of this section describes the matching rules that are present in Oracle Financial Services Transaction Filtering in greater detail.

## 6.3.2 Individual Name Matching Rules

Table 21 describes the individual name matching rules:

**Table 21: Individual Name Matching Rules**

| Group Code | Matching Rule | Logic Summary | Example Matching Data | |
|---|---|---|---|---|
| I001 | Exact name | Full name match after name standardization using full name map | | |
| I002 | Exact standardized Full name | Given names and family name match exactly. | | |
| | | | JOSEPH<br>JOSEPH | TSANGA<br>T'SANGA |
| I003 | Original script name exact | The original script Name fields match exactly. | | |
| | | | АЛЕКСАНДР ОСОКИН | АЛЕКСАНДР ОСОКИН |
| I004 | Standardized given name | Given names match after name standardization using Given name map. Family name matches exactly. | | |
| | | | BILL | JONES |
| | | | WILLIAM | JONES |
| I005 | Full name | The full name matches exactly, after standardization of all name tokens using the Given Name Map. | | |
| | | | JOHN MIKE SMITH | |
| | | | JOHN MICHAEL SMITH | |
| I006 | Full name without titles | The full name matches exactly, after standardization of all name tokens using the Given Name Map and removal of titles. | | |
| | | | DR DOUGLAS BAKER | |
| | | | DOUGLAS BAKER | |

| Group Code | Matching Rule | Logic Summary | Example Matching Data | |
|---|---|---|---|---|
| I007 | Abbreviated standardized given name | Given names match using a **Starts With** comparison, after name standardization using the Given Name Map. Family name matches exactly. | | |
| | | | JOSEPH ABANDA | TSANGA |
| | | | JOSEPH | T'SANGA |
| I008 | Given name similar and sounds like | Given name matches with an **Edit Distance** of 1 or 2 after name standardization. At least one of the given names, excluding initials, must match by a 4-character Metaphone key. Family name matches exactly | | |
| | | | JOSEPH | ABANDA |
| | | | JOESPH | ABANDA |
| I009 | First name similar and sounds like | The first given name matches with an **Edit Distance** of 1 or 2 and with a **Character Match Percentage** of 66% or more, after given name standardization. At least one of the given names, excluding initials, must match by a 4-character Metaphone key. Family name matches exactly. | | |
| | | | AMER MOHAMMAD RASHEED | AL UBAIDI |
| | | | AMIR RASHID MOHAMMED | AL UBAIDI |
| I010 | Additional given names | All name tokens from the given names field with fewest tokens must be present in the other given names field. Family name matches exactly . | | |
| | | | MOHAMMED | HANIF |
| | | | DIN MOHAMED | HANIF |

| Group Code | Matching Rule | Logic Summary | Example Matching Data | |
|---|---|---|---|---|
| I011 | Additional names | All name tokens from the full name with fewest tokens must be present in the other full name. At least 2 name tokens must match with the same matching logic; that is, if a name only has one token it is not considered a match. At least 2 name tokens must exist in the Full Name.<br>Word Match Count may return >1 if a single name matches twice in a longer name string. For example, 'ABDUL' matches 'ABDUL ABDUL' with a Word Match Count of 2.<br>Matching is order sensitive. | | |
| | | | LOTFI RIHANI | |
| | | | LOTFI BEN ABDUL HAMID BEN ALI RIHANI | |
| I012 | Original script name in any order | All names in the original script name fields match, regardless of order. | | |
| | | | Καρλος Μολινα | Μολινα Καρλος |
| I013 | Original script name with typos | Original script name fields match with an 80%+ **Character Match Percentage** score. | | |
| | | | Καρλος Μολινα | Καρλος Μολιννα |

| Group Code | Matching Rule | Logic Summary | Example Matching Data | |
|---|---|---|---|---|
| I014 | All names in any order | All names in the full name match (using a **Word Edit Distance** of 0) after name token standardization, in any order. A single typo (1 character edit) is allowed in each name token. | | |
| | | | ABDUL JABBER OMARI | |
| | | | OMARI ABDUL JABBER | |
| I015 | Abbreviated given name | Given names match using a **Starts With** comparison. Family name is a close metaphone match. | | |
| | | | CHRIS | HUNT |
| | | | CHRISTOPHER | HUNTER |
| I016 | Abbreviated given name and family name typos | Given names match using a **Starts With** comparison, after name standardization using Given Name Map. Family name matches with an edit difference of 1-2. At least one of the family name tokens, excluding initials must match by a 4-character Metaphone key. | | |
| | | | IBRAHIM ABDUL SALAM | MOHAMED BOYASSEER |
| | | | IBRAHIM | BOYASEER |

| Group Code | Matching Rule | Logic Summary | Example Matching Data | |
|---|---|---|---|---|
| I017 | Abbreviated given name without titles and family name with typos | The first given name matches with a **Starts With** match, after name token standardization and stripping titles. Family name matches with an edit difference of 1-2. At least one of the family name tokens, excluding initials, must match by a 4-character Metaphone key. | | |
| | | | SAHIR | BARHAN |
| | | | DR SAHIR MUSA | BERHIN |
| I018 | Original script name in any order with typos | All names in the original script name fields match, regardless of order, with each name requiring an 80%+ **Character Match Percentage** score. | | |
| | | | Хасан Ченгић | Ченгић Хассан |
| I019 | First name and full name similar and sounds like | The full name matches with a **Character Match Percentage** of 80% or above, after name token standardization. At least one of the family name tokens, excluding initials, must match by a 4-character Metaphone key. | | |
| | | | MOHAMMAD HUSAYN | MASTASAEED |
| | | | MOHAMMAD HASSAN | MASTASAEED |

| Group Code | Matching Rule | Logic Summary | Example Matching Data | |
|---|---|---|---|---|
| I020 | Given name similar and family names and sounds like | The given name matches with an **Edit Distance** of 1 or 2, after name standardization. The given name matches by 4-character Metaphone key, after name standardization. The family name matches with an **Edit Distance** of 1-2. The family name matches by 4-character Metaphone key. | | |
| | | | AMER MOHAMMAD RASHEED | AL UBAIDI |
| | | | AMIR RASHID MOHAMMED | AL UBEIDI |
| I021 | Abbreviated given name and family name similar | The first given name matches with a **Starts With** match, after name token standardization. The family name matches with an **Edit Distance** of 1 or 2. The family name matches by 4-character Metaphone key. | | |
| | | | VIKTOR ANATOLYEVICH | BOUT |
| | | | VICTOR | BOOT |
| I022 | Full Name no whitespace | Combination of Given name an Family name without spaces | CHRIS<br>CHRISTOPHER | HUNT<br>HUNTER |
| I023 | Original script name additional names | All names in one original script name field must be fully contained within the other field, provided there are at least two names in each field. | | |
| | | | Миленко Врачар | Миленко Иванович Врачар |

| Group Code | Matching Rule | Logic Summary | Example Matching Data |
|---|---|---|---|
| I024 | Additional names typo tolerant | All name tokens from the full name with fewest tokens must be present in the other full name. A character error tolerance of 20% is allowed (that is, one character edit every 5 characters). At least 2 name tokens must match with the same matching logic. If a name contains only one token it is not considered a match according to this rule. **NOTE:** Word Match Count may return >1 if a single name matches twice in a longer name string. For example, 'ABDUL' matches 'ABDUL ABDUL' with a Word Match Count of 2. Matching is order sensitive. | |
| | | | ABDUL WAHED SHAFIQ |
| | | | ABDUL WAHAD |
| I025 | Full name contained and multiple names in common | The full name matches with a **Contains** match, after standardization of all name tokens using the Given Name Map. At least 2 name tokens must match in the full name. | |
| | | | ABU BAKAR |
| | | | ABU BAKAR BA'ASYI |

| Group Code | Matching Rule | Logic Summary | Example Matching Data | |
|---|---|---|---|---|
| I026 | Full name characters longer | The full name matches with a **Longest Common Substring Sum Percentage** of 90%+, relating to the longer string, and considering substrings of 5 characters or more in length, after name standardization. | | |
| | | | MOHAMMED AL GHABRA | |
| | | | ALGHABRA MUHAMAD | |
| | | | RAMATULLAH WAHIDYAR FAQIR MOHAMMAD | |
| | | | WAHIDYAR RAMA TULLAH | |
| I027 | Original script name additional names with typos | All names in one original script name field must be fully contained within the other field, provided there are at least two names (all of which have an 80%+ Character Match Percentage) in each field. | | |
| | | | ЮРИ НЕЁЛОВ | Юрий Васильевич Неёлов |
| I028 | Abbreviated first name | The first given name matches with a **Starts With** match, after name token standardization. Family name matches exactly. | | |
| | | | KHADAF ABUBAKAR | JANJALANI |
| | | | KHADAFFI | JANJALANI |

| Group Code | Matching Rule | Logic Summary | Example Matching Data |
|---|---|---|---|
| I029 | Additional names in any order | All name tokens from the full name with fewest tokens must be present in the other full name. At least 2 name tokens must match with the same matching logic. If a name contains only one token it is not considered a match according to this rule. **NOTE:** Word Match Count may return >1 if a single name matches twice in a longer name string. For example, 'ABDUL' matches 'ABDUL ABDUL' with a Word Match Count of 2. Matching is order insensitive. | |
| | | | HA THI NGUYEN |
| | | | THI HA |

| Group Code | Matching Rule | Logic Summary | Example Matching Data |
|---|---|---|---|
| I030 | Additional names in any order typo tolerant | All name tokens from the full name with fewest tokens must be present in the other full name. A character error tolerance of 20% is allowed (that is, one character edit every 5 characters). At least 2 name tokens must match with the same matching logic. If a name contains only one token it is not considered a match according to this rule. **NOTE:** Word Match Count may return >1 if a single name matches twice in a longer name string. For example, 'ABDUL' matches 'ABDUL ABDUL' with a Word Match Count of 2. Matching is order insensitive. | <br>STEPHENS MARTIN<br>MARRTIN JOHN STEPHENS |

## 6.3.3 Loose Individual Name Matching Rules

Table 22 describes the loose individual name matching rules:

**Table 22: Loose Individual Name Matching Rules**

| Group Code | Matching Rule | Logic Summary | Example Matching Data |
|---|---|---|---|
| I031 | Full name characters shorter only | The full name matches with a **Longest Common Substring Sum Percentage** of 90%, relating to the shorter string, and considering substrings of 5 characters or more in length, after name standardization. At least 2 name tokens must exist in the full name. | |
| | | | ABU BAKAR |
| | | | ABU BAKAR BA'ASYI |
| I032 | Full name no initials match with initials in any order relating to shorter | All initials in one Full Name field must be fully contained within the initials of the other Full Name field; AND<br>the standardized Full Name field without initial must be fully contained within the other standardized Full Name field without initials; AND<br>both fields must contain at least two names. | |
| | | | CARL J FISHER |
| | | | J C FISHER |

| Group Code | Matching Rule | Logic Summary | Example Matching Data |
|---|---|---|---|
| I033 | Full name contained, last initial same, primary list is single token | The Full Name field from the watch list record contains only one name, which is fully contained within the record being screened;<br>AND<br>the initial of the last name in the record being screened must match the initial of the name in the watch list record. | <br>JANINE CHERRY<br>CHERRY |

### 6.3.4 Aircraft Matching Rules

Table 23 describes the details of the Aircraft matching rules:

**Table 23: Aircraft Matching Rules**

| Group Code | Matching Rule | Summary of Rule Logic | Example Matching Data |
|---|---|---|---|
| A001 | Aircraft part-standardized name exact | The part-standardized entity name matches the name of a listed aircraft exactly | |
| A002 | Aircraft name exact | The entity name matches the name of a listed aircrafts after number cardinal and ordinal standardization | 4TH YK-AYF<br>FOURTH YK-AYF |
| A003 | Aircraft part-standardized name with typos | The part-standardized entity name matches the name of a listed aircraft with a Character Match Percentage of 80-99%. | N840PN 1<br>N840PN |
| A004 | Aircraft name with typos | The entity names match with a Character Match Percentage of 80-99% after number cardinal and<br>Ordinal standardization | |

## 6.3.5 Vessel Matching Rules

Table 24 describes the details of the Vessel matching rules:

**Table 24: Full Name Trim Pairs Cluster**

| Group Code | Matching Rule | Summary of Rule Logic | Example Matching Data |
|---|---|---|---|
| V001 | Vessel<br>part-standardized name exact | The part-standardized entity name matches the name of a listed vessel exactly. | DYNASTY<br>DYNASTY |
| V002 | Vessel name exact | The entity name matches the name of a listed vessel after number cardinal and ordinal standardization | 4th OCEAN<br>FOURTH OCEAN |
| V003 | Vessel<br>part-standardized name with typos | The part-standardized entity name matches the name of a listed vessel with a Character Match Percentage of 80-99% | RAHIM<br>RAHIM 3 |
| V004 | Vessel name with typos | The entity name matches the name of a listed vessel after number cardinal and ordinal standardization | RAHUM 3<br>TRAHIM THREE |

## 6.3.6 Entity Matching Rules

The match rules in Oracle Financial Services Transaction Filtering are organized by the level of entity name match, with the strongest name matching rules at the top of the decision table.

There are two types of matching rules involved in entity screening:

- Entity name matching rules: Entity name matching rules are organized by the level of entity name match, with the strongest matching rules placed at the top of the decision table. This means that the match rules are not ordered by strength across all identifiers. For example, a weaker match rule that is strengthened by matches on City and Country is likely to be a stronger overall match than a strong match rule with strongly contradictory data in the other fields.
- Loose entity matching rules: These are also based around entity name matching, but identify looser matches and are not enabled by default. These rules are likely to result in a large number of false positive matches and are most likely to be of use when screening against sanctioned lists where it is important that no true matches are missed.

For the sake of clarity, match rules are divided into groups, as shown in the below tables.

All entity matching rules use a standardized form of the entity name. The strongest rules use the 'part-standardized name', meaning the entity names match after only simple global standardizations (such as considering AND and & as the same) are applied. Other rules apply additional rules for standardization as noted in the table below.

Usage of 'word' in the table below implies a space-delimited token in the prepared names.

Table 25 describes the Entity Name Matching Rules.

**Table 25: Entity Name Matching Rules**

| Group Code | Name Matching Rule | Summary of Rule Logic | Example Matching Data |
|---|---|---|---|
| E001 | Part-standardized name exact | The part-standardized entity name matches a listed entity name exactly | HUMAN APPEAL INTERNATIONAL<br>HUMAN APPEAL INTERNATIONAL |
| E002 | Name exact | The entity names match exactly after number cardinal and ordinal standardization. | ABN Bank<br>ABN |
| E003 | Original script name exact | The original script names match exactly. | НИАЭП ОАО<br>НИАЭП ОАО |
| E004 | Name without suffixes exact | The entity names match exactly after number cardinal and ordinal standardization, and after common company prefixes and suffixes are removed. | CAPITAL DIRECT LTD<br>CAPITAL |
| E005 | Name without business words<br>similar and sounds<br>like | The entity names match with a Word Match Percentage of 80% after number cardinal and ordinal standardization, and after common company prefixes, suffixes and other words are removed. The first word of each name has the same 4-character Metaphone key. | PARAGON INVESTMENT<br>CORPORATION<br>PIC |

| Group Code | Name Matching Rule | Summary of Rule Logic | Example Matching Data |
|---|---|---|---|
| E006 | Name without business words exact | The entity names match exactly after number cardinal and ordinal standardization, and after common company prefixes, suffixes and other words are removed. | LIFE HEALTHCARE GROUP<br>HOLDINGS LTD<br>LHCG |
| E007 | Name without business words has all words out-of-order | All remaining words in each entity name match exactly, but in any order, after number cardinal and ordinal standardization, and after common company prefixes, suffixes and other words are removed. | HEALTH EDUCATION SERVICES<br>HEALTH SERVICES |
| E008 | Name without<br>suffixes starts with and multiple names in common | The entity names are a Starts With match after number cardinal and ordinal standardization, and after common company prefixes and suffixes are removed. There are at least two significant words (not common business words) in common between the two names. The listed name is not an acronym alias of a longer primary entity name. | BAE SYSTEMS (LANCASTER<br>HOUSE) LIMITED<br>BAE SYSTEMS LIMITED |

| Group Code | Name Matching Rule | Summary of Rule Logic | Example Matching Data |
|---|---|---|---|
| E009 | Name without business words has all words with typos | All remaining words in each entity name match with a Character Match Percentage of 80 or more, after number cardinal and ordinal standardization, and after common company prefixes, suffixes and other words are removed | GERBERA ASSOCIATES LTD<br>GERBARA ASSOCIATES LTD |
| E010 | Original script name in any order | All words in the Original Script Names match exactly, in any order | НИАЭП ОАО<br>НИАЭП ОАО |
| E011 | Original script name with typos | The Original Script Names match with a Character Match Percentage of 80% or more. | НИАЭП ОАО<br>НИАЭП ОАО |
| E012 | Name without business words with typos, and sounds like | The entity names match with a Character Match Percentage of 80 ore more after number cardinal and ordinal standardization, and after common company prefixes, suffixes and other words are removed. The first word of each name has the same 4-character Metaphone key and the first three letters of each name are the same. | GOLDSTREAM PROPERTIES LTD<br>GOLDSTREAM PROPERTIES LTD |

| Group Code | Name Matching Rule | Summary of Rule Logic | Example Matching Data |
|---|---|---|---|
| E013 | Name without suffixes contains,<br>similar and multiple names in common | The entity names are a Contains match and the Word Edit Distance is no more than one between the names (where each word matches with a Character Match Percentage of 80 or more), after number cardinal and ordinal standardization, and after common company prefixes and suffixes are removed. There are at least two significant words (not common business words) in common between the two names. | HAMPSHIRE HERITAGE<br>DEVELOPMENTS LTD<br>HAMPSHIRE HERITAGE<br>DEVELOPMENTS LTD |
| E014 | Name has additional words, sounds like and multiple names in common | All words in the shorter entity name exist in the longer entity name (in order) after number cardinal and ordinal standardization, and after common company prefixes and suffixes are removed. There are at least two significant words (not common business words) in common between the two names. The list name is not an acronym alias of a longer primary entity name. | MOSCOW CITY CENTER PLC<br>MOSCOW CITY CENTER PLC |

| Group Code | Name Matching Rule | Summary of Rule Logic | Example Matching Data |
|---|---|---|---|
| E015 | Name without business words contains, sounds like and multiple names in common | The entity name is a Contains match with a listed entity name, after number cardinal and ordinal standardization, and after common company prefixes, suffixes and other words are removed. There are at least two significant words (not common business words) in common between the two names. The first word of each name has the same 4-character Metaphone key. | HI-TECH RECRUITMENT LTD<br>HI-TEC RECRUITMENT LTD |
| E016 | Original script name in any order with typos | All words in the original script name match with a Character Match Percentage of 80 or more, in any order. | НИАЭП ОАО<br>НИАЭП ОАО |
| E017 | Name without business words has most words out-of-order | The entity names match (in any order) with a Word Match Percentage of between 75 and 99, after number cardinal and ordinal standardization, and after common company prefixes, suffixes and other words are removed. The list name is not an acronym alias of a longer primary entity name. | BACK TO HEALTH CLINICS LIMITED<br>BACK TO HEALTH CLINICS LIMITED |

| Group Code | Name Matching Rule | Summary of Rule Logic | Example Matching Data |
|---|---|---|---|
| E018 | Name without business words, similar, sounds like, with multiple names<br>and a residual token in common.<br>The group name differs from the rule name. | All words in the shorter entity name exist in the longer entity name (in order) after number cardinal and ordinal standardization, and after common company prefixes, suffixes and other words are removed. There are at least two significant words (not common business words) in common between the two names, and at least one of these is not a word in the English dictionary or a very common word in Watchlist name data. The list name is not an acronym alias of a longer primary entity name. | CHARLES F ASH<br>CONSTRUCTION CO INC<br>CHARLES F ASH<br>CONSTRUCTION CO INC |

| Group Code | Name Matching Rule | Summary of Rule Logic | Example Matching Data |
|---|---|---|---|
| E019 | Name without business words, similar with typos, sounds like, with multiple names and<br>residual token in<br>common<br>The group name differs from the rule name. See the Match dialog for details. | All words in the shorter entity name match with a Character Match Percentage of 80 or more in the longer entity name (in order) after number cardinal and ordinal standardization, and after common company prefixes, suffixes and other words are removed. There are at least two significant words (not common business words) that match with a Character Match Percentage of 80 or more, and at least one of these is not a word in the English dictionary or a very common word in Watchlist name data. The list name is not an acronym alias of a longer primary entity name. | CLARKS HOME BAKERY LTD<br>CLARKS HOME BAKERY LTD |

| Group Code | Name Matching Rule | Summary of Rule Logic | Example Matching Data |
|---|---|---|---|
| E020 | Name has additional words tolerant, sounds like and multiple names in common | All words in the shorter entity name match in the longer entity name (in order) with a Character Match Percentage of 80 or more<br>after number cardinal and ordinal standardization. There are at least two significant words (not common business words) in common between the two names. The list name is not an acronym alias of a longer primary entity name. | ABU AL FULUS<br>ABU AL FULUS |
| E021 | Name without suffixes contains, similar and residual token in common | The entity names are a Contains match and the Word Edit Distance is no more than one between the names (where each word matches with a Character Match Percentage of 80 or more), after number cardinal and ordinal standardization, and after common company prefixes and suffixes are removed. There is at least one significant word in common (not a common business word, a word in the English dictionary or a very common word in Watchlist name data). | ACCLAIM ACM LTD<br>ACCLAIM ACM LTD |

| Group Code | Name Matching Rule | Summary of Rule Logic | Example Matching Data |
|---|---|---|---|
| E022 | Name without suffixes starts with and residual token in common | The entity names are a Starts With match after number cardinal and ordinal standardization, and after common company prefixes and suffixes are removed. There is at least one significant word in common (not a common business word, a word in the English dictionary or a very common word in Watchlist name data). The listed name is not an acronym alias of a longer primary entity name. | ENRON METALS BROKERS LTD<br>ENRON METALS BROKERS LTD |
| E023 | Name without suffixes starts with and substring in common | The entity names are a Starts With match, and there is a common substring at least 8 characters in length, after number cardinal and ordinal standardization, and after common company prefixes and suffixes are removed. The listed name is not an acronym alias of a longer primary entity name. | ACCURATE SECTION BENDERS LTD<br>ACCURATE SECTION BENDERS LTD |

| Group Code | Name Matching Rule | Summary of Rule Logic | Example Matching Data |
|---|---|---|---|
| E024 | Name without suffixes contains, residual token in common and significant overlap | The entity names are a Contains match and the Word Match Percentage is 50 or more, after number cardinal and ordinal standardization, and after common company prefixes and suffixes are removed. There is at least one significant word in common (not a common business word, a word in the English dictionary or a very common word in Watchlist name data). | NON EMERGENCY TRANSPORT INC<br>NON EMERGENCY TRANSPORT INC |
| E025 | Name without common tokens exact, and multiple residual tokens in common | The entity names match exactly, with at least two words matching, after number cardinal and ordinal standardization, and after common company prefixes, suffixes, and other words, and all English dictionary and common Watchlist name words are removed. | LIFE CARE CENTER PUNTA GORDA<br>LIFE CARE CENTER PUNTA GORDA |
| E026 | Original script name has additional names | All words in the shorter original script name match in the longer original script name (in order), and there are at least two matching words. | НИАЭП ОАО<br>НИАЭП ОАО |

| Group Code | Name Matching Rule | Summary of Rule Logic | Example Matching Data |
|---|---|---|---|
| E027 | Name without suffixes contains, multiple names in common and<br>significant overlap | The entity names are a Contains match and the Word Match Percentage is 50 or more, after number cardinal and ordinal standardization, and after common company prefixes and suffixes are removed. There is at least two significant words (not common business words) that match with a Character Match Percentage of 80 or more. | CAPITAL CITY TRANS SERV INC<br>CAPITAL CITY TRANS SERV INC |
| E028 | Name without business words similar and full name sounds like | The entity names match with a Character Match Percentage of between 80 and 99 after number cardinal and ordinal standardization, and after common company prefixes, suffixes and other words are removed. The names share the same metaphone key after number cardinal and ordinal standardization. | IBERIA AIRLINES<br>IBERIAN AIRLINES |

| Group Code | Name Matching Rule | Summary of Rule Logic | Example Matching Data |
|---|---|---|---|
| E029 | Name without business words similar with typos, sounds like and significant overlap | All words in the shorter entity name match with a Character Match Percentage of 80 or more in the longer entity name (in order) after number cardinal and ordinal standardization, and after common company prefixes, suffixes and other words are removed. The names match with a Word Match Percentage of 50 or more when common business words are not stripped. There are at least two significant words (not common business words) that match with a Character Match Percentage of 80 or more.<br>The first word of each name has the same 4-character Metaphone key. The list name is not an acronym alias of a longer primary entity name. | MED AMERICA CLINICS INC<br>MED AMERICA CLINICS INC |

| Group Code | Name Matching Rule | Summary of Rule Logic | Example Matching Data |
|---|---|---|---|
| E030 | Name has additional words, sounds like<br>and residual token in common | All words in the shorter entity name exist in the longer entity name (in order) after number cardinal and ordinal standardization. There is at least one significant word (not a common business word, an English dictionary word or a word or a common Watchlist name word) in common between the two names. The list name is not an acronym alias of a longer primary entity name. | DJ CASE AND ASSOCIATES INC<br>DJ CASE AND ASSOCIATES INC |
| E031 | Name has additional words with typos, sounds like and residual token in common | All words in the shorter entity name match with a Character Match Percentage of 80 or more in the longer entity name (in order) after number cardinal and ordinal standardization. There is at least one significant word (not a common business word, an English dictionary word or a word or a common Watchlist name word) that matches with a Character Match Percentage of 80 or more. The list name is not an acronym alias of a longer primary entity name. | GARLICK HELICOPTERS INC<br>GARLICK HELICOPTERS INC |

| Group Code | Name Matching Rule | Summary of Rule Logic | Example Matching Data |
|---|---|---|---|
| E032 | Name has additional words, sounds like and substring in common | All words in the shorter entity name exist in the longer entity name (in order) after number cardinal and ordinal standardization. There is a common substring of at least 8 characters in length between the two names after number cardinal and ordinal standardization, and after common company prefixes, suffixes and other words are removed. The list name is not an acronym alias of a longer primary entity name. | NATIONWIDE SECRETARIAL SERVICES LTD<br>NATIONWIDE SECRETARIAL SERVICES LTD |
| E033 | Name without business words, similar, sounds like and multiple names in common | All words in the shorter entity name match in the longer entity name (in order) after number cardinal and ordinal standardization, and after common company prefixes, suffixes and other words are removed. There are at least two significant words (not common business words) that match. The first word of each name has the same 4-character Metaphone key. The list name is not an acronym alias of a longer primary entity name. | CENTRAL OKLAHOMA FAMILY MEDICAL CENTER<br>CENTRAL OKLAHOMA FAMILY MEDICAL CENTER |

| Group Code | Name Matching Rule | Summary of Rule Logic | Example Matching Data |
|---|---|---|---|
| E034 | Name without business words, similar with typos, sounds like and multiple names in common | All words in the shorter entity name match with a Character Match Percentage of 80 or more in the longer entity name (in order) after number cardinal and ordinal standardization, and after common company prefixes, suffixes and other words are removed. There are at least two significant words (not common business words) that match with a Character Match Percentage of 80 or more. The first word of each name has the same 4-character Metaphone key. The list name is not an acronym alias of a longer primary entity name. | BLACK WORLD COLLEGE OF HAIR DESIGN<br>BLACK WORLD COLLEGE OF HAIR DESIGN |
| E035 | Name without business words has typos and sounds like | The entity names match with a Character Match Percentage of between 80 and 99 after number cardinal and ordinal standardization, and after common company prefixes, suffixes and other words are removed. The first word of each name has the same 4-character Metaphone key. | BOURNE CHIROPRACTIC LTD<br>BOURNE CHIROPRACTIC LTD |

| Group Code | Name Matching Rule | Summary of Rule Logic | Example Matching Data |
|---|---|---|---|
| E036 | Name without suffixes contains with typos and multiple names in common | The entity names are a 'Contains' match where each word matches with a Character Match Percentage of 80 or more after number cardinal and ordinal standardization, and after common company prefixes and suffixes are removed. There are at least two significant words (not common business words) that match. | MEDICAB OF METRO NEW ORLEANS<br>MEDICAB OF METRO NEW ORLEANS |
| E037 | Name without suffixes contains, similar, and multiple words in common | The entity names are a Contains match and the Word Edit Distance is no more than one between the names (where each word matches with a Character Match Percentage of 80 or more), after number cardinal and ordinal standardization, and after common company prefixes and suffixes are removed. There are at least two significant words (not common business words) that match with a Character Match Percentage of 80 or more. | GROSVENOR NURSING SERVICES<br>GROSVENOR NURSING SERVICES |

| Group Code | Name Matching Rule | Summary of Rule Logic | Example Matching Data |
|---|---|---|---|
| E038 | Original script name has additional names with typos | All names in one original script name field must be fully contained within the other field, provided there are at least two names (all of which have an 80%+ Character Match Percentage) in each field. | НИАЭП ОАО<br>НИАЭП ОАО |
| E039 | Name has additional words and sounds like | All words in the shorter entity name exist in the longer entity name (in order) after number cardinal and ordinal standardization. | ATRIUM INCORPORATORS WORLDWIDE LTD<br>ATRIUM INCORPORATORS WORLDWIDE LTD |
| E040 | Name has additional words with typos and sounds like | All words in the shorter entity name match in the longer entity name (in order) with a Character Match Percentage of 80 or more after number cardinal and ordinal standardization. The first word of each name has the same 4-character Metaphone key. | BRILLIANT GENERAL BUILDING CONTRACTOR LTD<br>BRILLIANT GENERAL BUILDING CONTRACTOR LTD |

| Group Code | Name Matching Rule | Summary of Rule Logic | Example Matching Data |
|---|---|---|---|
| E041 | Name without business words loose match and full name sounds like | The entity names match with a Character Match Percentage of between 60 and 79 after number cardinal and ordinal standardization, and after common company prefixes, suffixes and other words are removed. The names have the same Metaphone key. | BRC<br>BRC |
| E042 | Name without business words contains, sounds like, and residual token in common | The entity names are a Contains match after number cardinal and ordinal standardization, and after common company prefixes, suffixes and other words are removed. There is at least one significant word (not a common business word, and English dictionary word or a very common word in Watchlist name data) in common between the two names. The first word of each name has the same 4-character Metaphone key. The list name is not an acronym alias of a longer primary entity name. | GARLICK HELICOPTERS INC<br>GARLICK HELICOPTERS INC |

| Group Code | Name Matching Rule | Summary of Rule Logic | Example Matching Data |
|---|---|---|---|
| E043 | Name without business words contains, sounds like, and substring in common | The entity names match with a Character Match Percentage of between 60 and 79 after number cardinal and ordinal standardization, and after common company prefixes, suffixes and other words are removed. The names have the same Metaphone key. | NATIONWIDE SECRETARIAL<br>SERVICES LTD<br>NATIONWIDE SECRETARIAL<br>SERVICES LTD |
| E044 | Name without suffixes starts with | The entity names are a Starts With match after number cardinal and ordinal standardization, and after common company prefixes and suffixes are removed. The list name is not an acronym alias of a longer primary entity name. | MARK<br>MARK |
| E045 | Name without business words has additional words and sounds like | All words in the shorter entity name exist in the longer entity name (in order) after number cardinal and ordinal standardization, and after common company prefixes, suffixes and other words are removed. The first word of each name has the same 4- character Metaphone key. The list name is not an acronym alias of a longer primary entity name. | CROSS<br>CROSS |

| Group Code | Name Matching Rule | Summary of Rule Logic | Example Matching Data |
|---|---|---|---|
| E046 | Name without business words has additional words with typos and sounds like | All words in the shorter entity name match with a Character Match Percentage of 80 or more in the longer entity name (in order) after number cardinal and ordinal standardization,and after common company prefixes,suffixes and other words are removed.The first word of each name has the same 4-character Metaphone key. The list name is not an acronym alias of a longer primary entity name. | GROSVENOR NURSING SERVICES<br>GROSVENOR NURSING SERVICES |
| E047 | Name without business words contains and sounds like | The entity names are a Contains match after number cardinal and ordinal standardization, and after common company prefixes, suffixes and other words are removed. The first word of each name has the same 4-character Metaphone key. The list name is not an acronym alias of a longer primary entity name. | ENRON METALS BROKERS LTD<br>ENRON METALS BROKERS LTD |
| E048 | Name without suffixes starts with and allows acronyms | The entity names are a Starts With match after number cardinal and ordinal standardization, and after common company prefixes and suffixes are removed. | ANR<br>ANR |

| Group Code | Name Matching Rule | Summary of Rule Logic | Example Matching Data |
|---|---|---|---|
| E049 | Name without suffixes contains, significant overlap and multiple words in common | The entity names are a Contains match, there are at least two words that match<br>with a Character Match Percentage of 80 or more, and the two entity names<br>match with a Word Match Percentage of 50 or more after number cardinal and ordinal standardization, and after common company prefixes and suffixes are removed. | ALTAR<br>ALTAR |
| E050 | Name contains with typos and multiple words in common | The entity names are a 'Contains' match where each word matches with a Character Match Percentage of 80 or more after number cardinal and ordinal standardization, and after common company prefixes and suffixes are removed. There are at least two words (not prefixes or suffixes) that match. | CHERRY<br>CHERRY |

Table 26 describes the Loose Entity Matching Rules.

**Table 26: Loose Entity Matching Rules**

| Group Code | Name Matching Rule | Summary of Rule Logic | Example Matching Data |
|---|---|---|---|
| E0001 | Name exact | The entity names match exactly after number cardinal and ordinal standardization | NOVEMBER SEVENTEEN<br>NOVEMBER 17 |
| E0002 | Original script name in any order | All words in the Original Script Names match exactly, in any order. | ОАО НИАЭП<br>ОАО НИАЭП |
| E0003 | Original script name with typos | The Original Script Names match with a Character Match Percentage of 80% or more. | ΕΠΑΝΑΣΤΑΤΙΚΉ ΑΡΙΣΤΕΡΆ<br>ΕΠΑΝΑΣΤΑΤΙΚΉ ΑΡΙΣΤΕΡΆ |
| E0004 | Original script name in any order with typos | All words in the original script name match with a Character Match Percentage of 80 or more, in any order. | ΕΠΑΝΑΣΤΑΤΙΚΉ ΑΡΙΣΤΕΡΆ<br>ΕΠΑΝΑΣΤΑΤΙΚΉ ΑΡΙΣΤΕΡΆ |
| E0005 | Original script name has additional names | All words in the shorter original script name match in the longer original script name (in order), and there are at least two matching words. | ВЪОРЪЖЕНА ИСЛЯМСКА ГРУПА<br>ВЪОРЪЖЕНА ИСЛЯМСКА ГРУПА |
| E0006 | Original script name has additional names with typos | All words in the shorter original script name match in the longer original script name (in order) with a Character Match Percentage of 80 or more, and there are at least two matching words. | ВЪОРЪЖЕНА ИСЛЯМСКА ГРУПА<br>ВЪОРЪЖЕНА ИСЛЯМСКА ГРУПА |

| Group Code | Name Matching Rule | Summary of Rule Logic | Example Matching Data |
|---|---|---|---|
| E0007 | Name without business words contains, sounds like, and residual token in common | The entity names are a Contains match after number cardinal and ordinal standardization, and after common company prefixes, suffixes and other words are removed. There is at least one significant word (not a common business word, and English dictionary word or a very common word in Watchlist name data) in common between the two names. The first word of each name has the same 4-character Metaphone key. The list name is not an acronym alias of a longer primary entity name. | HENDERSON EQUITY PARTNERS GP LTD<br>HENDERSON EQUITY PARTNERS GP LTD |
| E0008 | Name without business words contains, sounds like, and substring in common | The entity names are a Contains match and there is a common substring at least 8 characters in length after number cardinal and ordinal standardization, and after common company prefixes, suffixes and other words are removed. The first word of each name has the same 4-character Metaphone key. The list name is not an acronym alias of a longer primary entity name. | HAMILTON NEWS<br>HAMILTON NEWS |

| Group Code | Name Matching Rule | Summary of Rule Logic | Example Matching Data |
|---|---|---|---|
| E0009 | Name without suffixes starts with | The entity names are a Starts With match after number cardinal and ordinal standardization, and after common company prefixes and suffixes are removed. The list name is not an acronym alias of a longer primary entity name. | JACOB<br>JACOB |
| E0010 | Name without business words has additional words and sounds like | All words in the shorter entity name exist in the longer entity name (in order) after number cardinal and ordinal standardization, and after common company prefixes, suffixes and other words are removed. The first word of each name has the same 4-character Metaphone key. The list name is not an acronym alias of a longer primary entity name. | IDEAL SOLUTION ESTATES<br>MANAGEMENT LTD<br>IDEAL SOLUTION ESTATES<br>MANAGEMENT LTD |
| E0011 | Name without business words has additional words with typos and sounds like | All words in the shorter entity name match with a Character Match Percentage of 80 or more in the longer entity name (in order) after number cardinal and ordinal standardization, and after common company prefixes, suffixes and other words are removed. The first word of each name has the same 4-character Metaphone key. The list name is not an acronym alias of a longer primary entity name. | AVANT GARD LTD<br>AVANT GARD LTD |

| Group Code | Name Matching Rule | Summary of Rule Logic | Example Matching Data |
|---|---|---|---|
| E0012 | Name without business words contains and sounds like | The entity names are a Contains match after number cardinal and ordinal standardization, and after common company prefixes, suffixes and other words are removed. The first word of each name has the same 4-character Metaphone key. The list name is not an acronym alias of a longer primary entity name. | MOREX TRADING LTD<br>MOREX TRADING LTD |
| E0013 | Name without suffixes starts with and allows acronyms | The entity names are a Starts With match after number cardinal and ordinal standardization, and after common company prefixes and suffixes are removed. | INTERTRADE CLASSIC LTD<br>INTERTRADE CLASSIC LTD |
| E0014 | Name without suffixes contains, significant overlap and multiple words in common | The entity names are a Contains match, there are at least two words that match with a Character Match Percentage of 80 or more, and the two entity names match with a Word Match Percentage of 50 or more after number cardinal and ordinal standardization, and after common company prefixes and suffixes are removed. | INTERTRADE CLASSIC LTD<br>INTERTRADE CLASSIC LTD |

| Group Code | Name Matching Rule | Summary of Rule Logic | Example Matching Data |
|---|---|---|---|
| E0015 | Name contains with typos and multiple words in common | The entity names are a 'Contains' match where each word matches with a Character Match Percentage of 80 or more after number cardinal and ordinal standardization, and after common company prefixes and suffixes are removed. There are at least two words (not prefixes or suffixes) that match. | MOREX TRADING LTD<br>MOREX TRADING LTD |

# 7 Identifier Matching

This chapter discusses the matching webservice used for Identifier matching against the sanctioned Identifier list such as Bad BICs Ref Data. This chapter includes the following topics:

- Identifier
- Cluster
- Match Rules

## 7.1 Identifier

The following identifier is used for matching:

- BIC CODE
- LEI CODE
- Passport Number
- National Idendifier

## 7.2 Cluster

The following cluster is used for BIC matching:

- BIC Code Cluster (BIC CODE)
- LEI CODE

## 7.3 Match Rules

Table 27 describes match rules are used for BIC matching:

**Table 27: Name Normalization**

| Group Code | Matching Rule | Summary of Rule Logic | Example Matching Data |
|---|---|---|---|
| B001 | Exact bank identification code (BIC) match | Exact match of Input BIC size is of 8 characters with reference data BIC is of 8 characters<br>OR<br>Input BIC size is of 11 characters with reference data BIC is of 11 characters | DEUTDEFFXXX<br>DEUTDEFFXXX<br><br>DEUTDEFF<br>DEUTDEFF |
| B0002 | Bank identification code (BIC) match contains | Matched input BIC size can be either 8 or 11 characters with reference data BIC of 8 or 11 characters | deutdeffXXX<br>DEUTDEFF<br><br>DEUTDEFF<br>DEUTDEFFXXX |
| B0003 | Exact Legal Entity Identification (LEI) Match | It is a unique 20-character alphanumeric code | HB7FFAZI0OMZ8PP8OE2 |
| B0004 | Legal Entity Identification (LEI) Contains | It is a unique 20-character alphanumeric code | HB7FFAZI0OMZ8PP8OE2 |
| B0005 | Exact Passport Number Match | Unique 9 alpha numeric value | H079XXXXXXXX |
| B0006 | Passport Number Contains | Unique 9 alpha numeric value | H079XXXXXXXX |

| Group Code | Matching Rule | Summary of Rule Logic | Example Matching Data |
|---|---|---|---|
| B0007 | Exact Identification Match | National ID<br>Driving Licence No.<br>National Tax No.<br>Company Identification No.<br>DUNS Number<br>International Maritime Organization (IMO) Ship No.<br>Manufacturer's Serial Number (MSN) Aircraft No.<br>International Securities Identification Number (ISIN) | |
| B0008 | Identification Contains | National ID<br>Driving Licence No.<br>National Tax No.<br>Company Identification No.<br>DUNS Number<br>International Maritime Organization (IMO) Ship No.<br>Manufacturer's Serial Number (MSN) Aircraft No.<br>International Securities Identification Number (ISIN) | |

The reference data is 'Bad BICs Ref Data' For information on the reference data for watchlists, see Oracle Financial Services Transaction Filtering Administration Guide.

# 8 Country and City Matching

This chapter discusses the matching webservice that is used for country and city matching against the sanctioned country and city list. The sanctioned country and city reference data names are Blacklisted Country Ref Data and Blacklisted Cities Ref Data respectively. The blacklisted city information is extracted from the party address field of the input message and the blacklisted country extraction happens from different fields such as Country from party address, Country ISO code from BIC, Country ISO from IBAN, and Country confirmation from party location.

This chapter includes the following topics:


- Identifiers
- Clusters
- Match Rules


## 8.1 Identifiers

Table 28 describes the identifiers are used for Country and City matching:.

**Table 28: Identifiers used for Country and City matching**

| Identifier Description | Standard prepared attribute name |
|---|---|
| Country | Country Code (ISO) |
| City | City |
| Country ISO3 Code | Country ISO3 Code |
| Country ISO2 Code | Country ISO2 Code |
| City ISO Code | City ISO Code |
| Country Record ID | Country Record ID |
| City Record ID | City Record ID |
| Country Fuzzy | Country Fuzzy |
| City Fuzzy | City Fuzzy |

## 8.2 Clusters

The following clusters are used for Country and City matching:

- Country: This cluster uses the Country to generate cluster keys.
- City: This cluster uses the City to generate cluster keys.
- Country ISO3 Code: This cluster uses the Country ISO3 Code to generate cluster keys.
- Country ISO2 Code: This cluster uses the Country ISO2 Code to generate cluster keys.
- City ISO Code: This cluster uses the City ISO Code to generate cluster keys.
- Country Fuzzy: This cluster uses the Country Fuzzy to generate cluster keys.

- City Fuzzy: This cluster uses the City Fuzzy to generate cluster keys.
- Country Synonyms: This cluster uses the Country Synonyms to generate cluster keys.
- City Synonyms: This cluster uses the City Synonyms to generate cluster keys.

## 8.3 Match Rules

Table 28 describes the match rules are used for country and city matching:

**Table 29: Match Rules Used for Country and City Matching**

| Matching Rule | Summary of Rule Logic | Example Matching Data |
|---|---|---|
| Exact country name | Matches country name exactly | UNITED STATES<br>UNITED STATES |
| Country name in any order | Matches country name in any order | STATES UNITED<br>UNITED STATES |
| Exact County Synonyms name | Matches country name with country synonyms exactly | NORTH KOREA<br>DPRK |
| Exact ISO country code (ISO 3) | Matches ISO 3 country code exactly | USA<br>USA |
| Exact ISO country code (ISO 2) | Matches ISO 2 country code exactly | KP<br>KP |
| Country name similar and sounds like | Matches country name which is similar and sounds like the matched country | CONGO<br>DR CONGO |
| Country name at least one word matching | Matches country name with at least one word matching | UNITED<br>STATES |
| Exact City Name with Country | Matches country with city name exactly | SINGAPORE<br>SINGAPORE |
| Exact City name | Matches city name exactly | ABU AL FULUS<br>AL ABU FULUS |
| City name in any order | Matches city name in any order | AL FULUS ABU<br>ABU AL FULUS |
| Exact ISO city code (ISO) | Matches city name with country synonyms | ALF<br>ALF |
| Exact City Synonyms name | Matches city with city synonyms exactly | CHENNAI<br>MADRAS |
| City name similar and sounds like | Matches city name which is similar and sounds like the matched city | SINGAPORE<br>SINGAPORE |
| City name at least one word matching | Matches city name with at least one word matching | ABU<br>FULUS |

# 9 Narrative Matching

This chapter discusses the matching webservice that is used for free text or narrative field screening such as remittance information, reference fields, and sender to receiver information. This particular webservice screens individual, entity, vessel and aircraft names (if present) against different records such as watchlist records, country against sanctioned country reference data, city against sanctioned city reference data, BIC against sanctioned BIC reference data and key words against Stopkeywords reference data or StopKeywords Ref Data.

This chapter includes the following topics:


- Identifiers
- Clusters
- Match Rules
- Enabling a Match Rule and Web Service


## 9.1 Identifiers

Table 30 describes the identifiers are used for Narrative matching:

**Table 30: Identifiers Used for Narrative Matching**

| Identifier Description | Standard prepared attribute name |
|---|---|
| dnFullName | dnFullName |
| dnGivenName | dnGivenName |
| dnINDOriginalScriptName | dnINDOriginalScriptName |
| dnEntityOriginalScriptName | dnEntityOriginalScriptName |
| dnEntityName | dnEntityName |
| Country | Country |
| City | City |
| BIC | BIC |
| Country ISO3 Code | Country ISO3 Code |
| Country ISO2 Code | Country ISO2 Code |
| City ISO Code | City ISO Code |
| Country Fuzzy | Country Fuzzy |
| City Fuzzy | City Fuzzy |
| Country Synonyms | Country Synonyms |
| City Synonyms | City Synonyms |
| Vessel | Vessel |
| dnAircraftIndicator | dnAircraftIndicator |

## 9.2 Clusters

The following clusters are used for Narrative matching:

- Full Name: This cluster uses the Full Name to generate cluster keys. Full Name Cluster allows new records to be matched against existing records in a system.
- Given Name: This cluster uses the Given Name to generate cluster keys. Given Name Cluster allows new records to be matched against existing records in a system.
- IND Script Name: This cluster uses the IND Script Name to generate cluster keys. IND Script Name Cluster allows new records to be matched against existing records in a system.
- Entity Name: This cluster uses the Entity Name to generate cluster keys. Entity Name Cluster allows new records to be matched against existing records in a system.
- ENT Script Name: This cluster uses the ENT Script Name to generate cluster keys. ENT Script Name Cluster allows new records to be matched against existing records in a system.

## 9.3 Match Rules

Table 31 describes the match rules used for Country, city, bad BIC and stop key words advanced match processing:

**Table 31: Identifiers Used for Advanced Match Processing**

| Matching Rule | Summary of Rule Logic | Example Matching Data |
|---|---|---|
| Exact bank identification code (BIC) match | Exact match of Input BIC size is of 8 characters with reference data BIC is of 8 characters<br>OR<br>Input BIC size is of 11 characters with reference data BIC is of 11 characters | DEUTDEFFXXX<br>DEUTDEFFXXX<br><br>DEUTDEFF<br>DEUTDEFF |
| Bank identification code (BIC) contains | Matches input BIC size can be either 8 or 11 characters with reference data BIC of 8 or 11 characters | deutdeffXXX<br>DEUTDEFF<br><br>DEUTDEFF<br>DEUTDEFFXXX |
| Exact country name | Matches country name exactly | UNITES STATES<br>UNITES STATES |
| Exact county synonym | Matches country name with country synonyms | NORTH KOREA<br>DPRK |
| Country name in any order | Matches country name in any order | STATES UNITED<br>UNITED STATES |
| Exact ISO country code (ISO 3) | Matches ISO 3 country code exactly | USA<br>USA |
| Exact ISO country code (ISO 2) | Matches ISO 2 country code exactly | KP<br>KP |
| Exact city name | Matches city name exactly | ABU AL FULUS |
| Exact city synonym | Matches against city synonym column of reference data | ABU AL<br>AL FULUS |
| City name in any order | Matches city name in any order | AL FULUS ABU<br>ABU AL FULUS |
| Exact ISO city code (ISO) | Matches city name with country synonyms | ALF<br>ALF |
| Exact Stop Keywords | Matches exact stop words | EXPLOSIVE<br>EXPLOSIVE |

Table 32 describes the match rules are used for Entity and individual match processing:

**Table 32: Identifiers Used for Entity and individual Match Processing**

| Matching Rule | Summary of Rule Logic | Example Matching Data- Given Names | Example Matching Data- Given Names |
|---|---|---|---|
| Exact Port name | Matches the exact Port name | Jawaharlal Nehru Port | |
| Exact Port synonym name | Matches the exact Port synonym name | JNU Port | |
| Exact Port Code | Matches the exact Port code | INVTZ1 | |
| Port name in any order | Matches the port name in any order | Jawaharlal Nehru Port<br>Nehru Jawaharlal Port | |
| Port name at least one word matching | Matches the Port name with at least one word matching | Jawaharlal Port<br>Nehru Port | |
| Exact goods name | Matches the exact goods name | Crude oil | |
| Exact goods synonym name | Matches the exact goods synonym name | Oil | |
| Goods name in any order | Matches the goods name in any order | Crude Oil<br>Oil Crude | |
| Goods name at least one word matching | Matches the goods name at least one word matching | Crude<br>Oil | |
| Exact Individual Full Name | Given names and family name match exactly | JOSEPH<br>JOSEPH | TSANGA<br>T'SANGA |
| Exact Individual Given Name | Given name matches exactly | JOSEPH<br>JOSEPH | MIKE MICHAEL<br>MICHAEL |
| Exact Individual Original Script Name | The original script Name fields match exactly | ΕΠΑΝΑΣΤΑΤΙΚΉ ΑΡΙΣΤΕΡΆ<br>ΕΠΑΝΑΣΤΑΤΙΚΉ ΑΡΙΣΤΕΡΆ | |
| Vessel part-standardized name exact | The part-standardized entity name matches the name of a listed vessel exactly | DYNASTY<br>DYNASTY | |

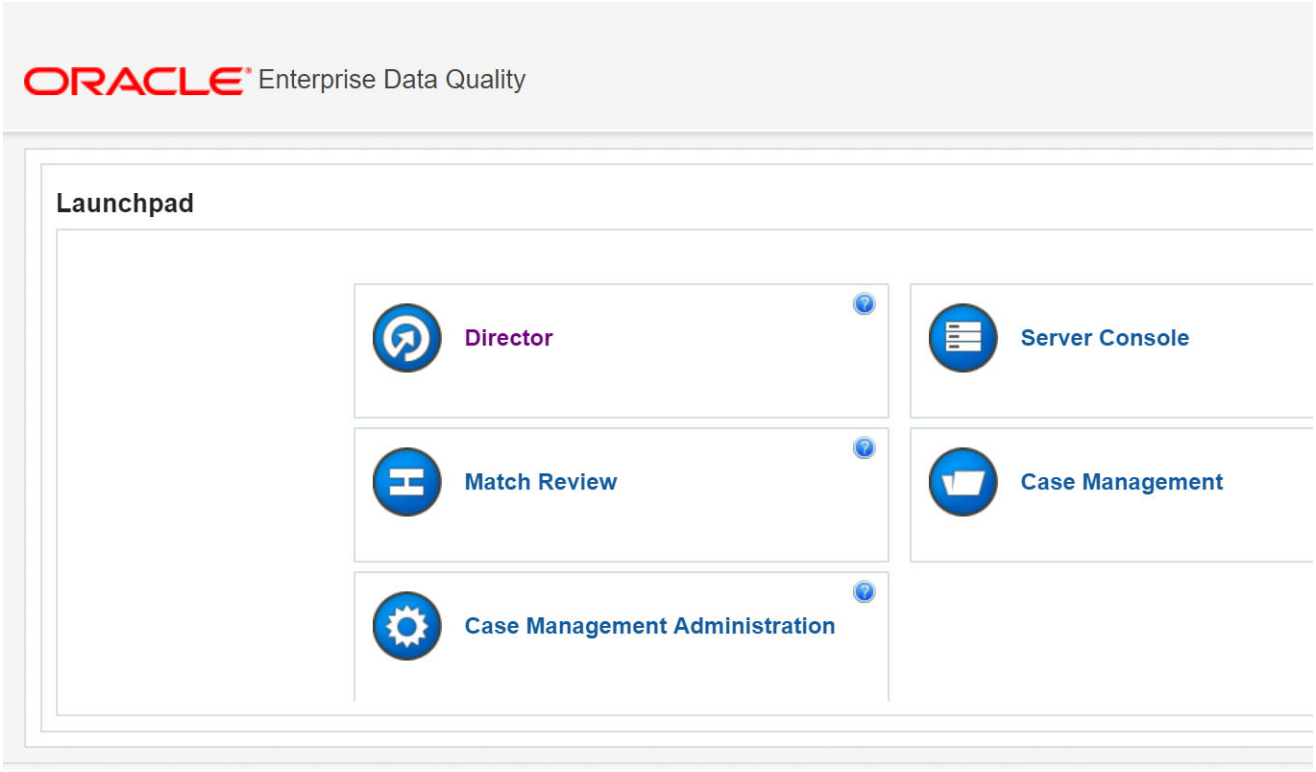
| Matching Rule | Summary of Rule Logic | Example Matching Data- Given Names | Example Matching Data- Given Names |
|---|---|---|---|
| Vessel name exact | The entity name matches the name of a listed vessel after number cardinal and ordinal standardization | 4TH OCEAN<br>FOURTH OCEAN | |
| Aircraft part-standardized name exact | The part-standardized entity name matches the name of a listed aircraft exactly | EP-GOM<br>EP-GOM | |
| Aircraft name exact | The entity name matches the name of a listed aircrafts after number cardinal and ordinal standardization | 4TH YK-AYF<br>FOURTH YK-AYF | |
| Exact Entity Name | The entity names match exactly<br>after number cardinal and ordinal<br>Standardization | NOVEMBER 17<br>NOVEMBER SEVENTEEN | |
| Exact Entity Original Script Name | The original script names match exactly | НИАЭП ОАО<br>НИАЭП ОАО | |
| Standardized given name | Matches the standardized given name | Mohammed | |
| Name in any order | Matches the full name in any order | Mohammed Ali<br>Ali Mohammed | |
| Original script name in any order | Matches the original script name in any order | НИАЭП ОАО<br>ОАО НИАЭП | |

## 9.4 Enabling a Match Rule and Web Service

By default, only stop keywords match rules are enabled. To enable any other match rules:

1. Open the Director menu in EDQ.

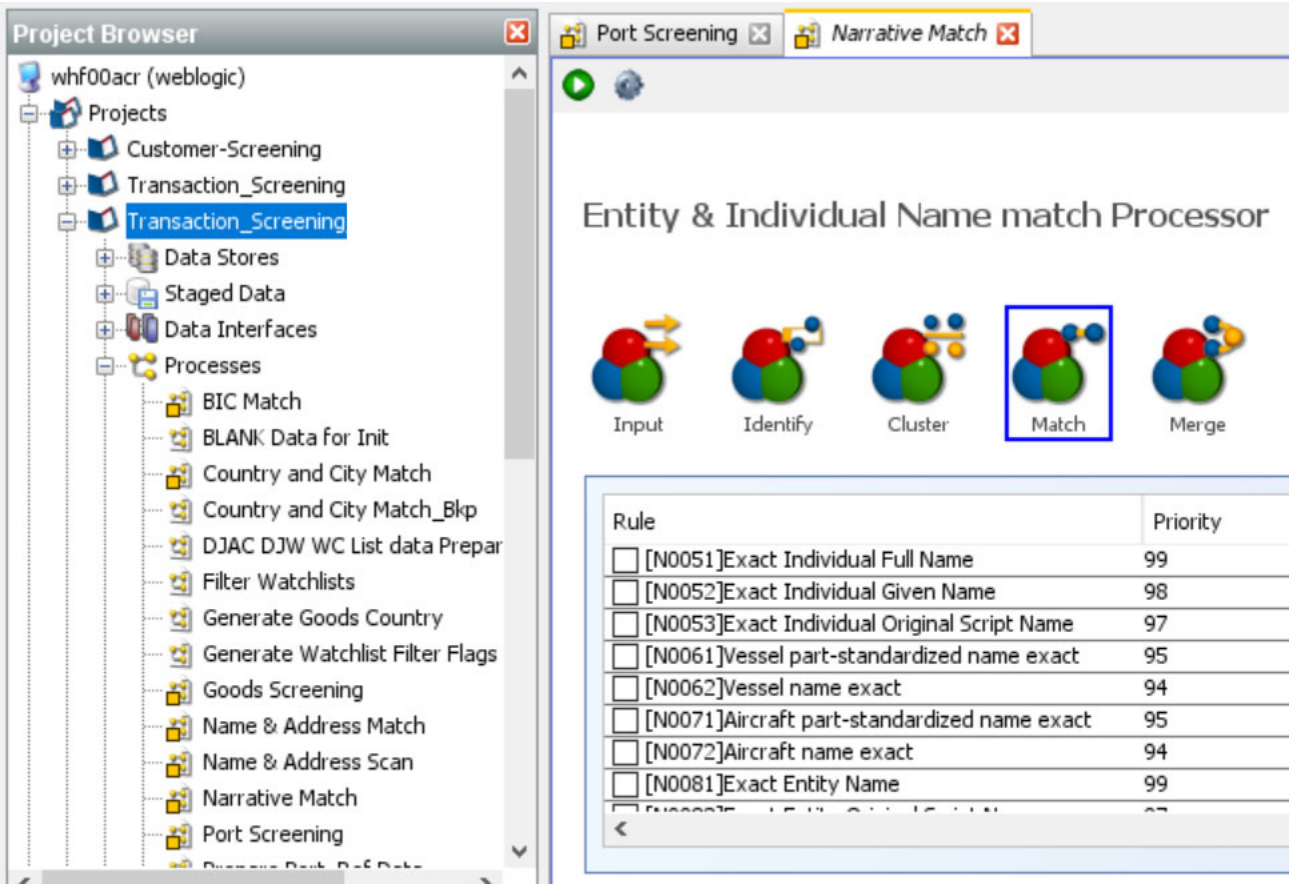
**Figure 3: Director menu in EDQ**




2. In the Project Browser pane, expand the **Transaction_Screening** project.
3. Expand the **Processes** node and double-click **Narrative Match**.

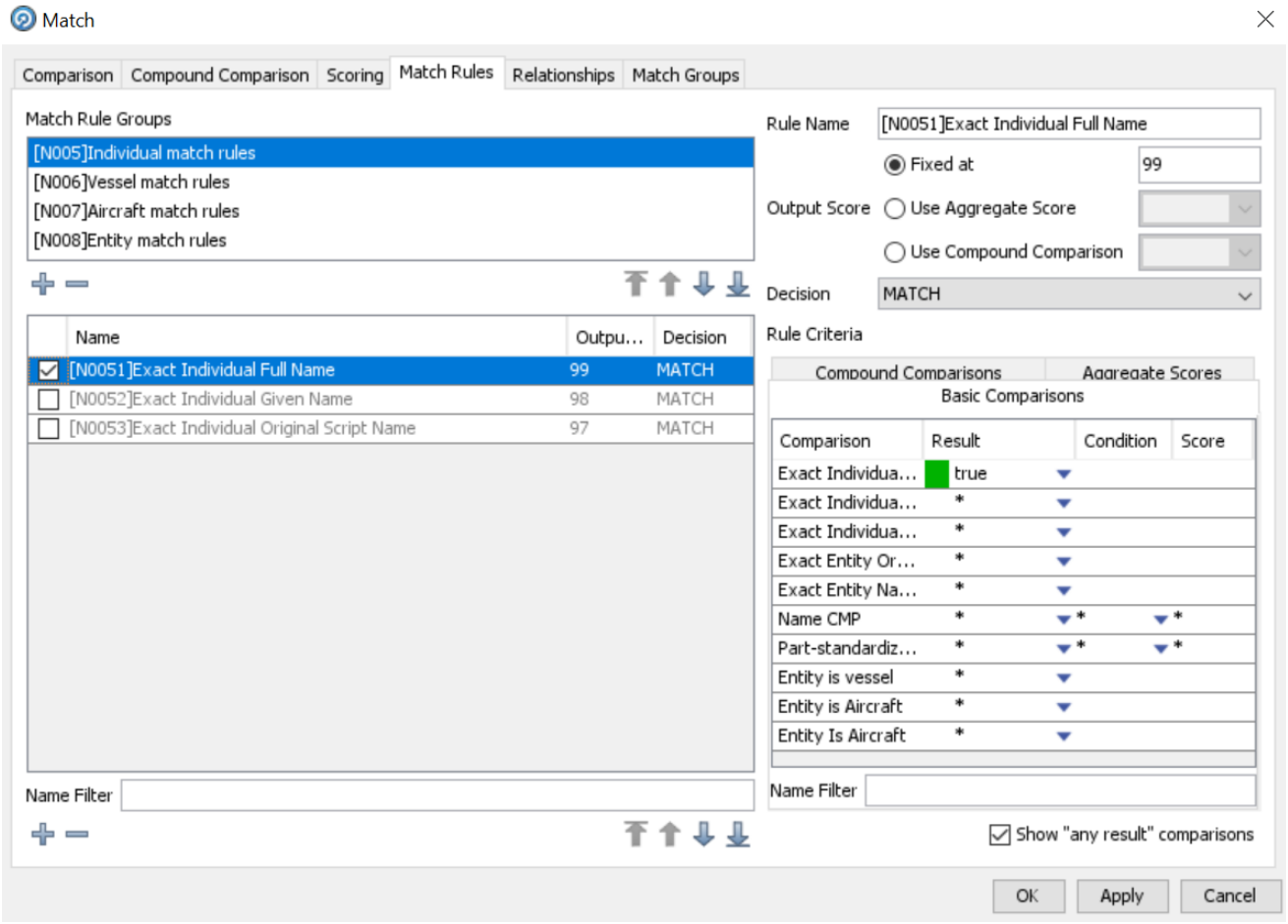
   In the Narrative Match tab, all match rules are displayed.

**Figure 4: Narrative Match Tab**




4. Double-click **Match**.
5. Click the **Match Rules** tab.
6. In the Match window, select the match rule that you want to enable and click **Apply**.

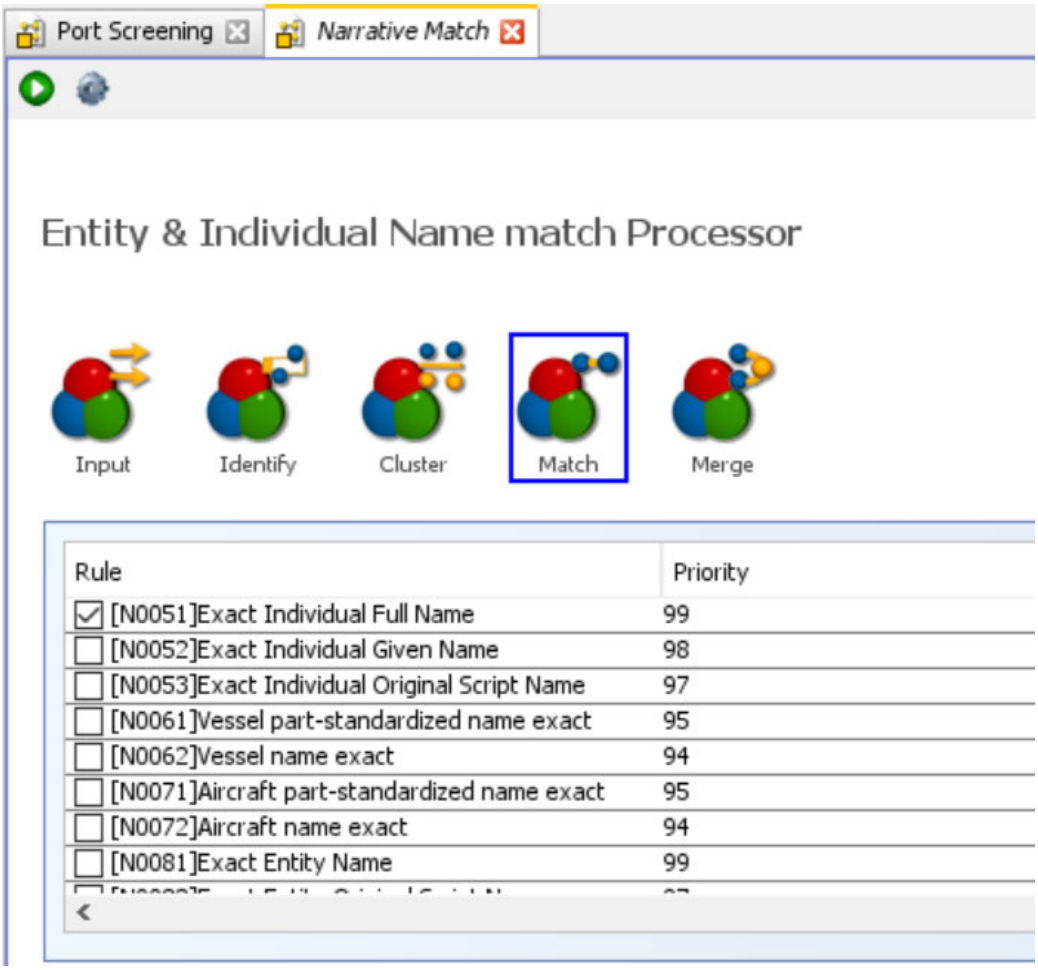
**Figure 5: Match Window**




7. Click **OK**.

   The match rule that you have enabled is displayed with a tick next to it in the Narrative Match tab.

**Figure 6: Narrative Match Tab**

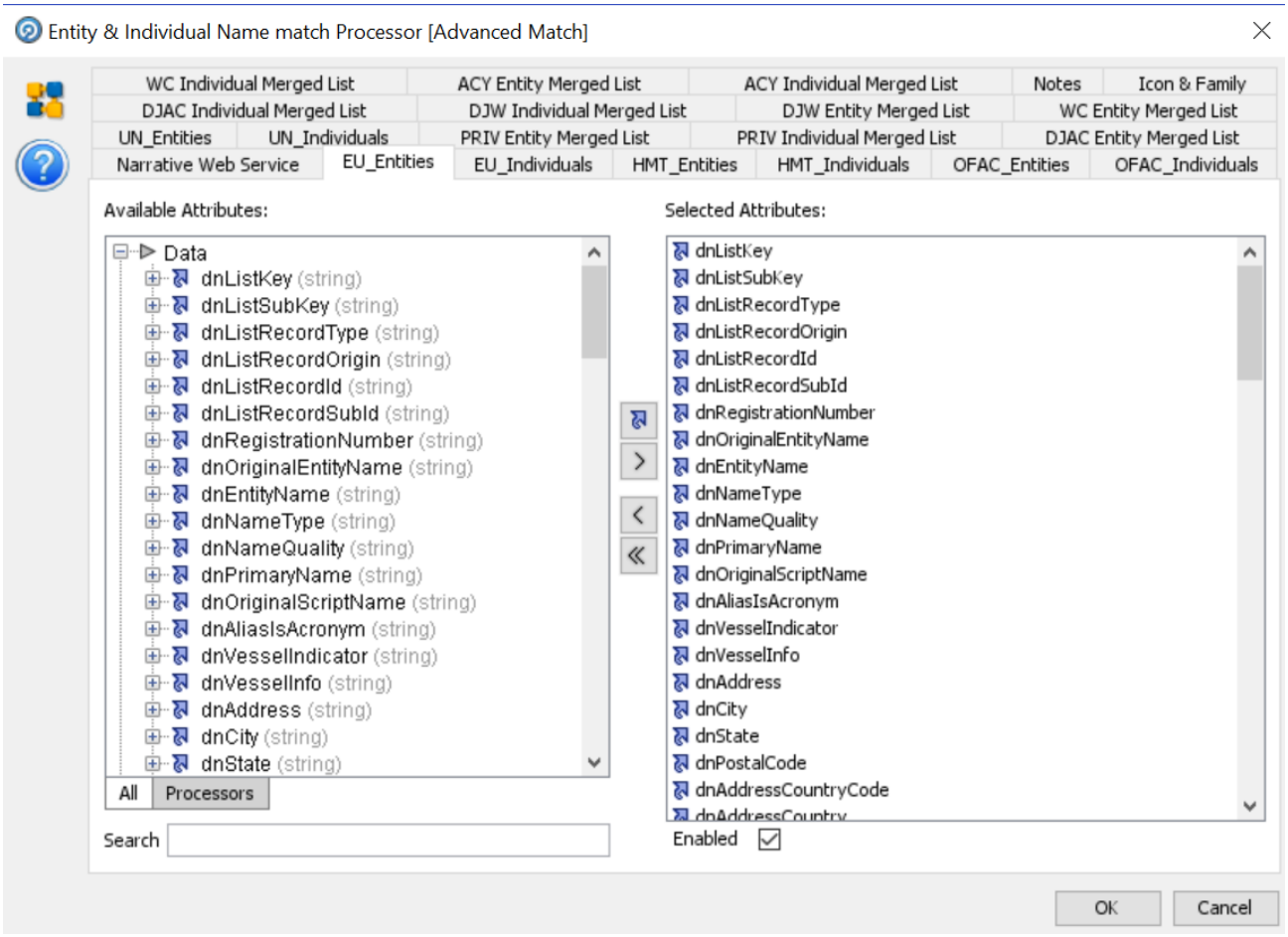



To disable the match rule, deselect the match rule, click **Apply**, and click **OK**.

To view the match rules for any web services which are disabled:

1. In the Narrative Match tab, Double-click **Input**.
2. In the **Entity & Individual Name Match Processor** window, select the web service which you want to enable.

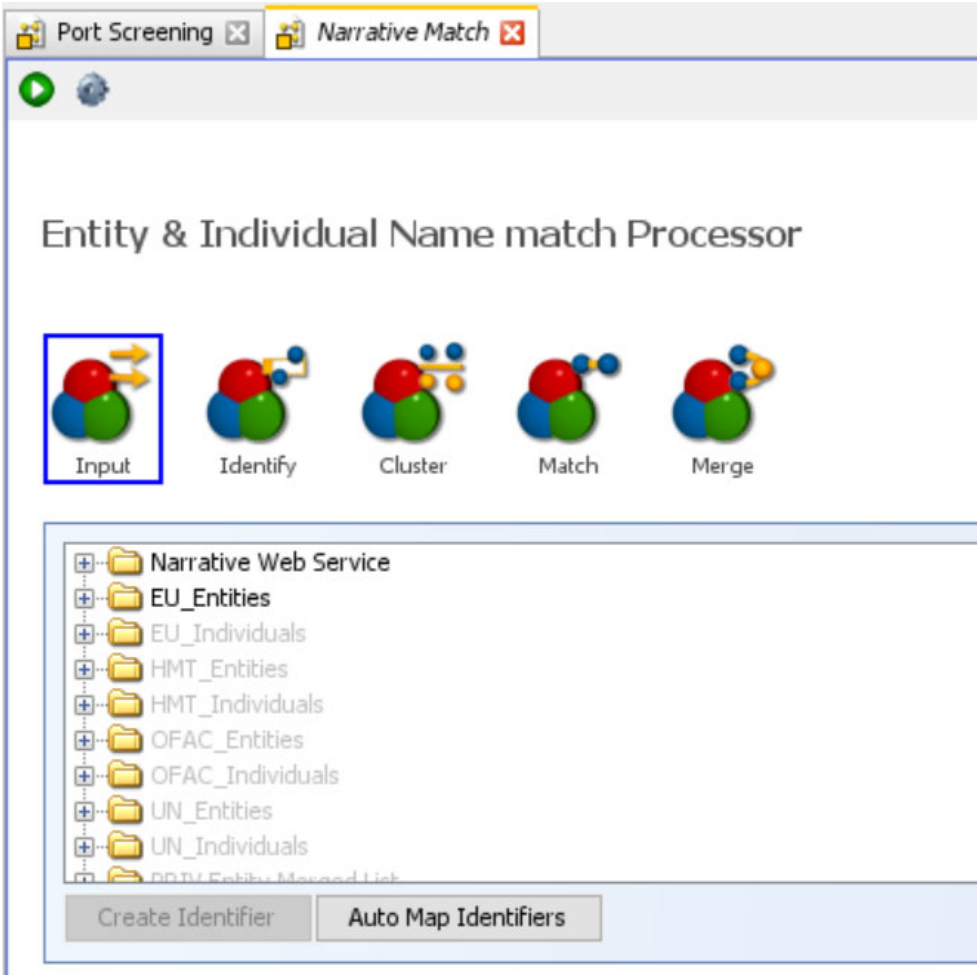
**Figure 7: Entity & Individual Name Match Processor Window**




3. Select the **Enabled** check box.
4. Click **OK**.

   The web service that you have enabled is highlighted in the Narrative Match tab.

**Figure 8: Narrative Match Tab**




To disable the web service, deselect the **Enabled** check box and click **OK**.

# 10 Trade Goods Matching

This chapter discusses This matching webservice is used to extract the trade goods name and match the name against the prohibited goods list.

This list provides country-wise data.

This chapter includes the following topics:

- Identifiers
- Clusters
- Match Rules

## 10.1 Identifiers

Table 33 describes identifiers are used for Trade Goods matching:

**Table 33: Identifiers Used for Trade Goods Matching**

| Identifier Description | Standard prepared attribute name |
|---|---|
| Goods Name | Goods Name |
| Country From | Country From |
| Country To | Country To |
| Goods Synonyms | Goods Synonyms |

## 10.2 Clusters

The following clusters are used for Trade Goods matching:

- Goods Name: This cluster uses the Goods Name to generate cluster keys. Goods Name Cluster allows new records to be matched against existing records in a system.
- Goods Synonyms: This cluster uses Goods Synonyms to generate cluster keys. Goods Synonyms Cluster allows new records to be matched against existing records in a system.

## 10.3 Match Rules

Table 34 describes the match rules used for Trade Goods match processing:

**Table 34: Identifiers Used for Trade Goods Matching**

| Matching Rule | Summary of Rule Logic | Example Matching Data- Goods Name | Example Matching Data- Import Country ISO from | Example Matching Data- Export Country ISO to |
|---|---|---|---|---|
| Exact goods name, Import Country ISO from, Export Country ISO to | Exact Match against goods name, Country ISO (Import Prohibition from) AND Country ISO (Export Prohibition to) columns of prohibited goods reference data | Crude Oil | CH | DN |
| Exact goods name, Import Country ISO from | Exact Match against goods name, Country ISO (Import Prohibition from) columns of prohibited goods reference data | Crude Oil | CH | CR |
| Exact goods name, Export Country ISO to | Exact Match against goods name, Country ISO (Export Prohibition to) columns of prohibited goods reference data | Crude Oil | VE | UG |
| Exact goods name | Exact Match against goods name column of prohibited goods reference data | Crude Oil | VE | UG |
| Exact goods synonym name, Import Country ISO from, Export Country ISO to | Exact Match against goods synonym name, Country ISO (Import Prohibition from) AND Country ISO (Export Prohibition to) columns of prohibited goods reference data | Oil | TW | GB |

| Matching Rule | Summary of Rule Logic | Example Matching Data- Goods Name | Example Matching Data- Import Country ISO from | Example Matching Data- Export Country ISO to |
|---|---|---|---|---|
| Exact goods synonym name, Import Country ISO from | Exact Match against goods synonym name, Country ISO (Import Prohibition from) columns of prohibited goods reference data | Oil | TW | US |
| Exact goods synonym name, Export Country ISO to | Exact Match against goods synonym name, Country ISO (Export Prohibition to) columns of prohibited goods reference data | Oil | UK | UM |
| Exact goods synonym name | Exact Match against goods synonym name column of prohibited goods reference data | Oil | UK | UM |
| Goods name in any order, Import Country ISO from, Export Country ISO to | Matches city ISO code exactly | Crude Oil<br>Oil Crude | UR | SS |
| Goods name in any order, Import Country ISO from | Match against goods name in any order exact Country ISO (Import Prohibition from) columns of prohibited goods reference data | Crude Oil<br>Oil Crude | UR | SS |

| Matching Rule | Summary of Rule Logic | Example Matching Data- Goods Name | Example Matching Data- Import Country ISO from | Example Matching Data- Export Country ISO to |
|---|---|---|---|---|
| Goods name in any order, Export Country ISO to | Match against goods name in any order, exact Country ISO (Export Prohibition to) columns of prohibited goods reference data | Crude Oil<br>Oil Crude | VN | CH |
| Goods name in any order | Match against goods name column in any order of prohibited goods reference data | Crude Oil<br>Oil Crude | VN | CH |
| Goods name at least one word matching, Import Country ISO from, Export Country ISO to | Word match count is >0 against goods name in reference data, exact Country ISO (Import Prohibition from) AND exact Country ISO (Export Prohibition to) | Crude<br>Oil | ZA | SW |
| Goods name at least one word matching, Import Country ISO from | Word match count is >0 against goods name in reference data, exact Country ISO (Import Prohibition from) | Crude<br>Oil | ZA | SW |
| Goods name at least one word matching, Export Country ISO to | Word match count is >0 against goods name in reference data, exact Country ISO (Export Prohibition to) | Crude<br>Oil | TW | GB |
| Goods name at least one word matching | Word match count is >0 against goods name in reference data | Crude<br>Oil | TH | TJ |

# 11 Trade Port Matching

This chapter discusses the matching webservice that is used to extract the trade port name and match it to the country while screening. This is done because two countries may not have the same port name, so this avoids duplication. The port name is also matched against the sanctioned port reference data.

This chapter includes the following topics:


- Identifiers
- Clusters
- Match Rules


## 11.1 Identifiers

Table 35 describes the identifiers used for Trade Port matching:

**Table 35: Identifiers Used for Trade Port Matching**

| Identifier Description | Standard prepared attribute name |
|---|---|
| Port Name | Port Name |
| Port Code | Port Code |
| Port Synonyms | Port Synonyms |
| Port In Any Order | Port In Any Order |
| Country | Country |

## 11.2 Clusters

The following clusters are used for Trade Port matching:

- Port Name: This cluster uses the Port Name to generate cluster keys. Port Name Cluster allows new records to be matched against existing records in a system.
- Port Code: This cluster uses the Port Code to generate cluster keys. Port Code Cluster allows new records to be matched against existing records in a system.
- Port Synonyms: This cluster uses the Port Synonyms to generate cluster keys. Port Synonyms Cluster allows new records to be matched against existing records in a system.
- Port Name In Any Order: This cluster uses the Port Name In Any Order to generate cluster keys. Port Name In Any Order Cluster allows new records to be matched against existing records in a system.
- ISO Country: This cluster uses the ISO Country to generate cluster keys. ISO Country Cluster allows new records to be matched against existing records in a system.

## 11.3 Match Rules

Table 36 describes the match rules used for Country, city, bad BIC and stop key words advanced match processing:

**Table 36: Match Rules Used Advanced Match Processing**

| Matching Rule | Summary of Rule Logic | Example Matching Data- Port Name | Example Matching Data- Country |
|---|---|---|---|
| Exact port name and country (ISO) | match against port name column of reference data along with country name where port should belong to that country | Port of Houston | CH |
| Exact port name only | match against port name column of reference data | Port of Houston | CH |
| Exact port synonym name and country (ISO) | match against port synonym column of reference data along with country name where port should belong to that country | Houston Port | VE |
| Exact port synonym name only | match against port synonym column of reference data | Houston Port | VE |
| Exact port code only | match against port code columns of reference data along with country name where port should belong to that country | 3003 | TW |
| Port name in any order and country (ISO) | match against port name in any order against port name column of reference data along with country name where port should belong to that country | ABERDEEN-HOQUIAM, WASH<br>ADDISON USER FEE AIRPORT, DALLA | TW |
| Port name in any order | match against port name in any order against port name column of reference data | ABERDEEN-HOQUIAM, WASH<br>ADDISON USER FEE AIRPORT, DALLA | UK |

| Matching Rule | Summary of Rule Logic | Example Matching Data- Port Name | Example Matching Data- Country |
|---|---|---|---|
| Port name at least one word matching and country (ISO) | word match count is >0 along with country name where port should belong to that country | ABERDEEN | UK |
| Port name at least one word matching | word match count is >0 along with country name | ABERDEEN | UR |

# OFSAA Support Contact Details

Raise a Service Request (SR) in My Oracle Support (MOS) for queries related to OFSAA applications.

# Send Us Your Comments

Oracle welcomes your comments and suggestions on the quality and usefulness of this publication. 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 about this manual?

If you find any errors or have any other suggestions for improvement, indicate the title and part number of the documentation along with the chapter/section/page number (if available) and contact the Oracle Support.

Before sending us your comments, you might like to ensure that you have the latest version of the document wherein any of your concerns have already been addressed. You can access My Oracle Support site which has all the revised/recently released documents.