CLEARMDM

QuickStart Guide 2 - Normalisation

Document Version: v1.4
Product Version: v2.26
Date: 14th April 2018

This document provides an overview and Step-by-Step implementation instructions for the clearMDM Normalisation MDM operation.

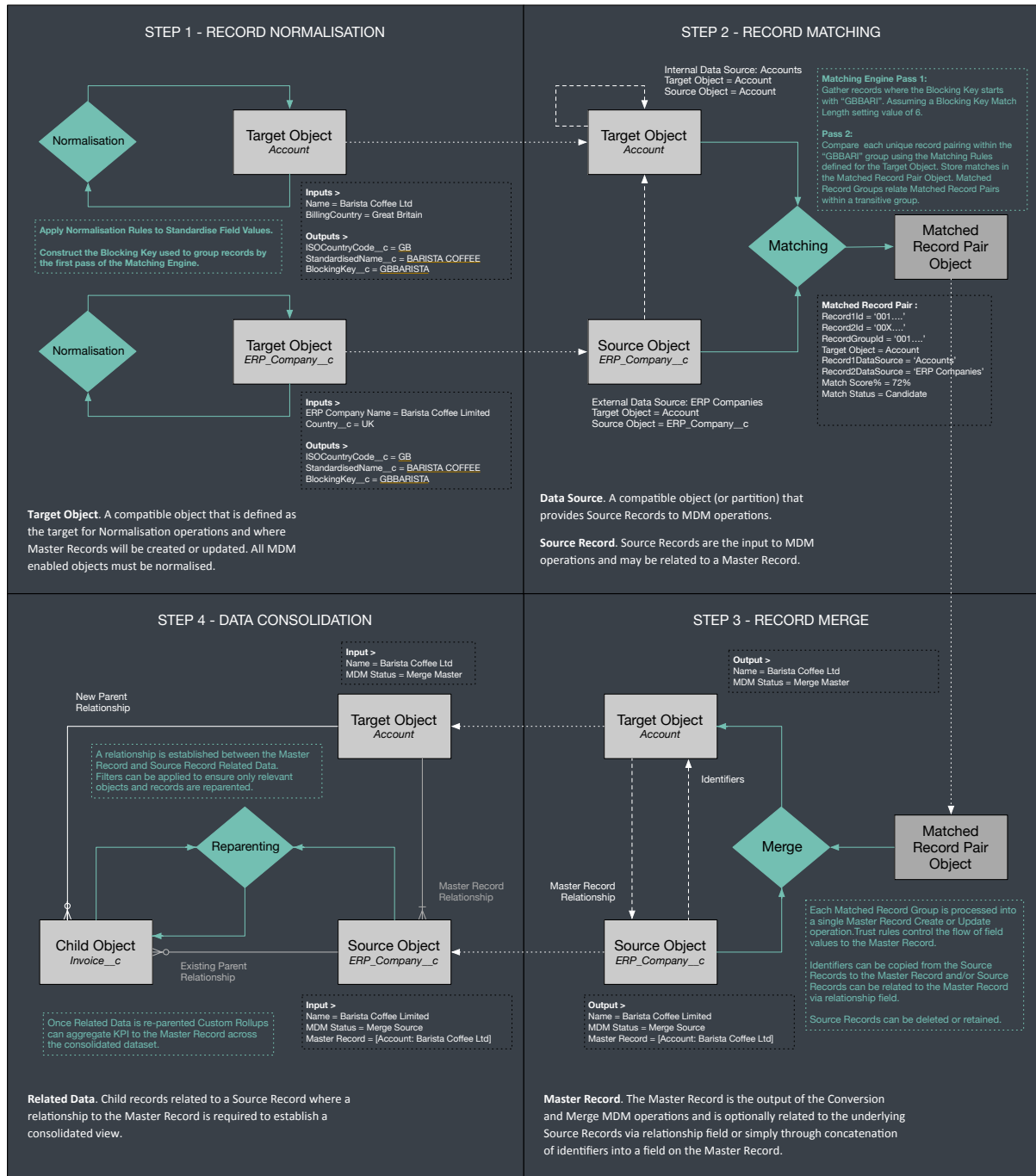The document Appendices also provide additional reference materials.

For practitioner guidance in respect to the implementation of clearMDM please refer to the Implementation Model documentation provided on the website, or upon request.

Table of Contents

CLEARMDM

MDM Process Overview

The diagram below provides an overview of the core MDM operations in sequence. This document relates to STEP1 – RECORD NORMALISATION.

STEP 1 - RECORD NORMALISATION

Normalisation → Target Object *Account*

Inputs >
Name = Barista Coffee Ltd
BillingCountry = Great Britain

Outputs >
ISOCountryCode__c = GB
StandardisedName__c = BARISTA.COFFEE
BlockingKey__c = GBBARISTA

Apply Normalisation Rules to Standardise Field Values.

Construct the Blocking Key used to group records by the first pass of the Matching Engine.

Normalisation → Target Object *ERP_Company__c*

Inputs >
ERP Company Name = Barista Coffee Limited
Country__c = UK

Outputs >
ISOCountryCode__c = GB
StandardisedName__c = BARISTA.COFFEE
BlockingKey__c = GBBARISTA

Target Object. A compatible object that is defined as the target for Normalisation operations and where Master Records will be created or updated. All MDM enabled objects must be normalised.

STEP 2 - RECORD MATCHING

Internal Data Source: Accounts
Target Object = Account
Source Object = Account

Target Object *Account*

Matching Engine Pass 1:
Gather records where the Blocking Key starts with "GBBARI". Assuming a Blocking Key Match Length setting value of 6.

Pass 2:
Compare each unique record pairing within the "GBBARI" group using the Matching Rules defined for the Target Object. Store matches in the Matched Record Pair Object. Matched Record Groups relate Matched Record Pairs within a transitive group.

Matching → Matched Record Pair Object

Source Object *ERP_Company__c*

Matched Record Pair :
Record1Id = '001....'
Record2Id = '00X....'
RecordGroupId = '001....'
Target Object = Account
Record1DataSource = 'Accounts'
Record2DataSource = 'ERP Companies'
Match Score% = 72%
Match Status = Candidate

External Data Source: ERP Companies
Target Object = Account
Source Object = ERP_Company__c

Data Source. A compatible object (or partition) that provides Source Records to MDM operations.

Source Record. Source Records are the input to MDM operations and may be related to a Master Record.

STEP 4 - DATA CONSOLIDATION

New Parent Relationship

Input >
Name = Barista Coffee Ltd
MDM Status = Merge Master

Target Object *Account*

A relationship is established between the Master Record and Source Record Related Data. Filters can be applied to ensure only relevant objects and records are reparented.

Reparenting

Child Object *Invoice__c*

Master Record Relationship

Existing Parent Relationship

Source Object *ERP_Company__c*

Input >
Name = Barista Coffee Limited
MDM Status = Merge Source
Master Record = [Account: Barista Coffee Ltd]

Once Related Data is re-parented Custom Rollups can aggregate KPI to the Master Record across the consolidated dataset.

Related Data. Child records related to a Source Record where a relationship to the Master Record is required to establish a consolidated view.

STEP 3 - RECORD MERGE

Output >
Name = Barista Coffee Ltd
MDM Status = Merge Master

Target Object *Account*

Identifiers

Merge → Matched Record Pair Object

Master Record Relationship

Source Object *ERP_Company__c*

Output >
Name = Barista Coffee Limited
MDM Status = Merge Source
Master Record = [Account: Barista Coffee Ltd]

Each Matched Record Group is processed into a single Master Record Create or Update operation. Trust rules control the flow of field values to the Master Record.

Identifiers can be copied from the Source Records to the Master Record and/or Source Records can be related to the Master Record via relationship field.

Source Records can be deleted or retained.

Master Record. The Master Record is the output of the Conversion and Merge MDM operations and is optionally related to the underlying Source Records via relationship field or simply through concatenation of identifiers into a field on the Master Record.

CLEARMDM

Normalisation Introduction

Definition

Normalisation is the first step of the MDM process and performs two key functions; field standardisation and Blocking Key population. In former case record matching accuracy is improved through the pre-standardisation of key input fields. Examples being country name standardisation to the ISO Country Code (United Kingdom, Great Britain etc. to GB), or removal of spurious special characters or irregularly occurring suffixes (Ltd, LLP, Sr etc.). In the latter case the first-phase of the Matching engine groups records by a common Blocking Key value before finer-grained Matching Rules are applied. The Blocking Key is constructed from characters taken from up to 3 input fields. Records without a complete Blocking Key are not matched at all, records must share a common Blocking Key to be matched together; the Blocking Key structure is therefore a critical decision point when configuring clearMDM that requires understanding of the population characteristics of the target dataset.

Key Concepts

Concept	Definition
Normalisation Settings	Normalisation is configured per Target Object on the Target Object Settings page. **Every object that exposes data for matching must have a Target Object definition and Normalisation settings.** Where cross-object matching is implemented the Blocking Key structure must be uniform across the Target Object settings for all objects. A setting-by-setting definition for the Normalisation Settings is provided in Appendix A.
Normalisation State	A given record is either in a Normalised state or not. The [Is Normalised?] record-level flag is typically configured to record this key state value. The Normalisation MDM operation will set this flag to true on completion, unless Auto Normalisation is applied where the record level flag is ignored. Where Auto Normalisation is not used the record-level flag is used to isolate records that require normalisation, i.e. only records where the flag is false are processed. clearMDM can be configured ("Check Normalisation State on Save?") to manage the flag value; record modifications are evaluated to determine whether a Normalisation significant change has occurred, i.e. any field referenced in the settings or rule definitions has changed. If such a change occurs the

	flag is set to False, i.e. the record is no longer considered to be in a normalised state.
Normalisation Rules	Normalisation Rules take an input field, apply a rule and populate an output field with the product. The output fields can then be utilised in the composition of the Blocking Key and within Matching Rules. The supported Normalisation Rule types are defined in the following subsection.
Blocking Key	The Blocking Key (or Matching Key) provides the initial grouping of records to which Matching Rules are applied. The Blocking Key is constructed from characters taken from up to 3 input fields – the selected fields must have a high population percentage, be static data i.e. not subject to frequent change (non-volatile) and in combination cover no more than approximately 250 records across the dataset. Records without a complete Blocking Key are not matched at all, records must share a common Blocking Key to be matched together; the Blocking Key structure is therefore a critical decision point when configuring clearMDM that requires understanding of the population characteristics of the target dataset.
Blocking Key Statistics	A robust Blocking Key Structure is critical to the ability to perform complete and accurate record matching. Given the stated dependency on the population characteristics of the dataset an empirical mechanism to project the distribution of records across Blocking Keys is required to support an iterative process for the determination of an optimal Blocking Key structure. clearMDM provides a Blocking Key Statistics tool for this purpose, further detail is provided in Appendix B.
Blocking Key Update Action	The Blocking Key Update Action support scenarios where multiple matching passes are required across different Blocking Key constructs. Process automations can be implemented (Process Builder, Flow etc.) that blend job control functions (i.e. Job Submission) with updates to the Blocking Key configuration. For example it may be necessary to match records on an Email based Blocking Key followed by a second pass (for unmatched records) based on Phone number.

CLEARMDM

Normalisation Rule Types

Normalisation Rules take an input field, apply a rule and populate an output field with the product. The supported rule types are outlined in the table below.

Rule Type	Definition
E164	{ 01912584998=>+441912584998} This normalisation rule type takes an input Phone field (any) and formats the value to the E164 format - i.e. +447730780803. The Custom Setting [ISO Country Codes] holds the prefix to add - plus the international dialing codes to check-for and remove first. Within normalisation settings (Target Object) there is a setting to specify which field on the object returns the Country ISO Code; this value is then matched to the custom setting above. The record's Country designation is used to determine the right prefixes to strip from the phone number and the prefix then to add. Typically, a lookup normalisation rule will be used to populate the record-level Country ISO Code field from the Country address field (BillingCountry etc.) A rule of this type is created by default for the Account Target Object on install. The same Custom Setting also translates the country names to the ISO Code. clearMDM installs a field NormalisedPhone__c as a place to output the E164 format rule to. **Note, all clearMDM Normalisation related Custom Settings are installed with a small number of example records only, before use such settings should be fully populated.**
Email Domain	{ john.smith@constosio.com=>contosio.com } Strip the domain from an email address.
Format	{ John smith=>JOHNSMITH } Remove whitespace and special characters such as * } [>.
Lookup	{ United Kingdom, UK=>GB } { Will, Bill=>William } A Custom Setting can be referenced (e.g. ISO Country Codes) that matches the input field value (e.g. BillingCountry = "United Kingdom") to a standarised value ("GB"). The Custom Setting should have a "Match List" field of comma-separated, non-standardised values relating to a

	standardised equivalent recorded in the Name field. The clearMDM package provides the ISO Country Code setting as an example. **Note, all clearMDM Normalisation related Custom Settings are installed with a small number of example records only, before use such settings should be fully populated.** Note, in addition to the example Custom Settings, any Custom Setting defined directly in the customer org can be referenced by Normalisation Rules.
Remove End	{ John Smith Snr=>John Smith } A Custom Setting can be referenced (e.g. Last Name Suffixes) that returns a list of values (e.g. "Ltd" or "Esq") to remove from the input field value.
Title	{ La terassa=La Terrassa } Ensure each word in the text is Title-cased.
Title Hyphen	{ La terassa=La-Terrassa } Ensure each word in the text is Title-cased and words are separated by hyphens not spaces.
URL	{ www.contosio.com=>contosio.com } Strip the domain from a web address (or fully-formed Url).

CLEARMDM

Normalisation Methods

The table below outlines the supported methods for invocation of the Normalisation MDM Operation. Setting references refer to the Target Object Normalisation settings.

Method	Definition
Batch Job	The clearMDM Jobs page can be used to schedule the Normalisation MDM operation to run for a given Target Object immediately or on a scheduled basis. The [Auto Normalise?] setting controls whether all records are normalised, irrespective of Normalisation State, or whether just records that require normalisation are processed. Where a daily batch processing model is implemented for MDM processing, the Normalisation MDM operation will typically be the first job and will invoke the next job in the sequence using the job chaining settings e.g. Matching Settings section, Is Invoked by Normalisation Job? flag. Note, where the volume of records to normalise exceeds 1 million, the process count must be set to 1. In this mode the process partitioning job phase is skipped and the hard platform limit of 1 million queried records is avoided.
On Record Create	Records can be Normalised on creation to ensure standardised field values and the Blocking Key are populated at the time of record creation. This method requires the [Normalise on Create?] setting to be set to True and the Application Settings – Is Triggers Active? setting to be set to True also.
On Record Save	Records can be Normalised on save to ensure standardised field values and the Blocking Key are always updated in line with field value modifications. This method requires the field specified by the [Normalise on Save Field Name] setting to be set to True on each record and the Application Settings – Is Triggers Active? setting to be set to True.
Action	Records can be Normalised by Process Builder as part of a custom process automation. To configure a Process Builder Action for this purpose add an Action with properties set as below. 1. Action Type = Apex 2. Apex Class = "Normalise Records Action" 3. Record ID parameter = Reference [Object Id field]. The Action can also be implemented within Visual Workflow.

API	Records can be Normalised by a custom action exposed via the standard Force.com REST API; endpoint below.
	
	`/services/data/v38.0/actions/custom/apex/clearmdm__`
	`RecordNormalisationAction`
	
	The API operation takes a single recordId parameter. Further details can be found in the clearMDM API Guide.

CLEARMDM

Step 1 – Configure Application Settings

Pre-requisite: clearMDM must be set to Active via the Application Settings page.
Pre-requisite: clearMDM must be set with Triggers Active via the Application Settings page.

To complete this step, first navigate to the MDM App, open the Settings tab, tick the two fields highlighted below and click the Save button.

CLEARMDM
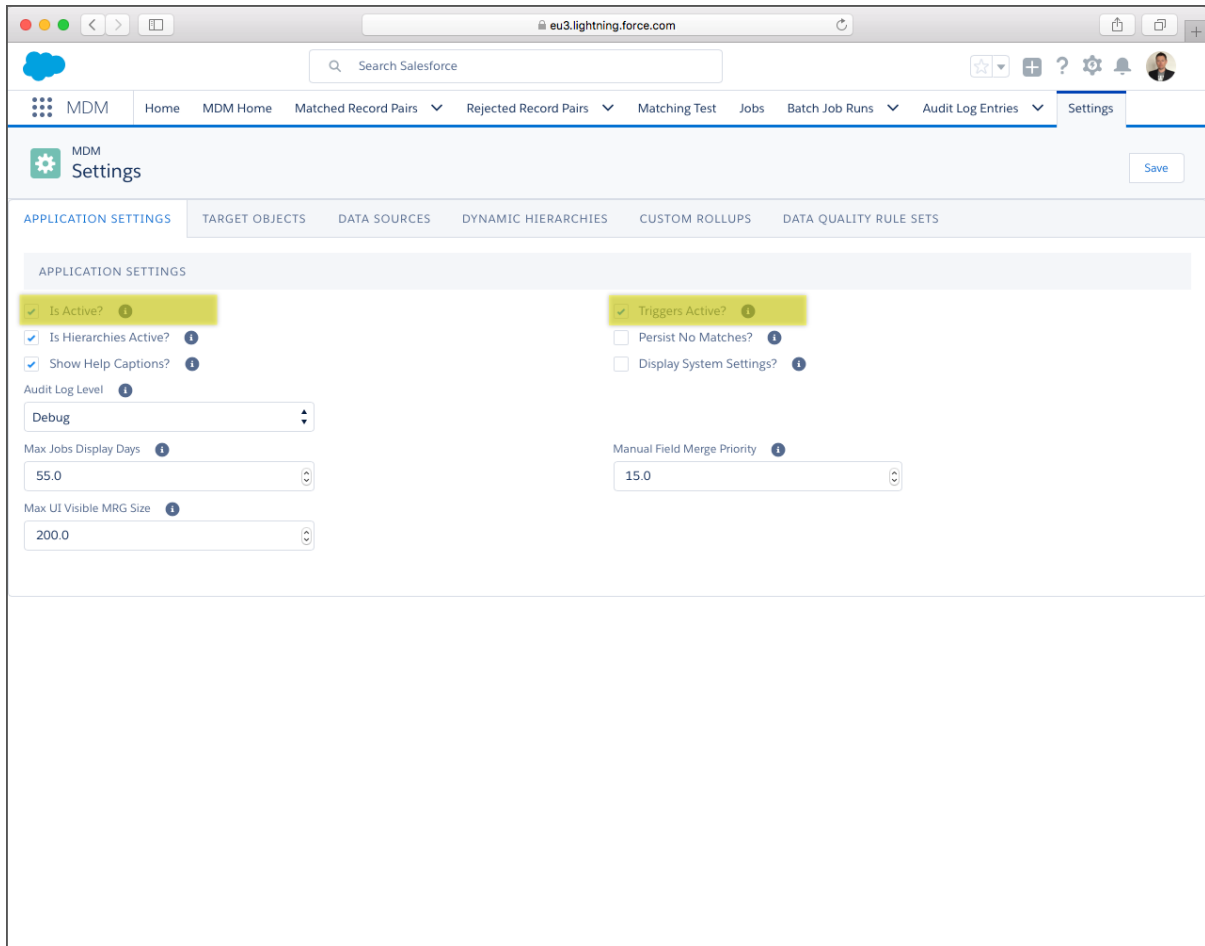
Step 2 – Configure Target Object Settings

To complete this step, first navigate to the MDM App, open the Settings tab, select the Target Objects tab and click the Edit link next to the required Target Object (or dropdown to the right if using Lightning Experience). By default, Account, Contact and Lead are configured.

Next, set the [Is Active?] flag equal to true and ensure the relevant settings are configured correctly. Appendix A provides a Settings reference. Click Save to store the changes.

CLEARMDM

Step 3 – Configure a Basic Normalisation Rule

To complete this step, first navigate to the MDM App, open the Settings tab, select the Target Objects tab and click the Edit link next to the required Target Object (or dropdown to the right if using Lightning Experience). On the Target Object settings page click the Rules tab.

Next, set the [Is Normalised?] flag equal to true next to the relevant field and select the required rule type (i.e. Format). Finally, select the Target Field and click the Save button.

In the example below the First Name is the source field, the Rule Type is Format (which removes special characters and standardises casing) and the Target Field for the rule is the Contact Normalised First Name field.

CLEARMDM
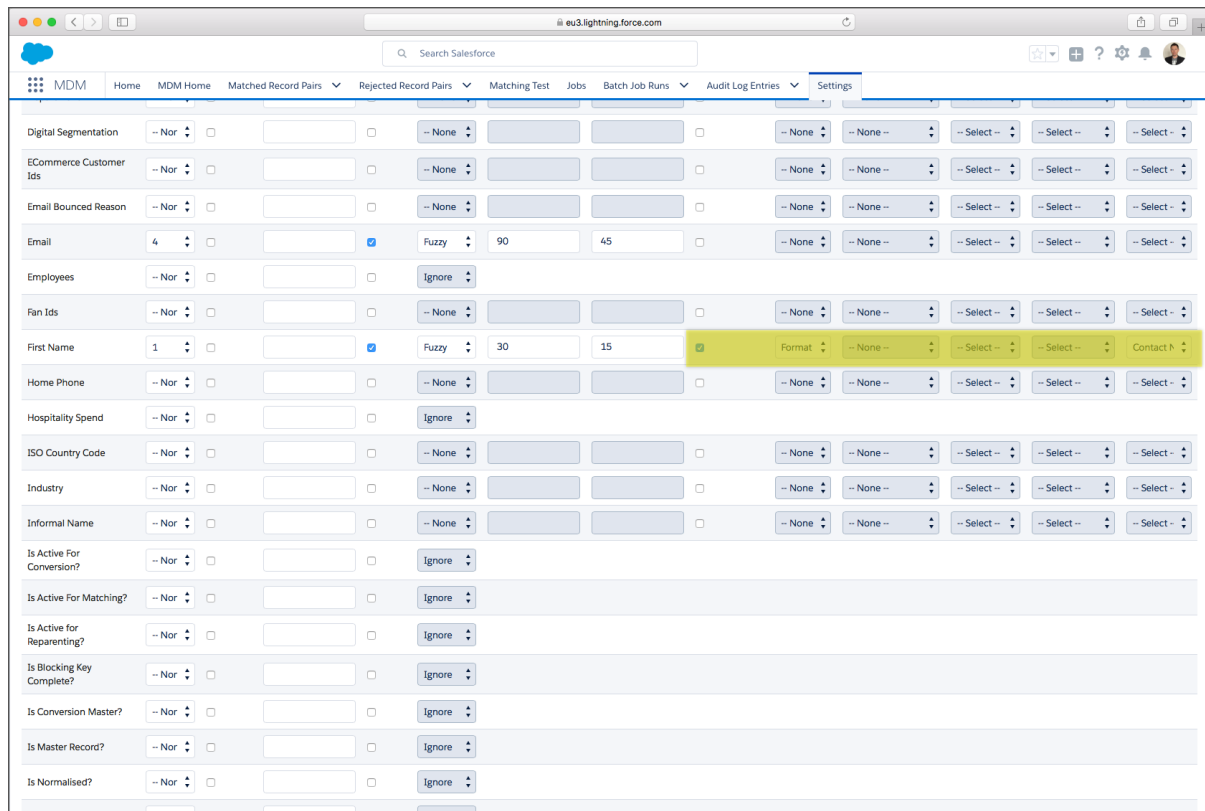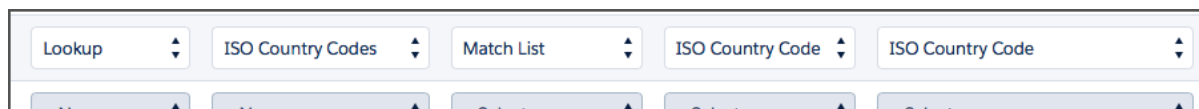
Step 4 – Configure a Lookup Normalisation Rule

To complete this step, first navigate to the MDM App, open the Settings tab, select the Target Objects tab and click the Edit link next to the required Target Object (or dropdown to the right if using Lightning Experience). On the Target Object settings page click the Rules tab.

Next, set the [Is Normalised?] flag equal to true next to the relevant field and select the required rule type (i.e. Lookup).

Next, set the Reference Setting, Lookup field (Match List), Reference Field (Standard Value Field Name), and the Target Field and click the Save button.

In the example below the Mailing Country is the Source Field, the Rule Type is Lookup and the Target Field for the rule is the ISO Country Code field.

Step 5 – Configure the Blocking Key

To complete this step, first navigate to the MDM App, open the Settings tab, select the Target Objects tab and click the Edit link next to the required Target Object (or dropdown to the right if using Lightning Experience).

Next, set the required Blocking Key settings as highlighted below. Appendix A provides a Settings reference. Finally click Save to apply the configuration change.

CLEARMDM

Step 6 – Run the Normalisation Job

To complete this step, first navigate to the MDM App, open the Jobs tab, select the Normalisation Job type and the required Target Object. Finally select the required Job Schedule and click the "Schedule Job" button. This will display in the Scheduled Jobs tab until complete.

Step 7 – View Normalisation Results

The screenshot below shows the Normalisation MDM fields displayed on an example record. Placement of MDM fields on layouts for administrators, data stewards and end-users is a key implementation decision.

CLEARMDM

Appendix A - Normalisation Settings Reference

Normalisation settings are configured per Target Object on the Target Object settings page.

Setting	Definition
Is Active?	If set to False the Target Object will be excluded from all Normalisation MDM operation.
Normalise on Create?	If set to True, new records will be Normalised automatically at the time of creation irrespective of how the record is created.
Auto Normalise Records?	If set to True, all records will be processed by the Normalisation MDM operation when executed as a Job. If set to False, only records where the record-level [Is Normalised?] flag is set to False will be processed.
Auto Normalise Recently Modified?	Where [Auto Normalise Records?] is set to True processing can be limited to records where the last modified date is today or yesterday.
Is Normalised Field Name	The field on the Target Object in which the record-level Normalisation State flag is held (True or False).
Check Normalisation State on Save?	If set to True, records modifications will be evaluated to determine whether a Normalisation significant change has occurred, i.e. any field referenced in the settings or rule definitions has changed. If such a change occurs the Normalisation State is set to False, i.e. the record is no longer considered to be in a normalised state.
Normalise on Save Field Name	A checkbox field on the Target Object which when set to True on a record forces normalisation to run for that record when saved. Once normalisation has completed the field will be reset to False.
Last Normalised Date Field Name	The datetime field on the Target Object in which the last normalised date and time will be set.
ISO Country Code Field Name	The field on the Target Object which is populated with the ISO Country Code (typically by a Lookup rule – such rules run before the E164 rule type).
Blocking Key Field Name	The field on the Target Object in which the Blocking Key is populated.
Blocking Key Group Field Name	The field on the Target Object in which the Blocking Key Group is populated.
Blocking Key Input 1 Field Name	A field on the Target Object which supplies the 1st Blocking Key Input.
Blocking Key Input 1 Length	The number of characters to taken from input 1.
Blocking Key Input 2 Field Name	A field on the Target Object which supplies the 2nd Blocking Key Input.
Blocking Key Input 2 Length	The number of characters to taken from input 2.
Blocking Key Input 3 Field Name	A field on the Target Object which supplies the 3rd Blocking Key Input.
Blocking Key Input 4 Length	The number of characters to taken from input 3.
Is Blocking Key Complete Field Name	The field on the Target Object in which the record-level Is Blocking Key Complete flag is held (True or False). Only records with a complete blocking key are matched. Records can have an incomplete blocking key where input fields are not populated or the length is less than the defined input length and padding is not applied.
Blocking Key Complete Strict?	If set to True, all Blocking Key Input field values must be populated and be at least the defined input length, otherwise the blocking key is not constructed.
Blocking Key Input Padding?	If set to True, padding characters '_' are added where Blocking Key Input field values are populated but less than the defined input length. Padding allows a blocking key to constructed in cases where minor adjustments are sufficient. Padding will not address cases where the input field is blank.

CLEARMDM

Appendix B – Blocking Key Statistics

The Blocking Key Statistics job calculates a distribution of records across the Blocking Key Match values generated by the current Normalisation and Matching settings.

The purpose of the job is to enable analysis of the Blocking Key Match value distribution. Key scenarios to identify are listed below.

1. High Selectivity. Each BKMV grouping contains a very small number of records and therefore matching will be limited.
2. Low Selectivity. BKMV groupings that contains a high number of records (more than 500). Such groups can't be processed by clearMDM with the default settings and will be skipped by the Matching engine.

The Blocking Key Statistics job records the distribution data in the Blocking Key Statistics object accessible by Tab or by standard Salesforce reporting. The key fields of this object are listed below.

- Target Object
- Data Source
- Blocking Key Group Value
- Blocking Key Match Value
- Master Records
- All Records

Where selectivity issues are identified, an incremental approach can be taken where Blocking Key structure and Blocking Key Match Value length changes can be applied and the statistics job re-run.

CLEARMDM

Appendix C – Troubleshooting

All clearMDM MDM operations log activity (Start and End times etc.) and errors to the Audit Log Entry object. The MDM application includes an **Audit Log Entry** tab to provide convenient access to this data. Each Audit Log Entry record is time-stamped and related to the parent Batch Job Run record: all MDM operations that run via the Job Method created a Batch Job Run record that records job statistics and status. Monitoring of the Audit Log should be a frequent activity performed by the Administrator or Data Steward responsible for the clearMDM implementation.

> **Best Practice:** Salesforce Reporting Notifications provide a proactive means of reporting on errors generated by clearMDM operations. In this model a standard report is used to return data from the Audit Log Entry object where required conditions are met. Only when records meet the criteria is a report sent to the Administrator or Data Steward responsible.

Normalisation Log Types

Rule Type	Definition
Record Count Exceeds Limit	The number of records in the Target Object exceeds the edition limit.
Normalisation Rule Lookup Failed	The Custom Setting (ISO Country Codes etc.) referenced by a Lookup rule does not contain a standardised value for non-standardised value encountered (on a record). Each instance of a missing value is logged once only. Administrators can use this information to identify cases where additional standardised values are necessary.
Missing or Inactive Target Object Normalisation settings	This error can occur where the Target Object Normalisation settings are inactive, deleted or the Salesforce User does not have permissions to the object or fields referenced.