Oracle Financial Services Customer Screening

Oracle Financial Services Customer Screening Matching Guide

Version 8.0.7.0.0

Jan 2019

ORACLE®

Table of Contents

Chapter 1: Introduction

Oracle Financial Services Customer Screening 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. In these cases, the additional review work of lower probability matches will be necessary. By contrast, a business carrying out **PEP screening** may choose a strategy of finding and investigating only the most likely matches against the PEP list, and the additional work required to confirm or eliminate weaker matches may not be cost effective for the business.

Oracle Financial Services Customer Screening 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 Customer Screening to your requirements.

The matching rules are built around name matching. Other identifiers are also used in the matching rules, but their main purpose is to rank matches by strength, and thereby to enable a most-likely approach to reviewing potential matches. Oracle Financial Services Customer Screening also includes an evaluation of the risk posed by the potential match, allowing both strength of match and risk profile to be used in prioritizing reviews. For example, strong matches to Sanctions lists should be regarded as the most urgent matches, requiring immediate attention. Strong matches to PEP records will require follow-up, but may not be so urgent. Looser matches to PEP records may not be worth the time and operational cost of review.

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.

1.1   General matching strategy

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

1.1.1   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).

1.1.2   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.

1.1.3   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 match rule also defines a decision to be applied to any records which satisfy the conditions of the rule. The majority of rules have a **Review** decision, meaning matches that hit the rule need to be reviewed. However, there are also elimination rules, where if the records being compared meet the rule's criteria, a **No Match** decision is reached and the two records will not be considered a match.

> **Note**: Oracle Financial Services Customer Screening does not use the **Match** decision as it never considers there to be an automatic match between two records that does not require review.

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.

1.2   Configuring Oracle Financial Services Customer Screening for different scenarios

As previously mentioned, Oracle Financial Services Customer Screening includes clusters and matching rules that are suited to various screening requirements. Tuning Oracle Financial Services Customer Screening 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 Customer Screening:

- 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 should consider the following actions:

- Disable all the elimination rules in the individual matching processors. If these rules are enabled, you run the risk of preventing close name matches from being detected by the remaining match rules.
- Enable the conflict[i] rules in the individual matching processors. These rules allow you to detect and manually review close name matches whose supporting information conflicts with the watch list records.
- If name only clusters are enabled, the **Name and Country** and **Name and YOB** clusters are redundant and therefore can be disabled for PEP and EDD screening. They are disabled by default for Sanctions screening.

- Country prohibition screening is enabled by default for Sanctions screening. To screen for prohibited countries during PEP and EDD screening, enable the relevant country prohibition clusters and match rules:
  - The clusters used for country prohibitions in entity screening are the **Registration Country** cluster and the **Operating Country** cluster.
  - The match rules used for country prohibitions in entity screening are the **Country Prohibition – Registration Country** and the **Country Prohibition – Operating Country** match rules.
  - The clusters used for country prohibitions in individual screening are the **Nationality Prohibition** and the **Residency Prohibition** clusters.
  - The match rules used for country prohibitions in individual screening are the **Country Prohibition – Nationality** and the **Country Prohibition – Residency** match rules.

- 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.

---

[i] The conflict rules raise possible matches when the individual name appears to match a given watch list record, but the supporting data (such as date of birth or nationality) is in conflict.

Chapter 2: Individual matching

This section details the default configuration when matching individuals to Sanctions, PEP and EDD lists.

2.1   Identifier preparation

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

Identifier Description	Standard prepared attribute name	Summary of preparation logic
Given Names	dnGivenNames	A space-separated list of the first and middle names of the individual, after normalization (see the name normalization section, below).
Family Name	dnFamilyName	A normalized version of the family name (see the name normalization section, below).
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.
City	dnCity	A pipe-separated list of cities associated with the individual data.
Country Code	A space separated, de-duplicated and sorted superset of all country codes provided in dnAddressCountryCode, dnResidencyCountryCode, dnNationalityCountryCodes and dnCountryOfBirthCode.	A space separated list of standard 2-character country codes.
Date of Birth	dnDOB	A date attribute containing the date of birth of the individual.
Year of Birth	dnYOB	A string attribute containing a space-separated list of possible years of birth, in a four-digit format.

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

2.1.1   Name normalization

The name identifiers map to the prepared attributes `dnGivenNames`, `dnFamilyName` and `dnFullName`. In all these fields, the following transformations are applied before matching:

1. Standardization of accented characters.

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

> **Note**: If matching data in the original language against original script names in watch lists, the appropriate character ranges should be removed from the Name Noise Characters Reference Data so that they are not replaced. If transliterating data before matching, transliteration must 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.

Examples

Input data		Identifiers		
Forename	Surname	dnGivenNames	dnFamilyName	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

2.1.2   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.

2.1.3   Date of birth and Year of birth identifiers

A formal Date attribute holds the date of birth, where known. The year of birth is stored as a string, and is either derived from the date of birth, or may be derived from other data. The year of birth may include several possible years. This is most likely to occur when a reference source lists the age of individuals as of a given date, which may lead to two possible years of birth.

For example, if an individual is listed as 27 years old on 01/05/2007, the year of birth may be 1980 (if born before 1st May), or 1979 (if born after 1st May). In this case, both possible years are derived and added to a list of possible years of birth. The year of birth comparison in matching looks for a common year of birth between the two records being compared.

2.2   Clustering

Oracle Financial Services Customer Screening provides eleven clusters for matching individuals to watch lists during Sanctions screening, and nine clusters for PEP and EDD screening:

Cluster Method	SAN	PEP	EDD
Family Name	Y	N	N
Full Name Metaphone	Y	N	N
Given Names	Y	N	N
Full Name Trim	Y	N	N
Nationality Prohibition	Y	N/A	N/A
Residency Prohibition	Y	N/A	N/A
Name and Country	N	Y	Y
Name and YoB	N	Y	Y
First and Last Name	N	Y	Y
Original Script Name	N	N	N
First Initial Last Name	N	N	N

Note: This table shows the default configuration of both Batch and 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` and `dnOriginalScriptName`. For further information see 2.1.1 : Name normalization.

2.2.1 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:

dnFullName	Name tokens and trimmed values		Cluster Keys	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.

NOTE: Many other punctuation and noise characters are normalized to spaces before generating the cluster. For further information see 2.1.1 : 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

Examples

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

2.2.2 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.

NOTE: Many other punctuation and noise characters are normalized to spaces before generating the cluster. For further information see 2.1.1 : 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.

Examples

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		
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		
REGINALD B SR GOODRICH	B	P	KTRRJN KTRSR RJNSR **NOTE:** Initials are ignored by default when generating cluster keys	KTRRJN|KTRSR|RJNSR
	GOODRIDGE	KTRJ		
	REGINALD	RJNLT		
	SR	SR		

dnFullName	Name tokens and Metaphone values		Distinct Cluster Keys	dnClusterFullNameMeta
STEPHEN JEQE NKOMO	JEQE	JK	JKNKM JKSTF NKMSTF	JKNKM|JKSTF|NKMSTF
	NKOMO	NKM		
	STEPHEN	STFN		
S J NKOMO	J	J	NKM **NOTE:** 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		

2.2.3   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.

> **NOTE:** 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.

   > **NOTE:** Many other punctuation and noise characters are normalized to spaces before generating the cluster. For further information see <u>2.1.1 : Name normalization</u>.

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.

Examples

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

dnGivenNames	Metaphone values	dnClusterGivenNames
STEPHEN JEQE	STFN	STFN
S J	SJ	SJ
STEPHEN JEKE	STFN	STFN

2.2.4 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.

Consider the following two example records:

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.

Examples

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	ABA	ABAJOS ABATSA JOSTSA	ABAJOS|ABATSA|JOSTSA
	JOSEPH	JOS		

dnFullName	Name tokens and trimmed values		Cluster Keys	dnClusterFullNameTrim
ABANDA	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		
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 **NOTE:** Initials are ignored by default when generating cluster keys	NKO
	NKOMO	NKO		
	J	J		
STEPHEN JEKE N KOMO	JEKE	JEK	JEKKOM JEKSTE KOMSTE **NOTE:** Initials are ignored by default when generating cluster keys	JEKKOM|JEKSTE|KOMSTE
	KOMO	KOM		
	N	N		
	STEPHEN	STE		

2.2.5 Nationality Prohibition (Nationality Code)

This cluster uses the space-delimited list of nationality country codes to generate cluster keys by generating an array of the component country codes.

2.2.6 Residency Prohibition (Residency Code)

This cluster uses the space-delimited list of residency country codes to generate cluster keys by generating an array of the component country codes.

2.2.7 Name and Country (dnClusterNameCountry)

The **Name and Country** cluster provides a backup using more detailed information about names and combining them with country information. The cluster is used to compare very similar names that are located over the same countries.

The default logic of the cluster builder is as follows:

1. Split the normalized Full Name into name tokens, using space as a delimiter.

    NOTE: Many other punctuation and noise characters are normalized to spaces before generating the cluster. For further information see 2.1.1 : Name normalization.

    ---

2. Apply the **Metaphone** transformation to each name token, outputting a key with a length of up to twelve characters.

3. Sort the Metaphone values alphabetically.

4. For each country code associated with the record:

    a. Concatenate the country code with the full set of Metaphone values, using an underscore as a separator.

    b. If more than two Metaphone values are present, then iterate through all groups of Metaphone values which have exactly one value from the set missing, concatenating the country code onto the front of the Metaphone value set.

    c. If the overall length of the **dnClusterNameCountry** field has exceeded 1000 characters, discard the last key and stop key generation.

Examples

dnFullName	Country Codes	Name tokens and Metaphone values		Cluster Keys	dnClusterNameCountry
MOHAMMED SANI	ES GB	MOHAMMED	MHMT	ES_MHMT_SN GB_MHMT_SN	ES_MHMT_SN|GB_MHMT_SN
		SANI	SN		
SULIMAN HAMD SULEIMAN	ES TH GB	HAMD	HMT	ES_HMT_SLMN_SLMN	ES_HMT_SLMN_SLMN|
		SULEIMAN	SLMN	ES_SLMN_SLMN	ES_SLMN_SLMN|ES_HMT_SLMN|
		SULIMAN	SLMN	ES_HMT_SLMN	ES_HMT_SLMN|TH_HMT_SLMN_SLMN|TH_SLMN_SLMN|TH_HMT_SLMN|GB_HMT_SLMN_SLMN|GB_SLMN_SLMN|GB_HMT_SLMN
				ES_HMT_SLMN	
				TH_HMT_SLMN_SLMN	
				TH_SLMN_SLMN	
				TH_HMT_SLMN	
				TH_HMT_SLMN	
				GB_HMT_SLMN_SLMN	
				GB_SLMN_SLMN	
				GB_HMT_SLMN	
				GB_HMT_SLMN	

2.2.8   Name and YOB (dnClusterNameYOB)

The **Name and YOB** cluster provides a backup using more detailed information about names and initials combining them with years of birth.

The default logic of the cluster builder is as follows:

1. Standardize dnGivenNames and dnFamilyName;

2. Apply transliteration followed by the **Metaphone** transformation to the standardized given name, outputting a key with a length of up to four characters;

3. Apply transliteration followed by the **Metaphone** transformation to the standardized family name, outputting a key with a length of up to four characters;

4. Extract and uppercase the first letter of the standardized dnGivenName;

5. Extract and uppercase the first letter of the standardized dnFamilyName;

6. Extract the first two years of birth from dnYOB to generate two values (referred to as 'First YOB' and 'Second YOB' in the remainder of this example);

7. Create up to four cluster keys by concatenating the following combinations of elements, using the underscore character:

   a. First YOB + dnFamilyName (uppercased initial) + dnGivenNames (Metaphone).

   b. First YOB + dnGivenNames (uppercased initial) + dnFamilyNames (Metaphone).

   c. Second YOB + dnFamilyName (uppercased initial) + dnGivenNames (Metaphone).

   d. Second YOB + dnGivenNames (uppercased initial) + dnFamilyNames (Metaphone).

   NOTE: If any of the required data elements are missing, then the corresponding cluster key will not be generated.

8. Deduplicate the list of keys

Examples

dnGivenNames, dnFamilyName	dnYOB	Name tokens and Metaphone values		Cluster Keys	dnClusterNameYOB
MOHAMMED, SANI	1969 1970 1971	MOHAMMED	MHMT	1969_S_MHMT 1969_M_SN 1970_S_MHMT 1970_M_SN	1969_S_MHMT| 1969_M_SN| 1970_S_MHMT| 1970_M_SN
		SANI	SN		
SULIMAN HAMD, SULEIMAN	1980 1981 1982	HAMD	HMT	1980_S_SLMN 1981_S_SLMN	1980_S_SLMN| 1981_S_SLMN
		SULEIMAN	SLMN		
		SULIMAN	SLMN		

2.2.9   First and Last Name (dnClusterFirstLast)

The **First and Last Name** cluster provides a tighter name only clustering method that relies on the first given name and last family name matching after standardization and allows for variation in any of the name tokens in-between.

The default logic of the cluster builder is as follows:

1. Strip initials from the normalized given names and family name.

2. Strip all common name qualifiers from the normalized given names and family name, e.g. Al, Bin, Von.

3. Extract the first token from the stripped given names. If all tokens were stripped in steps 1 and 2, then extract the first token from the original normalized given names.

4. Extract the last token from the stripped family name. If all tokens were stripped in steps 1 and 2, then extract the last token from the original normalized family name.

5. Trim the extracted values to a maximum length of 4 characters.

6. Sort the trimmed values alphabetically and concatenate to generate the final key value.

Examples

dnGivenNames	dnFamilyName	Extracted Values		dnClusterFirstLast
OSVALDO ANTONIO	CASTELL VALDEZ	OSVALDO	VALDEZ	OSVAVALD
ABU MAHDI	AL MUHANDIS	MAHDI	MUHANDIS	MAHDMUHA
ABU	NIDAL	ABU	NIDAL	ABUNIDA
V U	SHEIMAN	V	SHEIMAN	SHEIV

2.2.10 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.

> **Note**: 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.

Examples

dnOriginalScriptName	dnClusterOriginalScript
Іван Антонавіч Шчурок	Іван|Антон|Шчуро
林 紹 蔵	林|紹|蔵
ဆင် ဆင် ဆော	Myanmar
محمد محمد ملكسوري	ملكسور|محمد

2.2.11 First Initial Last Name (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:

    a. 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.

    b. 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.

Examples

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

2.3   Matching

Oracle Financial Services Customer Screening uses different approaches to matching for different use cases. For Sanctions screening, a zero tolerance approach to matching is assumed, where secondary data such as dates and years of birth, and nationalities cannot necessarily be assumed to be correct. In this case, it may be important to present matches where there is a level of name match even if other data would indicate that a match is unlikely. When screening against lists of Politically Exposed Persons (PEPs) or other individuals on watchlists (Enhanced Due Diligence matching), where the occasional 'false negative' may be tolerable from a business perspective, match rules are generally 'tighter' and demand at least one item of secondary data (such as a nationality, year of birth or date of birth) matches as well as a name of match. However, the screening rules for each screening process can, and should, be tailored according to the business appetite to risk. Oracle Financial Services Customer Screening also provides separate processes for Batch and Real-Time screening, as these may be subject to different matching strategies.

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 Customer Screening 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 Customer Screening 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.

NOTE: 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 customer data; and

- the provenance of the customer 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 Customer Screening.

2.3.1 Match Rules

There are several different types of match rule involved in Individual Screening:

- The elimination rules. These are used in various positions in the rule templates to eliminate any records that have conflicting supporting data. The elimination rules may be moved up and down in order to change when they are applied during the matching process.

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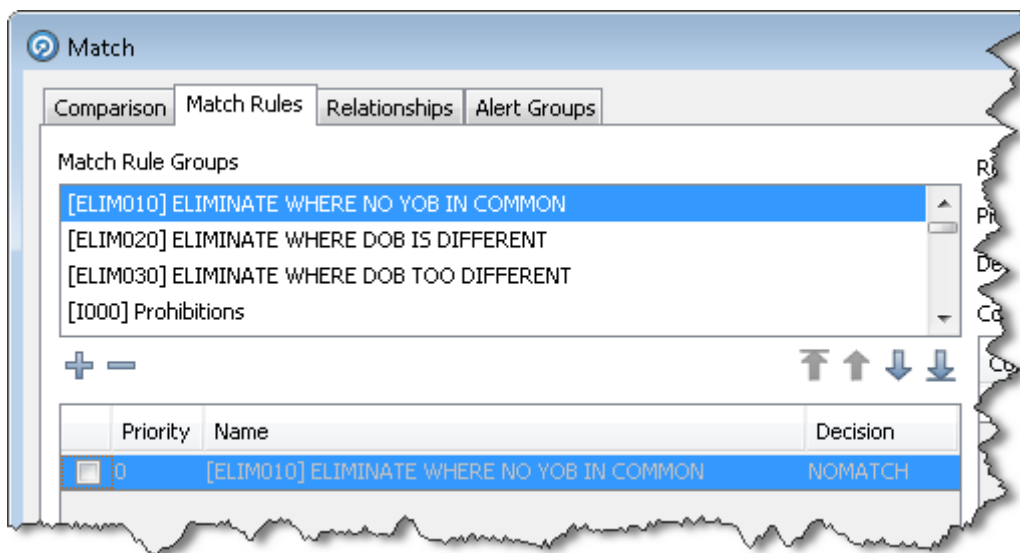
> **NOTE:** This means that the match rules are not ordered by strength across all identifiers. For example, a weaker name match that is strengthened by matches on date of birth, city and country is likely to be a stronger overall match than a strong name with strongly contradictory data in the other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

Oracle Financial Services Customer Screening includes many match rules for each level of name match, reflecting the match strength of any additional information, particularly date of birth and location data. 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 shown below:

As each group is selected, the match rules it contains are displayed in the window below.

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 Customer Screening in greater detail.

2.3.2 Prohibition Rules

The Prohibition rules check for country information in an individual's record against the list of prohibited countries and nationalities maintained in List Management.

Group Code	Matching Rule	Summary of Rule Logic
I000A	Country prohibition - Residency	The country of residence given matches a prohibited country.
I000B	Country prohibition - Nationality	The nationality given matches a prohibited nationality.

2.3.3 Elimination Rules

Elimination Rule	Summary of Rule Logic	Enabled by default?
ELIMINATE WHERE NO YOB IN COMMON	This rule will eliminate pairs of records if both YOB fields are populated and there is no value in common.	Yes
ELIMINATE WHERE DOB IS DIFFERENT	This rule will eliminate pairs of records if both DOB fields are populated and there is no value in common.	No
ELIMINATE WHERE DOB TOO DIFFERENT	This rule will eliminate pairs of records if the date of birth differs too greatly between the two records. Pairs are eliminated if there are 6 or ore years difference between DoBs, **and** one typographical error, **and** one typographical error in a month.	No
ELIMINATE WHERE GENDER IS DIFFERENT AND BOTH DERIVED OR BOTH STATED	This rule will eliminate pairs of records if the genders are different, and EITHER both records had the gender specified as part of the input record, OR both records have a gender value which was derived from other fields.	Yes
ELIMINATE WHERE NO COUNTRY SHARED AND ALL SAFE	This rule will eliminate pairs of records if there are no countries in common in the Country fields, AND if all countries listed are on the Safe list. The Safe list is maintained in the **Match - Individual Safe Countries ISO Codes** Reference Data.	Yes
ELIMINATE WHERE NO NATIONALITIES IN COMMON	This rule will eliminate pairs of records if the Nationality fields contain no common entries.	Yes
ELIMINATE WHERE LIST OCCUPATION IS SAFE	This rule will eliminate pairs of records if the List Occupation field contains only values in the **Match - Safe Occupations** Reference Data.	Yes
ELIMINATE WHERE CUSTOMER RISK SCORE BELOW THRESHOLD	This rule will eliminate pairs of records if the Customer Risk Score is below a threshold specified in the corresponding screening process.	No
ELIMINATE WHERE LIST RISK SCORE BELOW THRESHOLD	This rule will eliminate pairs of records if the List Risk Score is below a threshold specified in the corresponding screening process.	No
ELIMINATE WHERE LIST PEP RISK SCORE BELOW THRESHOLD	This rule will eliminate pairs of records if the List PEP Risk Score is below a threshold specified in the corresponding screening process.	No

Note: No elimination rules are enabled by default for Sanction records.

2.3.4 Name Matching Rules

Group Code	Matching Rule	Logic Summary	Example Matching Data	
			Given Names	**Family Name**
I010	Exact name	Given names and family name match exactly.		

Group Code	Matching Rule	Logic Summary	Example Matching Data	
			JOSEPH	TSANGA
			JOSEPH	T'SANGA
I020	**Original script name exact**	The original script Name fields match exactly.	**Original Script Name**	**Original Script Name**
			АЛЕКСАНДР ОСОКИН	АЛЕКСАНДР ОСОКИН
I030	**Standardized given name**	Given names match after name standardization using Given name map. Family name matches exactly.	**Given Names**	**Family Name**
			BILL	JONES
			WILLIAM	JONES
I040	**Full name**	The full name matches exactly, after standardization of all name tokens using the Given Name Map.	**Full Names**	
			JOHN MIKE SMITH	
			JOHN MICHAEL SMITH	
I050	**Full name without titles**	The full name matches exactly, after standardization of all name tokens using the Given Name Map and removal of titles.	**Full Names**	
			DR DOUGLAS BAKER	
			DOUGLAS BAKER	
I060	**Abbreviated standardized given name**	Given names match using a **Starts With** comparison, after name standardization using the Given Name Map. Family name matches exactly.	**Given Names**	**Family Name**
			JOSEPH ABANDA	TSANGA
			JOSEPH	T'SANGA
I070	**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	**Given Names**	**Family Name**
			JOSEPH	ABANDA
			JOESPH	ABANDA
I080	**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.	**Given Names**	**Family Name**
			AMER MOHAMMAD RASHEED	AL UBAIDI
			AMIR RASHID MOHAMMED	AL UBAIDI
I090	**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 .	**Given Names**	**Family Name**
			MOHAMMED	HANIF
			DIN MOHAMED	HANIF
I100	**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. 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.	**Full Name**	
			LOTFI RIHANI	
			LOTFI BEN ABDUL HAMID BEN ALI RIHANI	
I110	**Original script name in any order**	All names in the original script name fields match, regardless of order.	**Original Script Name**	**Original Script Name**
			ΚΑΡΛΟΣ ΜΟΛΙΝΑ	ΜΟΛΙΝΑ ΚΑΡΛΟΣ
I120	**Original script name with typos**	Original script name fields match with an 80%+ **Character Match Percentage** score.	**Original Script Name**	**Original Script Name**
			Καρλος	Καρλος

Group Code	Matching Rule	Logic Summary	Example Matching Data	
			Μολίνα	Μολίννα
I130	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.	**Full Name**	
			ABDUL JABBER OMARI	
			OMARI ABDUL JABBER	
I140	Abbreviated given name	Given names match using a **Starts With** comparison. Family name is a close metaphone match.	**Given Names**	**Family Name**
			CHRIS	HUNT
			CHRISTOPHER	HUNTER
I150	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.	**Given Names**	**Family Name**
			IBRAHIM ABDUL SALAM	MOHAMED BOYASSEER
			IBRAHIM	BOYASEER
I160	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.	**Given Names**	**Family Name**
			SAHIR	BARHAN
			DR SAHIR MUSA	BERHIN
I170	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.	**Original Script Name**	**Original Script Name**
			ХАСАН	ЧЕНГИЋ
			ЧЕНГИЋ	ХАССАН
I180	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.	**Given Names**	**Family Name**
			MOHAMMAD HUSAYN	MASTASAEED
			MOHAMMAD HASSAN	MASTASAEED
I190	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.	**Given Names**	**Family Name**
			AMER MOHAMMAD RASHEED	AL UBAIDI
			AMIR RASHID MOHAMMED	AL UBEIDI
I200	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.	**Given names**	**Family name**
			VIKTOR ANATOLYEVICH	BOUT
			VICTOR	BOOT
I210	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.	**Original Script Name**	**Original Script Name**
			МИЛЕНКО ВРАЧАР	МИЛЕНКО ИВАНОВИЧ ВРАЧАР
I220	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	**Full Name**	
			ABDUL WAHED SHAFIQ	
			ABDUL WAHAD	

Group Code	Matching Rule	Logic Summary	Example Matching Data	
		example, 'ABDUL' matches 'ABDUL ABDUL' with a Word Match Count of 2. Matching is order sensitive.		
I230	**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.	**Full Name** ABU BAKAR ABU BAKAR BA'ASYI	
I240	**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.	**Full Name** MOHAMMED AL GHABRA ALGHABRA MUHAMAD RAMATULLAH WAHIDYAR FAQIR MOHAMMAD WAHIDYAR RAMA TULLAH	
I250	**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.	**Original Script Name** ЮРИ НЕЁЛОВ	**Original Script Name** ЮРИЙ ВАСИЛЬЕВИЧ НЕЁЛОВ
I260	**Abbreviated first name**	The first given name matches with a **Starts With** match, after name token standardization. Family name matches exactly.	**Given Names** KHADAF ABUBAKAR KHADAFFI	**Family Name** JANJALANI JANJALANI
I270	**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.	**Full Name** HA THI NGUYEN THI HA	
I280	**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.	**Full Name** STEPHENS MARTIN MARRTIN JOHN STEPHENS	

2.3.5 Loose Name Matching Rules

Group Code	Matching Rule	Summary of Rule Logic	Example Matching Data
I290	**Full name characters shorter**	The full name matches with a **Longest Common Substring Sum Percentage** of 90%, relating to the shorter string, and considering	**Full Name** ABU BAKAR

Group Code	Matching Rule	Summary of Rule Logic	Example Matching Data
		substrings of 5 characters or more in length, after name standardization. At least 2 name tokens must exist in the full name.	ABU BAKAR BA'ASYI
I300	**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 the standardized Full Name field without initial must be fully contained within the other standardized Full Name field without initials; AND both fields must contain at least two names.	**Full Name** CARL J FISHER J C FISHER
I310	**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; AND the initial of the last name in the record being screened must match the initial of the name in the watch list record.	**Full Name** JANINE CHERRY CHERRY

2.3.6 Deprecated Name Matching Rules

All these rules are assigned the Rule Group Code I990. These are legacy rules that are superseded by the above rule set, and are included here to assist existing Customer Screening customers with the transition to the current version.

Name matching rule	Summary of rule logic	Example matching data	
Given name in common	At least one given name is found in common, after name standardization. Family name matches exactly.	**Given Names** HASSAN ALI IBRAHIM HASSAN	**Family Name** AL TIKRITI AL TIKRITI
Full name similar and family name sounds like	Full name matches with a **Character Match Percentage** of 80% or more after name standardization. At least one of the family name tokens (excluding initials) must match by a 4-character Metaphone key.	**Full Name** AKHYAR MOHAMMED MANSOUR AKHTAR MUHAMED MANZUR	
Similar first name	The first given name matches with an edit distance of between 1 and 2 after name standardization, and with a **Character Match Percentage** of 66% or more. Family name matches exactly.	**Given Names** MIKOLAI NIKOLAI TIMOFEEVICH	**Family Name** METELITSA METELITSA
Similar first name and family name	The first given name matches with an edit distance of between 1 and 2 after name standardization, and with a **Character Match Percentage** of 66% or more. Family name matches with a **Character Match Percentage** of 66% or more. At least one of the family name tokens (excluding initials) must match by a 4-character Metaphone key.	**Given Names** GENNADY GENNADIY	**Family Name** NEVYGLAS NYAVIGLAS
Given names in common and similar family name and sounds like	At least one given name is found in common, after name standardization. The family name matches with a **Character Edit Distance** of 1-2. The family name matches by 4-character Metaphone key.	**Given Names** ABDUL JABBAR FAROUK ABDUL	**Family Name** OMAIRI OMARI
Abbreviated standardized given name and family name contained	Given names match using a **Starts With** comparison, after name standardization using the Given Name Map. Family name matches using **Contains** comparison after token standardization.	**Given Names** A ABDUL	**Family Name** RAHIMI RAHIM

Name matching rule	Summary of rule logic	Example matching data	
Similar given name	The given name matches with a **Character Edit Distance** of between 1 and 2 after name standardization. Family name matches exactly.	**Given Names**	**Family Name**
		NAY	WIN
		NYAW	WIN
Full name contained	The full name matches with a **Contains** match, after standardization of all name tokens using the Given Name Map.	**Full Name**	
		CHARNI KOKO	
		KO KO	
Full name similar	The full name matches with a **Character Match Percentage** of 80% or above, after name token standardization.	**Full Name**	
		JUAN LOIS RUBENACHROIG	
		JUAN LOIS RUBENACH ROIZ	
Abbreviated first name and similar family name	The first given name matches with a **Starts With** match, after name token standardization. The family name matches with a **Character Edit Distance** of 1-2.	**Given Names**	**Family Name**
		A	UMARI
		ABU	OMAR
Given name in common and similar family name	At least one given name is found in common, after name standardization. The family name matches with a **Character Edit Distance** of 1-2. The family name matches by 4-character Metaphone key.	**Given Names**	**Family Name**
		NURJAMAN RIDUAN	ISAMU DIN
		RIDUAN	ISOMUDDIN
First name and family similar	The first given name matches with a **Character Edit Distance** of between 1 and 2 after name standardization, and with a **Character Match Percentage** of 66% or more. Family name matches with a **Character Match Percentage** of 66% or more.	**Given Names**	**Family Name**
		REGINADL	GOODRIDGE
		REGINALD	GOODRICH

2.3.7   Ranking matches within Name rules

Match Rule	Summary of Matching Logic	Example Matching Data		
[Name rule], city, DoB	At least one city matches. Date of birth matches exactly.	**DoB**	**City**	
		01/11/1963	London	
		01/11/1963	New York| London	
[Name rule], country, DoB	At least one country matches. Date of birth matches exactly.	**DoB**	**Country**	
		25/01/1959	PK IN US	
		25/01/1959	PK	
[Name rule], DoB	Date of birth matches exactly.	**DoB**		
		19/09/1968		
		19/09/1968		
[Name rule], city, YoB, no DoB	At least one city matches. Year of birth matches. No date of birth provided.	**YoB**	**City**	**DoB**
		1978	Lahore| Mumbai	-
		1978	Lahore	-
[Name rule], country, YoB, no DoB	At least one country matches. Year of birth matches. No date of birth provided.	**YoB**	**Country**	**DoB**
		1962	IQ US	-
		1962	IQ	-
[Name rule], YoB, no DoB	Year of birth matches. No date of birth provided.	**YoB**		**DoB**
		1975		-
		1975		-
[Name rule], city, DoB similar	At least one city matches. Dates of birth are a close match, according to **one** of the following parameters only: DD and MM values are transposed, but YYYY matches exactly. DD and MM match, YYYY does not. DD and YYYY match, MM does not. DD values differ by 5 or less.	**DoB**	**City**	
		08/04/1967	Riyadh	
		04/08/1967	Riyadh	
[Name rule], country,	At least one country matches. Dates of birth are a	**DoB**	**Country**	

Match Rule	Summary of Matching Logic	Example Matching Data		
DoB similar	close match, according to **one** of the following parameters only: DD and MM values are transposed, but YYYY matches exactly. DD and MM match, YYYY does not. DD and YYYY match, MM does not. DD values differ by 5 or less.	08/04/1967	SA	
		08/04/1977	SA	
[Name rule], DoB similar	Dates of birth are a close match, according to **one** of the following parameters only: DD and MM values are transposed, but YYYY matches exactly. DD and MM match, YYYY does not. DD and YYYY match, MM does not. DD values differ by 5 or less.	**DoB**		
		19/06/1967		
		16/06/1967		
[Name rule], city, YoB (DoB conflict)	At least one city matches. Year of birth matches. Dates of birth do not match.	**YoB**	**City**	**DoB**
		1978	Lahore| Mumbai	13/04/1978
		1978	Lahore	04/08/1978
[Name rule], country, YoB (DoB conflict)	At least one country matches. Year of birth matches. Dates of birth do not match.	**YoB**	**Country**	**DoB**
		1962	IQ	05/07/1962
		1962	IQ	04/11/1962
[Name rule] YoB (DoB conflict)	Year of birth matches. Dates of birth do not match.	**YoB**		**DoB**
		1962		05/07/1962
		1962		04/11/1962
[Name rule], city	At least one city matches.	**City**		
		Lahore| Mumbai		
		Lahore		
[Name rule], country	At least one country matches.	**Country**		
		IQ PK		
		IQ		
[Name rule] only	Name rule returns a match. No data in other fields.	**Name**	**Country**	**DoB**
		J SMITH	-	-
		J SMITH	-	-
[Name rule] (conflict)	Name rule returns a match. Data in other fields does not match.	**Name**	**Country**	**DoB**
		J SMITH	IQ	05/07/1962
		J SMITH	UK	04/11/1974

Chapter 3: Entity Matching

This section details the default configuration when matching entities to Sanctions, PEP and EDD lists. In general, and by default, the matching strategy for entities in Oracle Financial Services Customer Screening will raise 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 the following techniques (amongst others):

- 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

3.1   Identifier Preparation

The following identifiers are prepared for use in the entity matching process:

Identifier	Summary of preparation logic
Original Entity Name	The original entity name, after Name Normalization. See section 3.1.1 "Name Normalization" below.
Standardized Entity Name	A standardized version of the entity name, with common entity name suffixes standardized. The standardization process may be amended by changing the Reference Data used to standardize tokens (such as LTD) and phrases (such as FIN SERVS).
Original Script Name	A whitespace normalized version of the original script name.
City	A pipe-separated list of cities.
Country Codes	A space separated list of standard 2-character country codes.

3.1.1   Name Normalization

Entity names are normalized using the following logic:

1. Standardization of accented characters.
2. Removal of apostrophes.
3. Replacement of all other characters apart from alpha (A-Z or a-z), numeric (0-9) or ampersand (&) characters with spaces.

   > **NOTE:** If matching data in the original language against original script names in watch lists, the appropriate character ranges should be removed from the Name Noise Characters Reference Data so that they are not replaced. In addition, if transliterating data before matching, transliteration must be done before name normalization.

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

3.2 Clustering

Oracle Financial Services Customer Screening provides three different clustering strategies for matching entities: **Entity Name Tokens**, **Name Metaphone**, and **Name Trimmed**. Any of the clusters may be activated or deactivated, as required, and different cluster limits can be configured.

3.2.1 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.

Examples

dnEntityName	Name with initials and common name tokens stripped	dnClusterNameTokens
ANGLO CARIBBEAN CO LTD	ANGLO CARIBBEAN	ANGLO|CARIBBEAN
GUAMATUR S A	GUAMATUR	GUAMATUR

3.2.2 Name Metaphone (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.

Examples

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

3.2.3   Name Trimmed (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.

Examples

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

3.2.4   Registration Country Prohibition (Registration Country Code)

This cluster uses the space-delimited list of registration country codes to generate cluster keys by generating an array of the component country codes.

3.2.5   Operating Country Prohibition (Operating Country Code)

This cluster uses the space-delimited list of operating country codes to generate cluster keys by generating an array of the component country codes.

3.2.6   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:
   a. The first five characters of the token.
   b. The last five characters of the token.

Examples

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

3.2.7 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.

Note: 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 name in Burmese 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 delimiter
4. Deduplicate the list of keys.

Examples

dnOriginalScriptName	dnClusterOriginalScript
ЧЕРЕН СЕПТЕМВРИ	ЧЕРЕН|СЕПТЕ
北 京 航 空 航 天 大 学	北|京|航|空|航|天|大|学
အိပုန်	Myanmar

3.3 Matching

Entity matching is centered around entity names. Other items of data, such as associated countries and cities, are used to strengthen a possible match.

3.3.1 Match Rules

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

Optional elimination rules exist that allow lower risk matches to be suppressed.

There are three types of match rule involved in entity screening:

- The elimination rules, which are used in the rule templates to suppress the generation of lower risk matches - for example, low quality matches against list records with a low risk score. The elimination rules may be moved up and down in order to change where they apply.

- The 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.

NOTE: 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.

- The 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 sanctions lists, where it is important that no true matches are missed.

For the sake of clarity, match rules are divided into groups, as shown below:

As each group is selected, the match rules it contains are displayed in the window below.

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 entity matching rules that are present in Oracle Financial Services Customer Screening in greater detail.

3.3.2   Prohibition Rules

The Prohibition rules check for country information in an entity's record against the list of prohibited countries and nationalities maintained in List Management.

Group Code	Matching Rule	Summary of Rule Logic
E000A	Country prohibition - Operating country	The country or countries of operation given match at least one prohibited country.
E000B	Country prohibition - Registration country	The country or countries of registration given match at least one prohibited country.

3.3.3 Elimination Rules

Elimination Rule	Summary of Rule Logic	Enabled by default?
ELIMINATE WHERE NO COUNTRY SHARED AND ALL SAFE	This rule will eliminate pairs of records if there are no countries in common in the Country fields, AND if all countries listed are on the Safe list. The Safe list is maintained in the **Match - Entity Safe Countries ISO Codes** Reference Data.	Yes
ELIMINATE WHERE CUSTOMER RISK SCORE BELOW THRESHOLD	This rule will eliminate pairs of records if the Customer Risk Score is below a threshold specified in the corresponding screening process.	No
ELIMINATE WHERE LIST RISK SCORE BELOW THRESHOLD	This rule will eliminate pairs of records if the List Risk Score is below a threshold specified in the corresponding screening process.	No
ELIMINATE WHERE LIST PEP RISK SCORE BELOW THRESHOLD	This rule will eliminate pairs of records if the List PEP Risk Score is below a threshold specified in the corresponding screening process	Yes

Note: No elimination rules are enabled by default in Sanctions screening.

3.3.4 Entity Matching Rules

Note that 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.

NOTE: that wherever 'word' is used below, this means a space-delimited token in the prepared names.

Group Code	Name Matching Rule	Summary of Rule Logic	Example Matching Data
V010	**Vessel part-standardized name exact**	The part-standardized entity name matches the name of a listed vessel exactly.	DYNASTY DYNASTY
V020	**Vessel name exact**	The entity name matches the name of a listed vessel after number cardinal and ordinal standardization.	4TH OCEAN FOURTH OCEAN
V030	**Vessel 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 RAHIM 3
V040	**Vessel name with typos**	The entity names match with a Character Match Percentage of 80-99% after number cardinal and ordinal standardization.	RAHUM 3 TRAHIM THREE
E010	**Part-standardized name exact**	The part-standardized entity name matches a listed entity name exactly.	HUMAN APPEAL INTERNATIONAL HUMAN APPEAL INTERNATIONAL
E020	**Name exact**	The entity names match exactly after number cardinal and ordinal standardization.	NOVEMBER 17 NOVEMBER SEVENTEEN
E030	**Original script name exact**	The original script names match exactly.	НИАЭП ОАО НИАЭП ОАО
E040	**Name without**	The entity names match exactly after number	CAPITAL DIRECT LTD

Group Code	Name Matching Rule	Summary of Rule Logic	Example Matching Data
	suffixes exact	cardinal and ordinal standardization, and after common company prefixes and suffixes are removed.	CAPITAL DIRECT AG
E050	Name without business words similar and sounds 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 UK PARAGON INVESTMENT CORPORATION
E060	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 HOLDINGS LTD LIFE HEALTH CARE INC
E070	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.	EDUCATION FOR HEALTH HEALTH EDUCATION SERVICES
E080	Name without 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 HOUSE) LIMITED BAE SYSTEMS PLC
E090	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 BERBERA
E100	Original script name in any order	All words in the Original Script Names match exactly, in any order.	ОАО НИАЭП НИАЭП ОАО
E110	Original script name with typos	The Original Script Names match with a Character Match Percentage of 80% or more.	ΕΠΑΝΑΣΤΑΤΙΚΉ ΑΡΙΣΤΕΡΆ ΕΠΑΝΑΣΤΑΙΚΉ ΑΡΙΣΤΕΡΆ
E120	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 GOLDSTEIN PROPERTIES INC
E130	Name without suffixes contains, 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	HAMPSHIRE HERITAGE DEVELOPMENTS LTD HERITAGE DEVELOPMENT CORPORATION

Group Code	Name Matching Rule	Summary of Rule Logic	Example Matching Data
		business words) in common between the two names.	
E140	**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 MOSCOW CENTER
E150	**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 HI TECH GROUP
E160	**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.	ΜΑΎΡΟΣ ΣΕΠΤΈμΒΡΗΣ ΣΕΠΤΈμΒΡΗΣ ΜΑΎΡΟΣΣ
E170	**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 BACK TO HEALTH CHIROPRACTIC
E180	**Name without business words, similar, sounds like, with multiple names and a residual token in common** NOTE: The group name differs from the rule name. See the Match dialog for details.	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 ASH UK LTD CHARLES F ASH CONSTRUCTION CO INC
E190	**Name without business words, similar with typos, sounds like, with multiple names and residual token in common** NOTE: The group namediffers 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	CLARKS HOME BAKERY LTD CLARK HOMES INC

Group Code	Name Matching Rule	Summary of Rule Logic	Example Matching Data
		primary entity name.	
E200	Name without business words, similar, sounds like, and residual token 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. 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. 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.	AMERICAN MILITARY SUPPLY AMERICAN SUPPLY CO
E210	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 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.	GENERAL ATOMICS GENERAL BUREAU OF ATOMIC ENERGY GBAE
E220	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 ACM
E230	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 ENRON CORP
E240	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 ACCURATE
E250	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	NON EMERGENCY TRANSPORT INC ACTION NON EMERGENCY TRANSPORTATION

Group Code	Name Matching Rule	Summary of Rule Logic	Example Matching Data
		business word, a word in the English dictionary or a very common word in Watchlist name data).	
E260	**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 PORT OF PUNTA GORDA
E270	**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 ВЪОРЪЖЕНА ГРУПА
E280	**Name without suffixes contains, multiple names 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 two significant words (not common business words) that match with a Character Match Percentage of 80 or more.	CITY TRANS LTD CAPITAL CITY TRANS SERV INC
E290	**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 IBERAIR LINES
E300	**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. 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 CLINIC LTD MED AMERICA CLINICS INC
E310	**Name has additional words, sounds like 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 DJ AND ASSOCIATES INC
E320	**Name has additional words with typos,**	All words in the shorter entity name match with a Character Match Percentage of 80 or more in	GARLOCK GARLICK

Group Code	Name Matching Rule	Summary of Rule Logic	Example Matching Data
	sounds like and residual token in common	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.	HELICOPTERS INC
E330	**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 NATIONWIDE SERVICES
E340	**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 CENTRAL MEDICAL INC
E350	**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 CHAIR LTD BLACK WORLD COLLEGE OF HAIR DESIGN
E360	**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 BARNO CHIROPRACTIC
E370	**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.	NEW ORLEANS MEDICAB OF METRO NEW ORLEANS
E380	**Name without**	The entity names are a Contains match and the	GROSVENOR

Group Code	Name Matching Rule	Summary of Rule Logic	Example Matching Data
	suffixes contains, similar, and multiple words in common	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.	NURSING SERVICES NURSING SERVICES INC
E390	**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.	АРАБСКИ РЕВОЛЮЦИОННИ БРИГАДИ АРАБСКИ РЕВОЛЮЦИОНИ
E400	**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 ATRIUM
E410	**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 BRILLIANCE
E420	**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 PRC

3.3.5 Loose Entity Matching Rules

Group Code	Name Matching Rule	Summary of Rule Logic	Example Matching Data
E430	**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 HENDERSON MANAGEMENT GROUP INC
E440	**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 HAMILTON INVESTMENT CORP
E450	**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 JACOBSON MANAGEMENT CO
E460	**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 MANAGEMENT LTD IDEAL ENTERPRISES INC
E470	**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 AVANTI ENTERPRISES INC
E480	**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 MOREXPRESS SA DE CV
E490	**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 INTER
E500	**Name without suffixes contains, significant overlap and**	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	EG ANDG TECHNICAL SERVICES INC TECHNICAL

Group Code	Name Matching Rule	Summary of Rule Logic	Example Matching Data
	multiple words in common	after number cardinal and ordinal standardization, and after common company prefixes and suffixes are removed.	SERVICES
E510	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.	FIRSA INTERNATIONAL LTD FIRST INTERNATIONAL COMMERCE BANK LTD

3.3.6 Ranking matches within Entity Name rules

For each entity or vessel name matching rule, matches are ranked according to how much and how strongly additional data matches between the customer record and the watch list:

Match Rule	Summary of Matching Logic	Example Matching Data		
[Entity name rule], city, country	At least one city matches. At least one country matches.	**City**		**Country**
		New York|London		GB
		London		GB US
[Entity name rule], city	At least one city matches.	**City**		
		Paris | London		
		Paris		
[Entity name rule], country	At least one country matches.	**Country**		
		US		
		PK IN US		
[Entity name rule] only	Entity name rule returns a match. No data in other fields.	**Name**	**Country**	**City**
		ACM	-	-
		ACM	-	-
[Entity name rule] (conflict)	Entity name rule returns a match. Data in other fields does not match.	**Name**	**Country**	**City**
		ACM	UK	London
		ACM	FR	Paris
[Vessel name rule] country	At least one country matches.	**Country**		
		US		
		PK IN US		
[Vessel name rule] only	Vessel name rule returns a match. No data in other fields.	**Name**	**Country**	**City**
		Dynasty	-	-
		Dynasty	-	-
[Vessel name rule] (conflict)	Vessel name rule returns a match. Data in other fields does not match.	**Name**	**Country**	**City**
		Dynasty	UK	London
		Dynasty	FR	Paris