CLEARMDM

QuickStart Guide 3 - Matching

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 Matching 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.

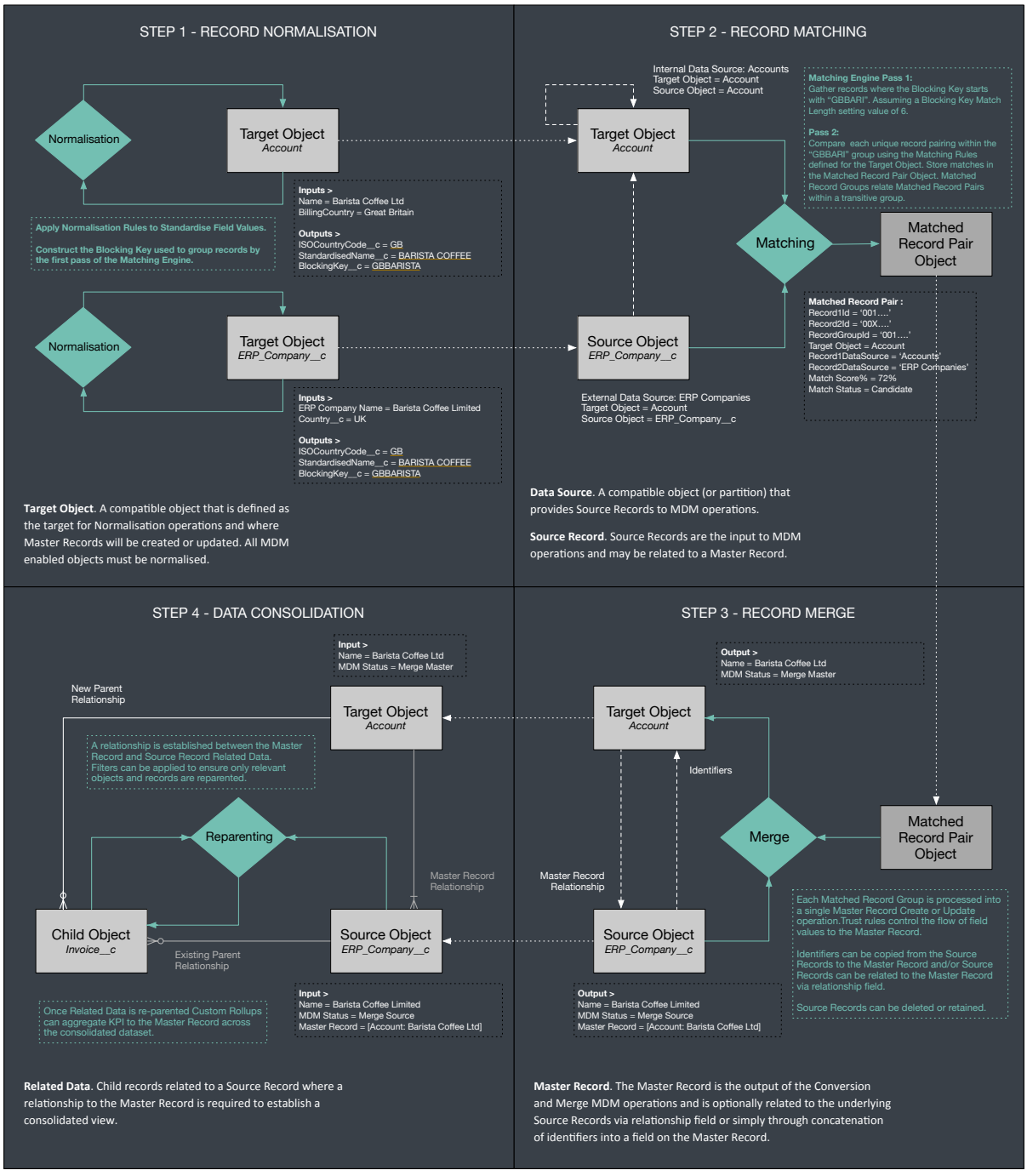
Table of Contents

clearMDM® is a registered trademark of Audit9 Ltd a company registered in the United Kingdom with company number 08088394.

CLEARMDM

MDM Process Overview

The diagram below provides an overview of the core MDM operations in sequence. This document relates to STEP2 – RECORD MATCHING.



clearMDM® is a registered trademark of Audit9 Ltd a company registered in the United Kingdom with company number 08088394.

Matching Introduction

Definition

Record Matching is the process of identifying potential duplicate records within the Salesforce dataset. In summary, the Matching MDM operation applies Matching Rules (Fuzzy, Exact, Key etc.) to identify groupings where each record has a common **Blocking Key Match Value**. Within a given record grouping every record is matched against every other record (i.e. the Cartesian product) by **Matching Rules** and a matching score is calculated for each record pair. If the matching score for a given pair of records is above the matching threshold setting value (for the Target Object), then a **Matched Record Pair** record (MatchedRecordPair__c) is created with the Match Status set to "Candidate". If the matching score for a given pair of records is above the auto-accept threshold setting value (for the Target Object), then a **Matched Record Pair** record is created with the Match Status set to "Accepted". **Matching Record Groups** are created where multiple records for the same Blocking Key Match Value match in a transitive manner, i.e. record A matches to B and record B matches to record C; the resultant **Matched Record Group** will contain both **Matched Record Pairs**. The RecordGroupId__c field on MatchedRecordPair__c object is populated with the Matched Record Group Id. The Merge MDM operation processes each individual **Matched Record Group** into either a **Master Record** create or update operation; **Matched Record Groups** that contain "Candidate" status **Matched Record Pairs** are skipped; such groups require stewarding before merge processing.

For further information in relation to the **Blocking Key** and related concepts please refer to the document *clearMDM – QuickStart Guide 2 – Normalisation*.

Key Concepts

Concept	Definition
Target Objects	A compatible object that is defined as the target for Matching operations and where Master Records will be created or updated. Account, Contact, Lead Standard Objects are typically configured as Target Objects, where duplicate records may exist directly in the object or indirectly in a separate object. A broad range of **Standard Objects** (including Person Accounts) are supported as both Target Objects and Data Sources. **Custom Objects** are also supported.
Data Sources	A compatible object that provides data to MDM operations. Each Data Source has a Source Object and a Target Object setting. Internal Data Sources expose data held in the Target Object. **Data Source (Account) > Target Object (Account)** External Data Sources expose data held in a different object. **Data Source (ERP Companies) > Target Object (Account)** **Partition Data Sources** enable a single object to support multiple Data Sources with distinct settings. Partition Data Sources are typically used to group records relating to an external system (e.g. SAP, Sage X3) or to isolate records at different quality grades (e.g. High, Medium and Low).

clearMDM® is a registered trademark of Audit9 Ltd a company registered in the United Kingdom with company number 08088394.

	Master Record Data Sources allow Master Records to be exposed to MDM operations via a distinct Data Source with appropriate settings such as elevated merge field priorities. A Master Record Partition data source is implemented as a partition data source that references the record MDM status value. **Check-only Data Sources** support cross-object matching between standard objects. The primary use case for Check Only Data Sources is duplicate checking across Leads, Contacts and/or Person Accounts. For example, Lead creation (file import, UI data entry etc.) can be blocked where the Lead matches to an existing Contact.
Master Records	The **Master Record** is the output of MDM processing and is optionally related to the underlying Source Records (or duplicates) via relationship field or simply through the concatenation of record identifiers into a field on the Master Record. Most typically a Master Record is a de-duplicated Account, Person Account, Contact or Lead record enriched with data from its related Source Records. **Where Source Records (or duplicates) are not removed, it is typical to use the Salesforce sharing model to restrict Salesforce end-user record access to Master Records exclusively.**
Source Records	**Source Records** are the input to MDM operations and may be related to a Master Record. Source Records can be retained (persistent model) or deleted (transient model). Source Records can be considered as the underlying duplicates that are often hidden from view for Salesforce end-users or deleted entirely once processed.
Matching Settings	Matching is configured per Target Object on the Target Object Settings page and also at the Data Source level on the Data Source settings page. A setting by setting definition for the Matching Settings is provided in Appendix A.
Matched State	A given Source Record is either exposed to the Matching Engine or not. The **[Is Active for Matching?]** field (record-level flag) is typically configured to express this key activation state value. The Matching MDM operation will set the flag to False on completion, unless Auto Matching is applied (where the record-level flag is ignored). Where Auto Matching is not used the record-level flag is used to isolate records that require matching, i.e. only records where the flag is true are processed. clearMDM can be configured ("Check Matching State on Save?") to manage the flag value; record modifications are evaluated to determine whether a Matching 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 True, i.e. the record has changed materially and requires either re-Matching or Synchronisation.
Blocking Key Match Value	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. The Blocking Key is populated by the Normalisation MDM operation. The **Blocking Key Match Value** is a defined subset of the full **Blocking Key** populated by the Normalisation MDM operation. The number of characters used controls the selectivity of the matching behaviour. The Blocking Key Match Value (or Matching Key) provides the initial grouping of records to which Matching Rules are applied. For example; Blocking Key = GBFEDEXSW Blocking Key Match Value Length = 5 Blocking Key Match Value = GBFED In the above case, records only require the first 5 characters to be common, enabling matching between records that have GBFEDEXSW and GBFEDCONE blocking keys – for example. Where the GBFED group size exceeds the configured processing limit, additional characters will be added to create sub-groupings i.e. GBFEDE and GBFEDC. Auto-adjustment of this type can be applied automatically by the Matching engine and requires the full Blocking Length to be greater then the Blocking Key Match Value Length. Records without a complete Blocking Key are not matched at all, records must share a common Blocking Key Match Value 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.
Matching Algorithm	**Phase 1: Blocking Key Match Value** • Group Source Records by Blocking Key Match Value (BKMV) – where the Data Source setting **[Auto Match Records?]**=True, or the record-level flag **[Is Active for Matching?]**=True. • For each BKMV add Master Records where the Data Source setting **[Master Record Active for Matching?]**=True. **Phase 2: Matching Rules** • For each BKMV group attempt to match together each distinct pairing of records (Cartesian product). • Where the Match Score is above the configured Match Score % create a **MatchedRecordPair__c** record at "Candidate" or "Accepted" status. • Structure the **MatchedRecordPair__c** records into Matched Record Groups via the RecordGroupId__c field, where a

	transitive relationship exists. **Note, a single Source Record can exist in one Matched Record Group only.** • Reset the **[Is Active for Matching?]** flag to False, if the Data Source setting **[Auto Match Records?]**=False. • Set the record level **MDM Status** field to "Matched" or "No Match" depending on whether matches have been identified for the record – finally update the **Last Matching Date** field. **Note, Source Records with a completely unique Blocking Key Match Value are skipped by the Matching Engine and will retain a blank MDM Status.**
Matching Rules	Matching Rules take an input field, apply a rule and calculate a direct outcome (Key Match or Deterministic Fail) or field-level fuzzy score. Field-level fuzzy scores are aggregated to the record level to determine the record level matching score against which the configured score threshold is compared to determine the match outcome (Match=Candidate, Match=Auto Accepted, No Match). The supported Matching Rule types are defined in the following subsection.
Matching Results	Each matched record pair is recorded in the **MatchedRecordPair__c** object. For each record the following attributes are stored: • Record 1 Id + Name + Data Source • Record 2 Id + Name + Data Source • Target Object • Threshold Match Score % • Actual Match Score % • Match Status (Candidate, Rejected, Accepted) • Blocking Key Match Value • Match Data (Field by Field Score Calculations) • Record Group Id (Matched Record Group) The Merge MDM operation processes each Matched Record Group into either a **Master Record** create or update operation. However, Matched Record Groups that contain "Candidate" status pairings will be skipped. It is therefore a key **Data Stewarding** requirement that Candidate matches are reviewed and either Accepted or Rejected in order for the group to be processed. Auto-acceptance settings provide control over the level of manual intervention required.
Auto Acceptance	To provide control over the level of effort required for manual Data Stewarding, auto-acceptance can be implemented. A second (higher) threshold score % is defined, Matched Record Pairs that score above this level will be set to Accepted (rather than Candidate). In the example below, matches above 65% and below 75% will require manual data stewarding. Fuzzy Match Threshold % = 65%

clearMDM® is a registered trademark of Audit9 Ltd a company registered in the United Kingdom with company number 08088394.

	Auto Accept Match Threshold % = 75%
Match Rejection	The record Id pairing within **Matched Record Pairs** that are explicitly set to the status "Rejected" are recorded in the **RejectedRecordPair__c** object. This object is referenced by the Matching Engine to ensure the pairing is not matched again.
Internal Matching	In cases where a large volume of existing records must be processed (in excess of 1 million) a specialised matching job is provided that can scale to 50 million records in a single operation. The Internal Matching job must be run with a process count equal to 1 at record levels beyond 1 million. Internal Matching references the internal Data Source settings for the given Target Object; external and partition data sources are ignored. Cross-object matching is also not implemented.

Matching Rule Types

Matching Rules are defined **per-field** on the Target Object and applied one-at-a-time to each record pair comparison for a given Blocking Key Match Value.

Matching Rules are evaluated individually and return one of the following outcomes;

1) The Record Pairing is confirmed as a **Match** (due to a commonality of field values for a particularly discriminating field such as Social Security Number or Mobile Phone Number).
2) The Record Pairing is confirmed as a **No Match** (due to the absence of commonality of a required field value across both records, for example Gender).
3) A Field Level Score is returned and added to the Record Level Match Score.

The supported rule types are outlined in the table below.

Rule Type	Order	Definition
Key	1	If the two field values (on the record pair) are equal then the record pairing is deemed to be Key match at 100%. Key rules can be used on single fields or formula fields that concatenate a number of inputs to return a Key Matching Value upon which a Key rule is applied. Key rules run before Deterministic and Fuzzy rules. Note, where field values contains a pipe (|) delimited list then matching will be applied to each individual value.
Deterministic	2	If the two field values (on the record pair) are not equal then the record pairing is deemed to be a Non-match.

		Deterministic rules provide a means to rule-out matches before the application of Fuzzy rules. Gender is often a good example for the Deterministic rule type.
Fuzzy	3	The two field values (on the record pair) are evaluated through the Edit distance algorithm to determine a field-level matching score which is a % of the Max Field Score setting based on the number of edit operations required to align the values from the total possible operations. Where one or both values are blank the Null Field Score setting value is returned. The field-level fuzzy (and exact) matching scores are aggregated to the record-level to return an overall record matching score for the record pair.
Exact	3	The two field values (on the record pair) are evaluated for an exact match, where this is the case the Max Field Score setting is returned. Where one or both values are blank the Null Field Score setting value is returned.
Ignore	0	Fields that are relevant to Merge but have no Matching significance should be set with the Ignore type. Conversely fields that are Matching significant but must not be merged are should be set with the "Matching Only?] flag equal to True.

Matching Methods

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

Method	Definition
Batch Job	The clearMDM Jobs page can be used to schedule the Matching MDM operation to run for a given Target Object immediately or on a scheduled basis. The **[Auto Match Records?]** Data Source setting controls whether all records are matched, irrespective of Matched State, or whether just records that require matching are processed. Where a daily batch processing model is implemented for MDM processing, the Matching MDM operation will typically be the third job and will invoke the next job in the sequence using the job chaining Target Object settings e.g. Merge Settings section, **[Is Invoked by Matching Job?]** flag. In addition to the standard Matching job, a second matching job named "**Internal Matching**" can be enabled (via Target Object Matching Settings). This job is intended for processing large data volumes (multiple million records) during initial migration and is limited to processing records in a single object only. This job applies the internal Data Source settings for the specific Target Object only, partition Data Sources are ignored. Note, the process count for this

	job must be set to 1 at record volumes greater than 1 million. Cross-object matching rules are not processed by the Internal Matching job.
On Record Create	Records can be Matched on creation to prevent creation of duplicate records. This method requires the **[Check for Matches on Record Creation?]** Target Object setting to be set to True and the Application Settings – **[Is Triggers Active?]** setting to be set to True also. This behaviour can be bypassed for specific field conditions or user profiles.
On Record Save	Records can be Matched on save to prevent creation of a duplicate via record modification. This method requires the field specified by the **[Matching Check on Save Field Name]** Target Object setting to be set to True on each record and the Application Settings – **[Is Triggers Active?]** setting to be set to True. This behaviour can be bypassed for specific field conditions or user profiles.
Action	Records can be Matched 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 = "Match Records Action" 3. Record ID parameter = Reference [Object Id field]. The Action can also be implemented within a Visual Workflow (or Flow).

API	Records can be Matched by a custom action exposed via the standard Force.com REST API; endpoint below.
	
	`/services/data/vXX.0/actions/custom/apex/clearmdm__RecordMatchingAction`
	
	The API operation takes a single recordId parameter. Further details can be found in the clearMDM API Guide.
	
	A second API operation "Quick Matching" extends point-of-entry checking out to external systems in a distributed MDM model.
	
	`/services/apexrest/clearmdm/1.0/QuickMatching`
	
	The API operation takes record attributes such as Name and Address to enable matching plus the list of fields to return on matched results.
	
	Further details can be found in the *clearMDM API Guide*.

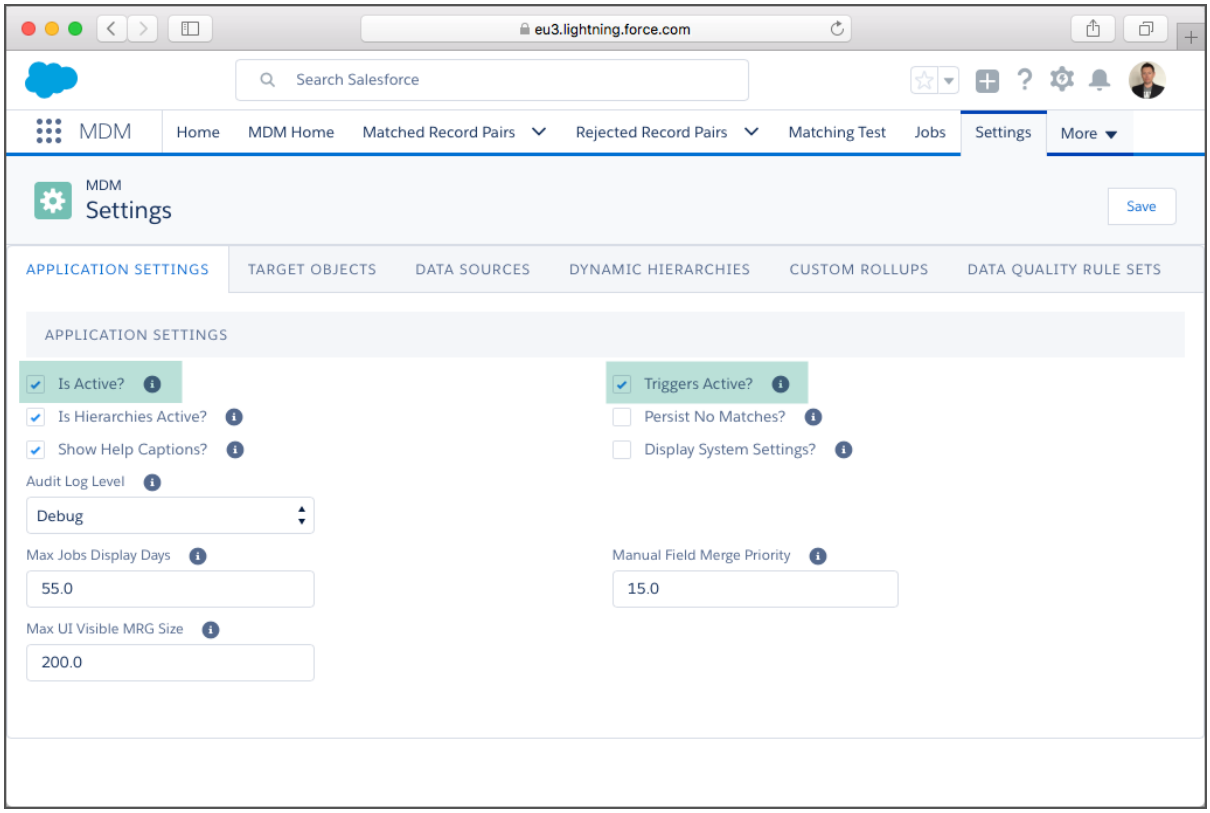
UI	Record Matching can be applied manually via the 2 pages below. **FindMatches**

	The “Find Matches” page is invoked via Url with the format below. The clearMDM packaged layouts for Account, Contact and Lead contain Custom Buttons for convenient access. `/apex/FindMatches?id=[recordId]` The Find Matches page supports 2 real-time search scenarios: 1. “Find”: Find matches that are a direct match to the given record. 2. “Find All”: Find all matches for the given record’s Blocking Key Match Value. Matching results can be manually saved as **Matched Record Pairs** within **Matched Record Groups**. Such groups will be processed by the next Merge MDM operation job or can be manually merged via the Merge page. **Matching Test** The “Matching Test” page is accessed via a tab in the MDM app. A record search can be performed (across data sources) using a compound field filter. Two records from the search results can be selected and compared using the current matching settings for the Target Object. The Matching Test page allows the settings to experimented with using exemplar records; refined settings can be applied directly to the Target Object settings. The two selected records can also be manually merged irrespective of the Blocking Key commonality or matching score.

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.

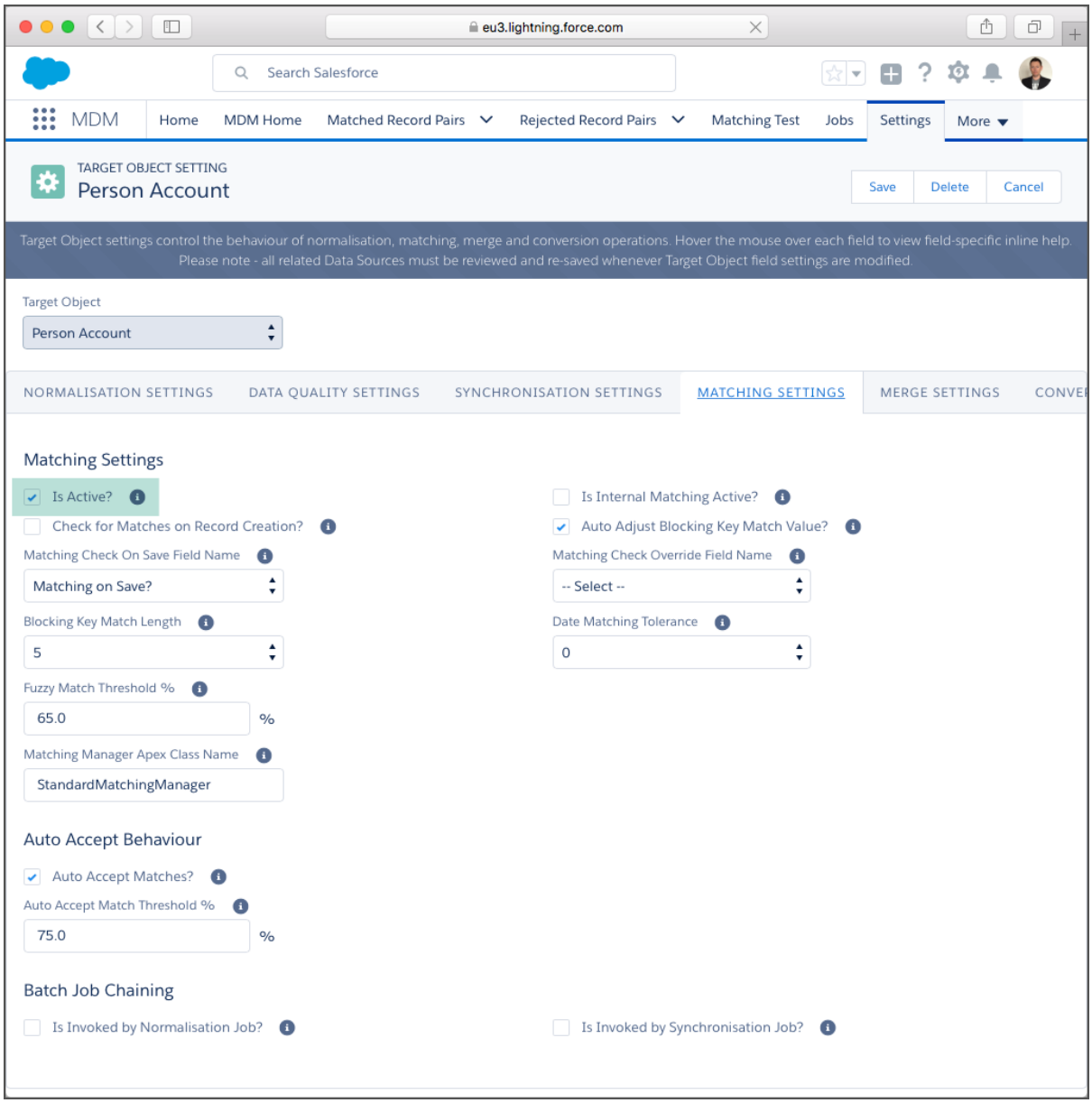


clearMDM® is a registered trademark of Audit9 Ltd a company registered in the United Kingdom with company number 08088394.

Step 2 – Activate the Target Object for Matching

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. Note, for Lightning Experience the Edit menu is accessible via the Dropdown menu in the rightmost table column.

Next, set the [Is Active?] flag equal to true in the Matching Settings section (or Matching Settings tab in Lightning Experience) and ensure the relevant settings are configured correctly. Appendix A provides a Settings reference. Click Save to store the changes.

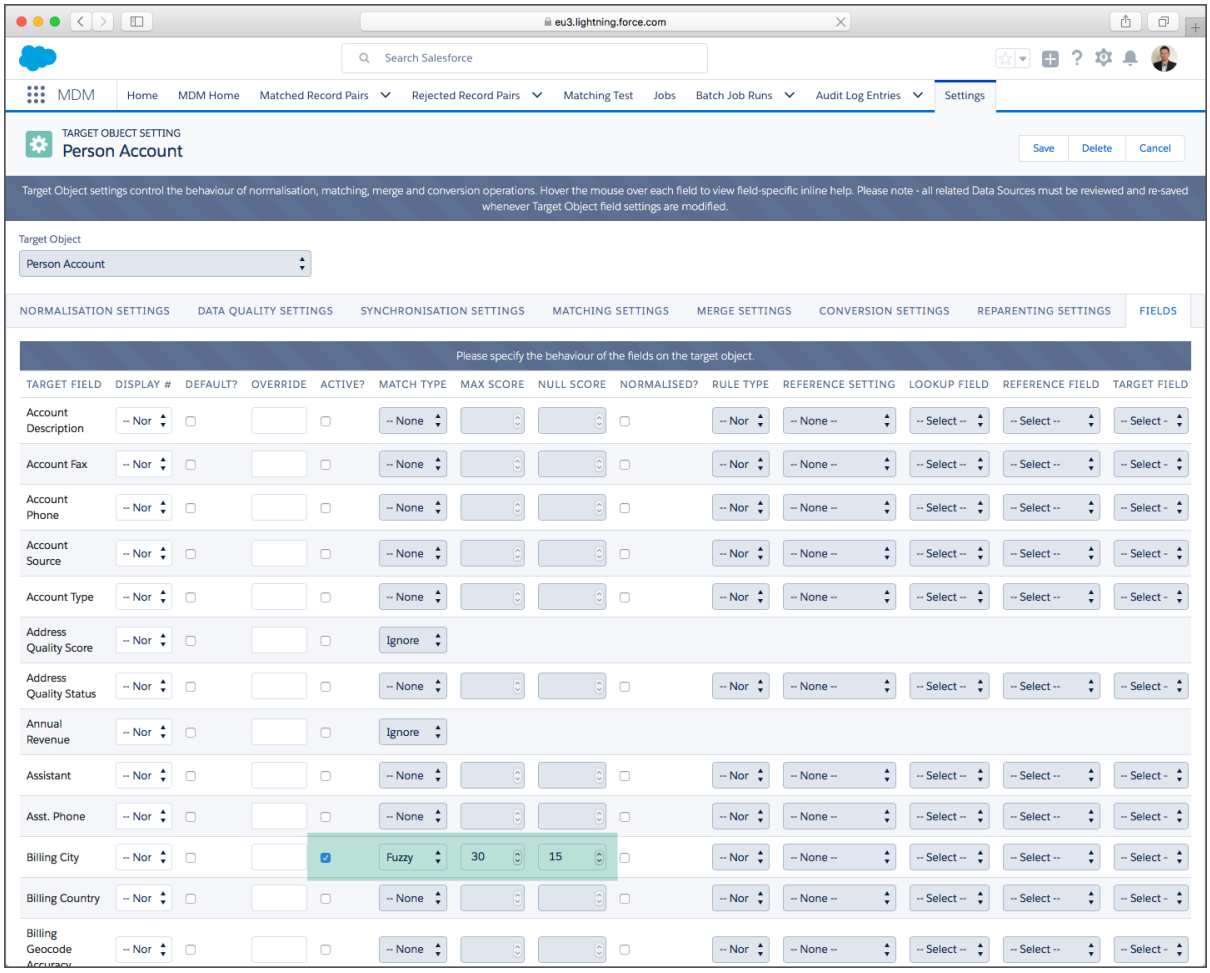


Step 3 – Configure Matching Rules

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. Note, for Lightning Experience the Edit menu is accessible via the Dropdown menu in the rightmost table column. On the Target Object settings page click the Fields tab.

Next, set the [Is Active?] flag equal to true next to the relevant field and select the required Matching Rule type (e.g. Key). For the Fuzzy and Exact rule types a Max Score and Null Score value must be entered.

Finally click the Save button.

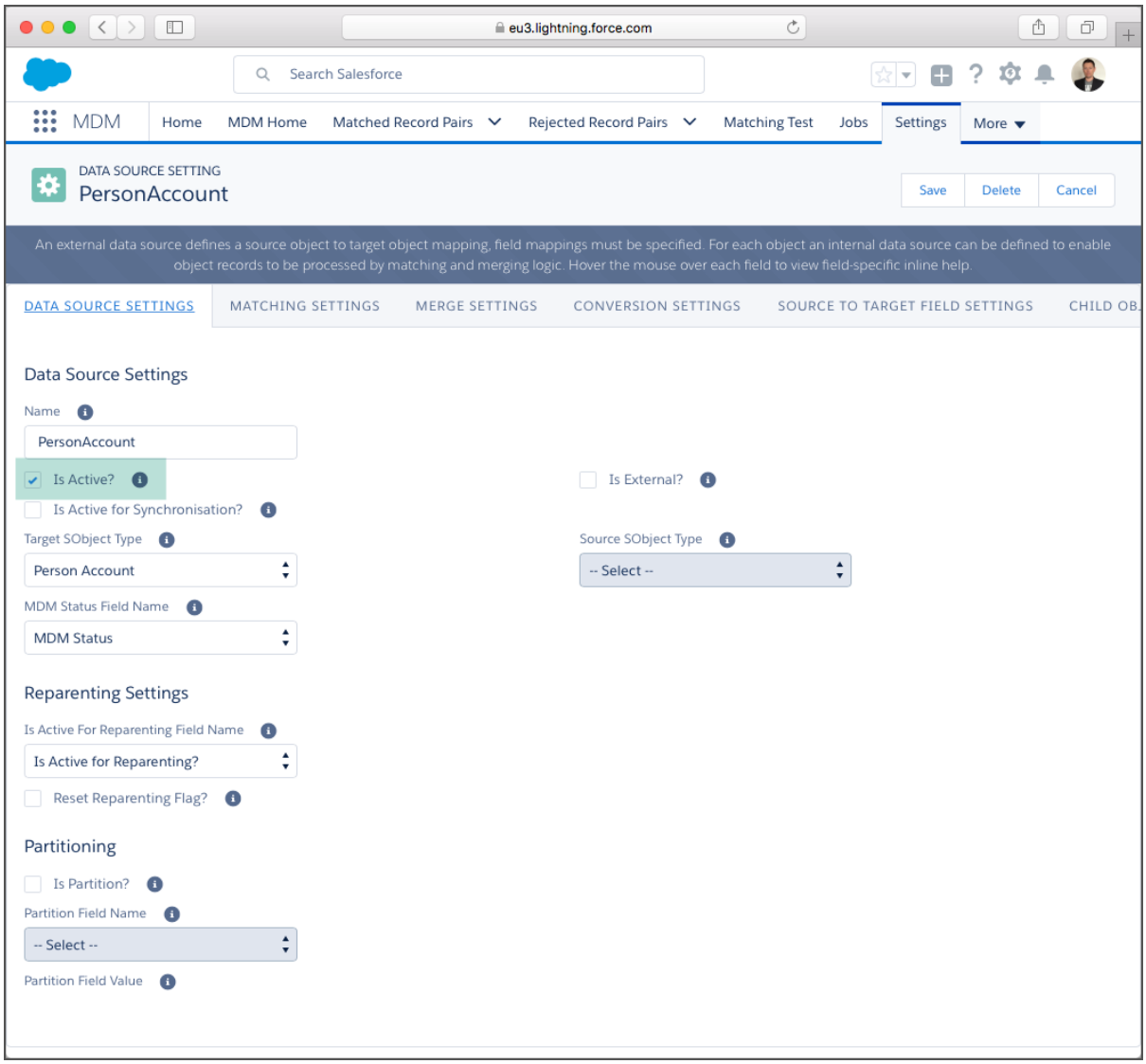


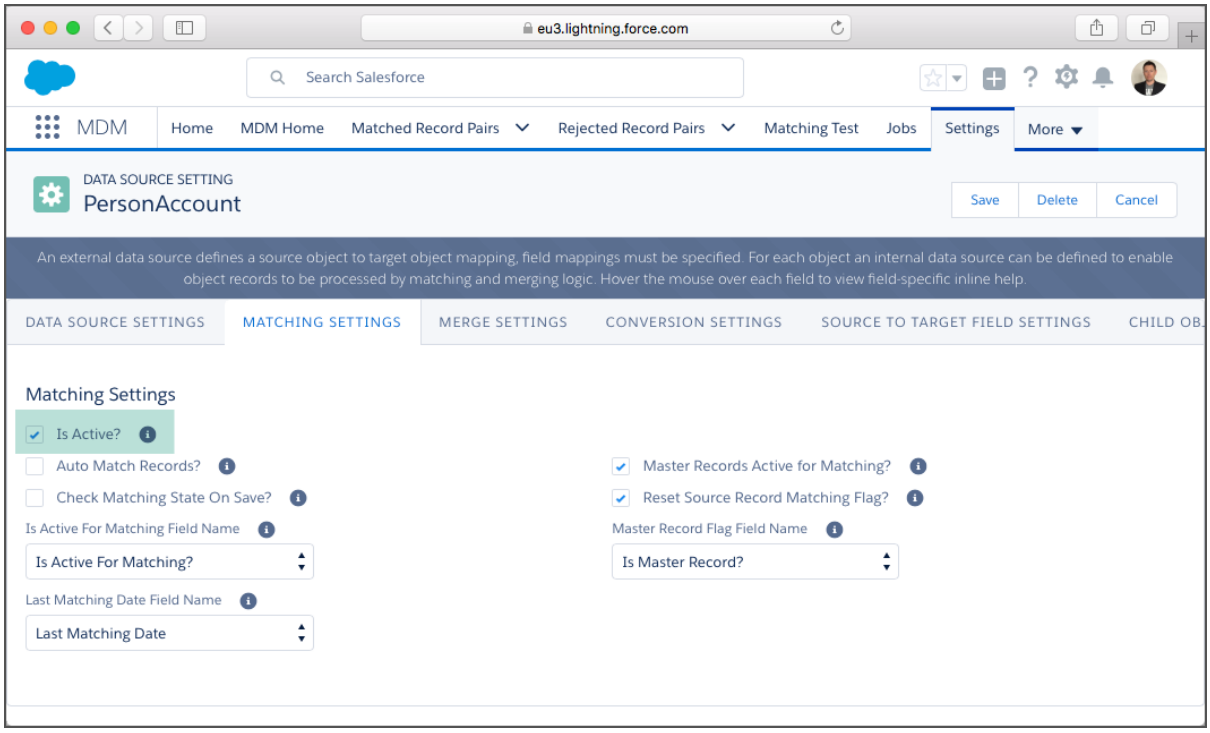
Step 4 – Activate the Data Source for Matching

To complete this step, first navigate to the MDM App, open the Settings tab, select the Data Sources tab and click the Edit link next to the required Data Source. Note, for Lightning Experience the Edit menu is accessible via the Dropdown menu in the rightmost table column.

Next, set the [Is Active?] flags equal to true in the Data Source Settings and Matching Settings sections (or tabs in Lightning Experience) and ensure the relevant settings are configured correctly. Click Save to store the changes.

Appendix A provides a Settings reference.

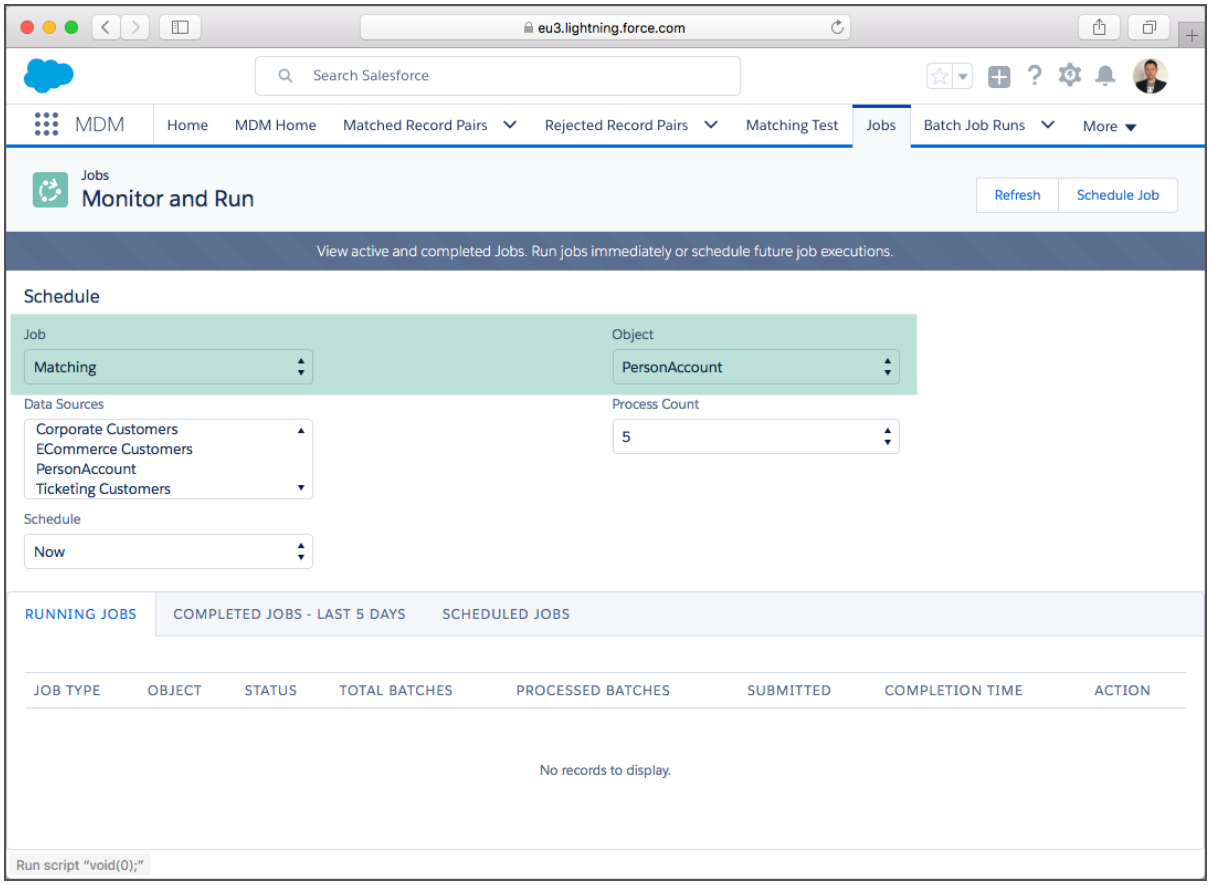




NOTE: The Master Record Flag Field Name is not set by default.

Step 5 – Run the Matching Job

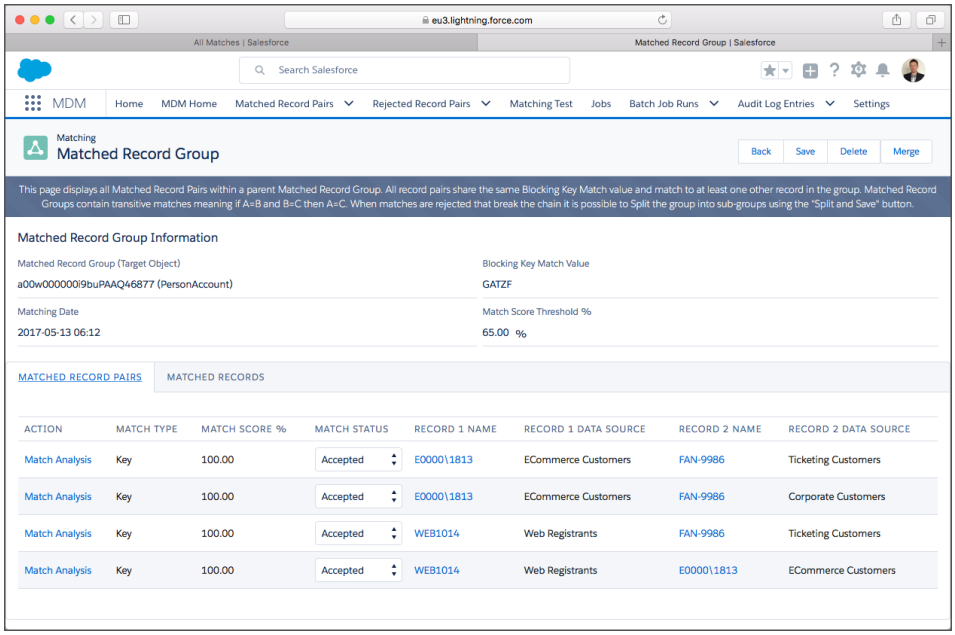
To complete this step, first navigate to the MDM App, open the Jobs tab, select the Matching Job type and the required Target Object. Finally select the required Job Schedule and click the “Schedule Job” button. This will display in the scheduled jobs tab until complete.

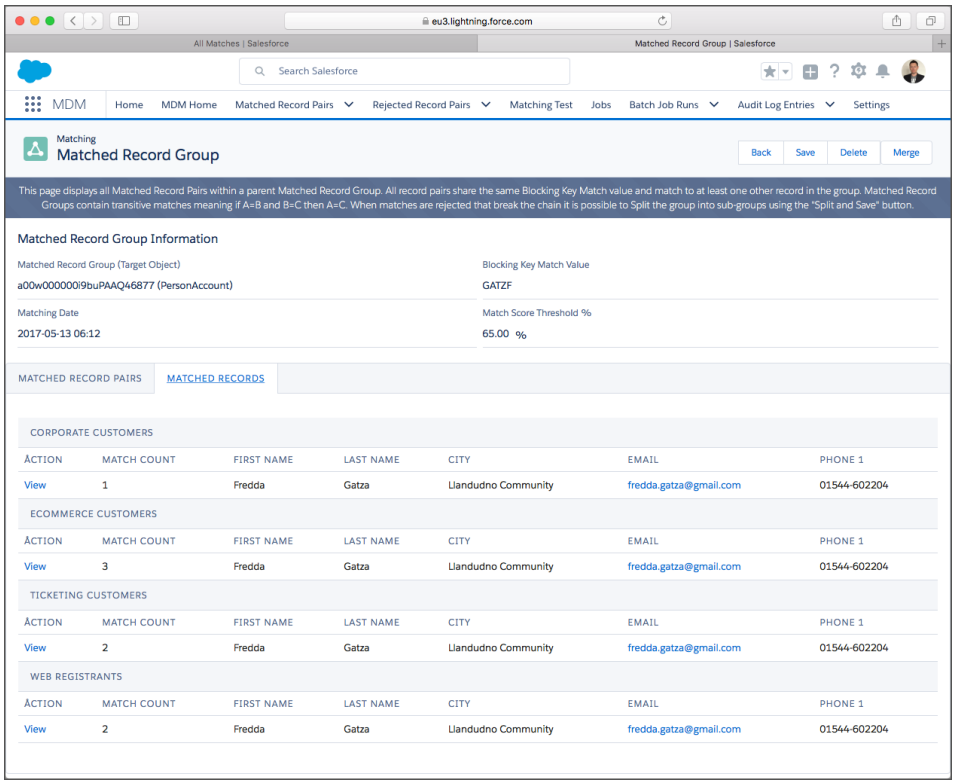


Step 6 – View Matching Results

To complete this step, first navigate to the MDM App, open the Matched Record Pairs tab and then open a Matched Record Pair record by clicking on the hyperlink in the Matched Record Pair Number column. On the Matched Record Pair detail page navigate to the Matched Record Group via the “View Matched Record Group” button (or dropdown menu option in Lightning Experience).

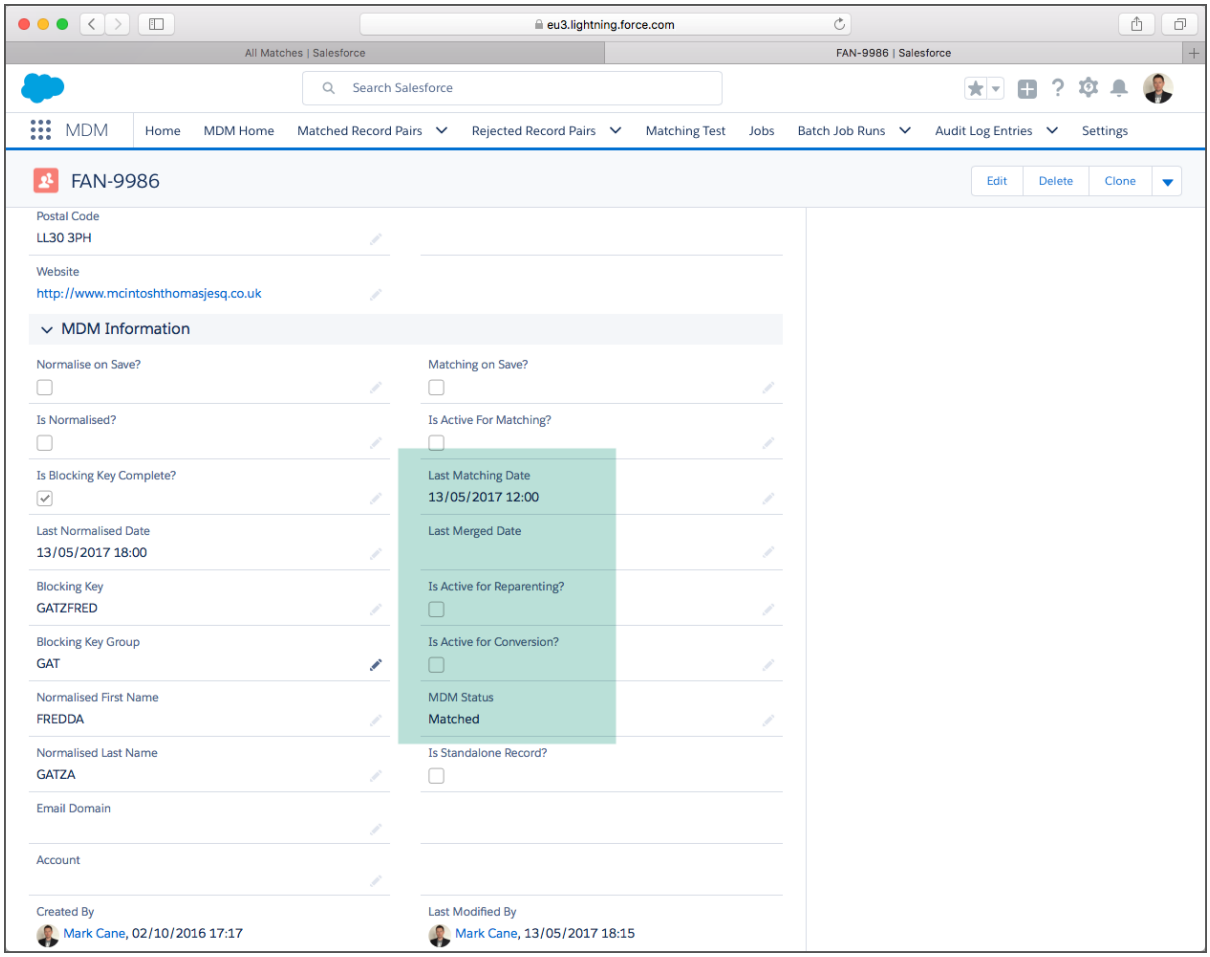
The screenshots below show a Matched Record Group created by the Matching MDM operation.





clearMDM® is a registered trademark of Audit9 Ltd a company registered in the United Kingdom with company number 08088394.

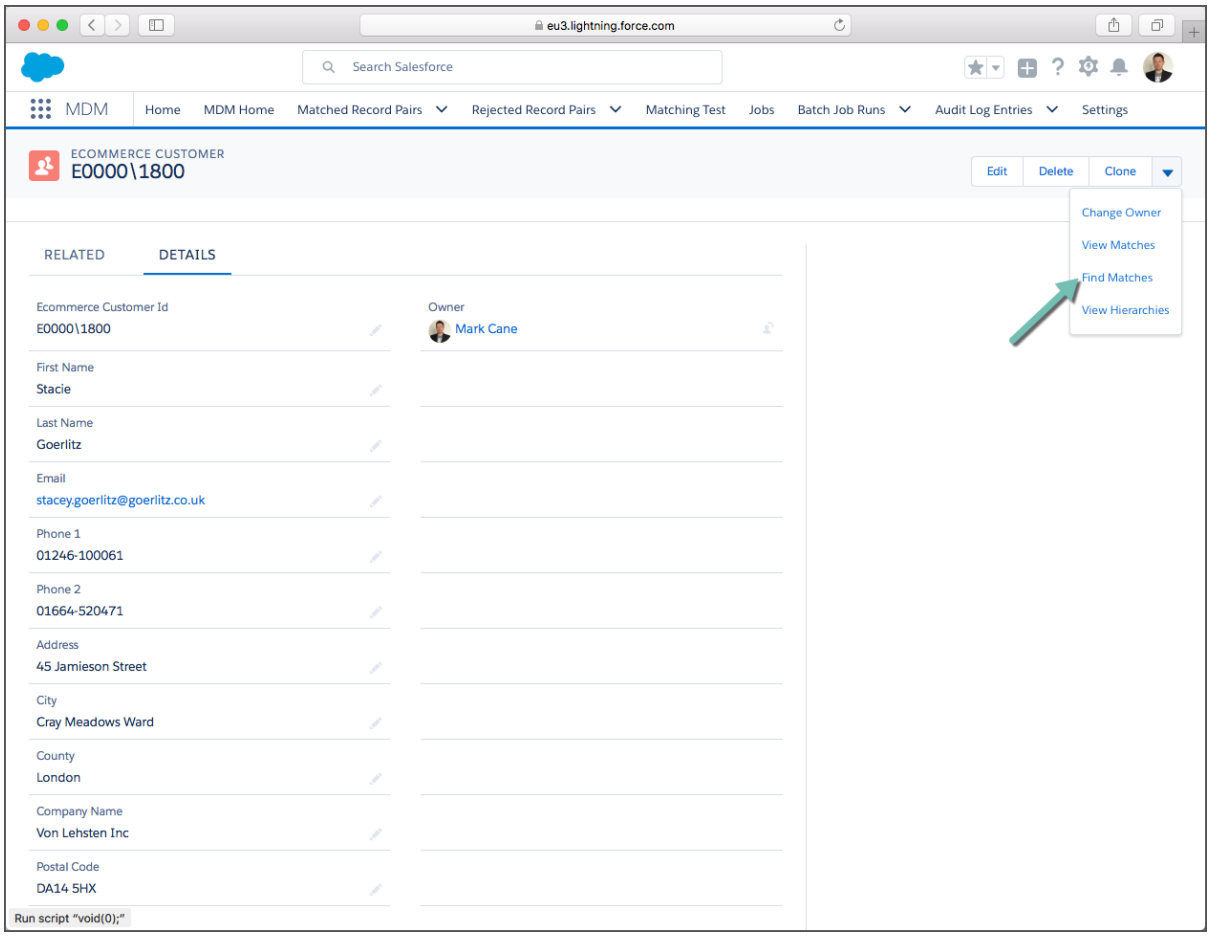
The screenshot below shows the Matching MDM fields displayed on an example record. Placement of MDM fields on layouts for administrators, data stewards and end-users is a key implementation decision.

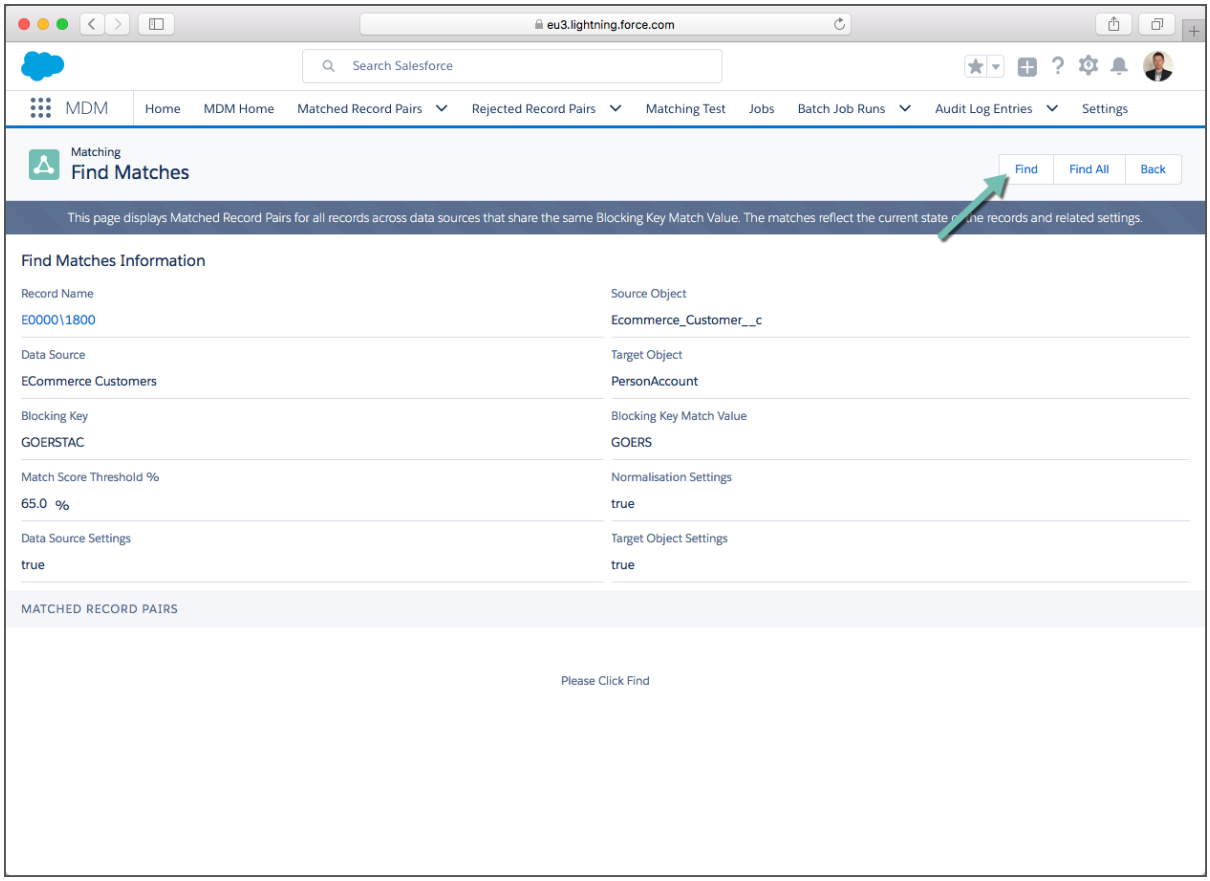


Step 7 – Manual Matching – Find Matches

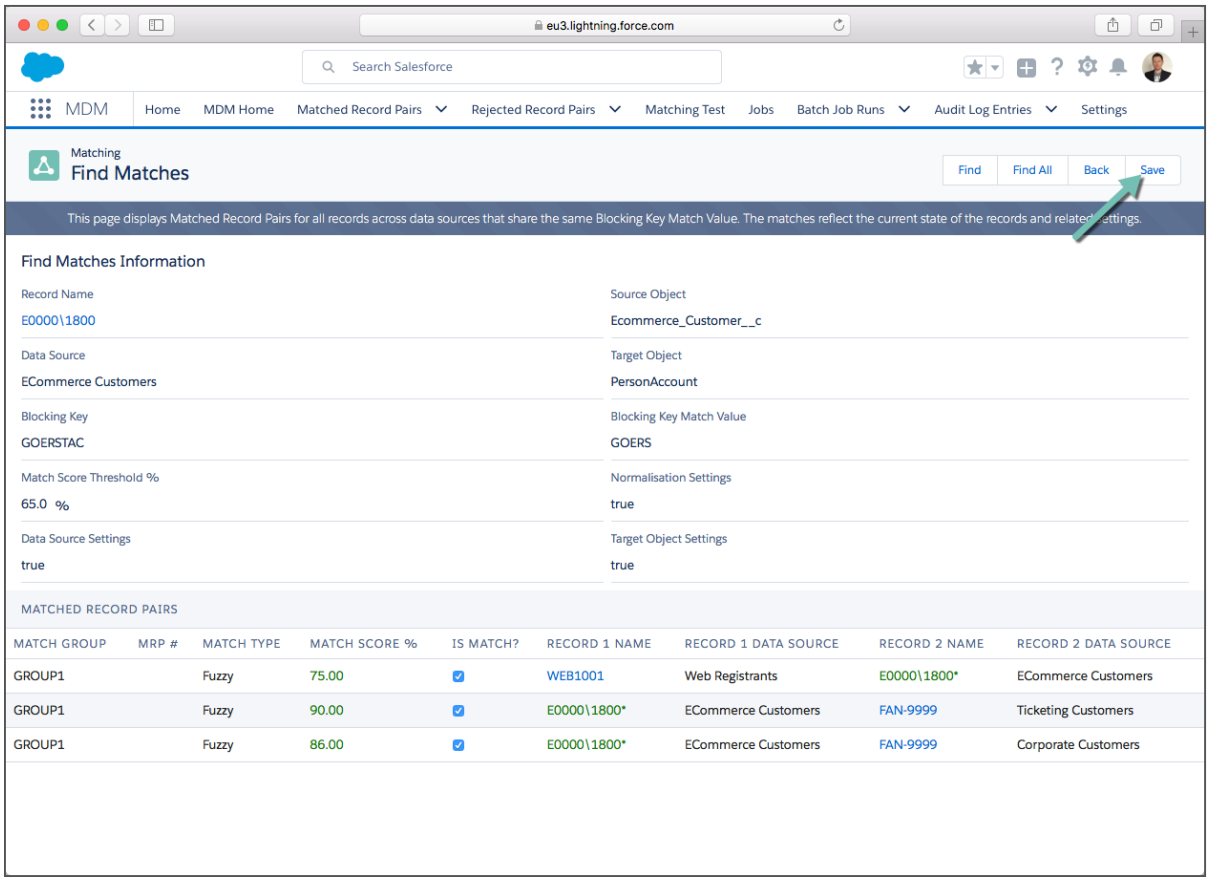
The screenshots below show the steps of the “Find Matches” feature that allows real time matching to be invoked from a button or link displayed on the record detail page.

Open a Source Record and click the “Find Matches” button or link. The “Find Matches” Custom Button for Account provides a template that can be copied for other objects. In Salesforce classic this is displayed as a button.



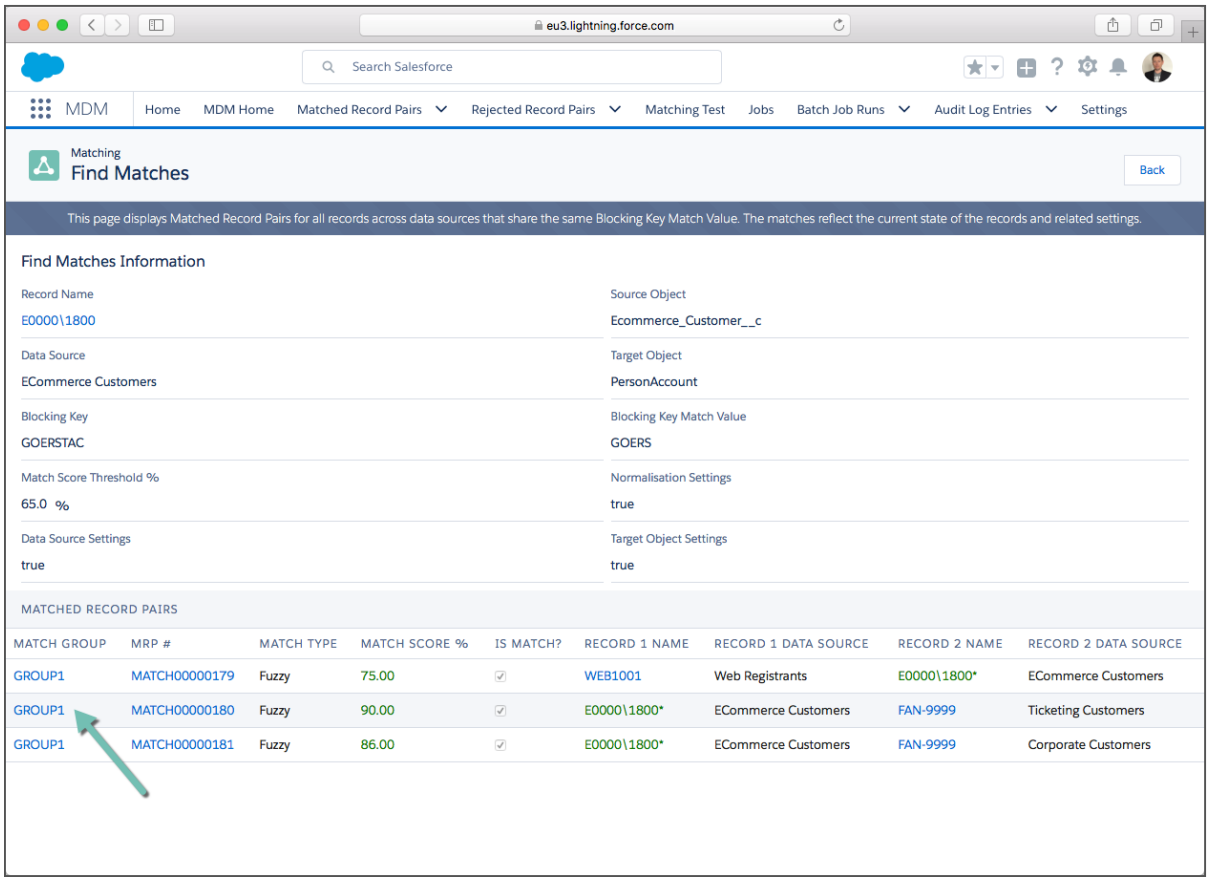


Next click the Find button (direct matches to the record) or Find Matches button (all matches for the blocking key match value) to perform a real-time matching operation.

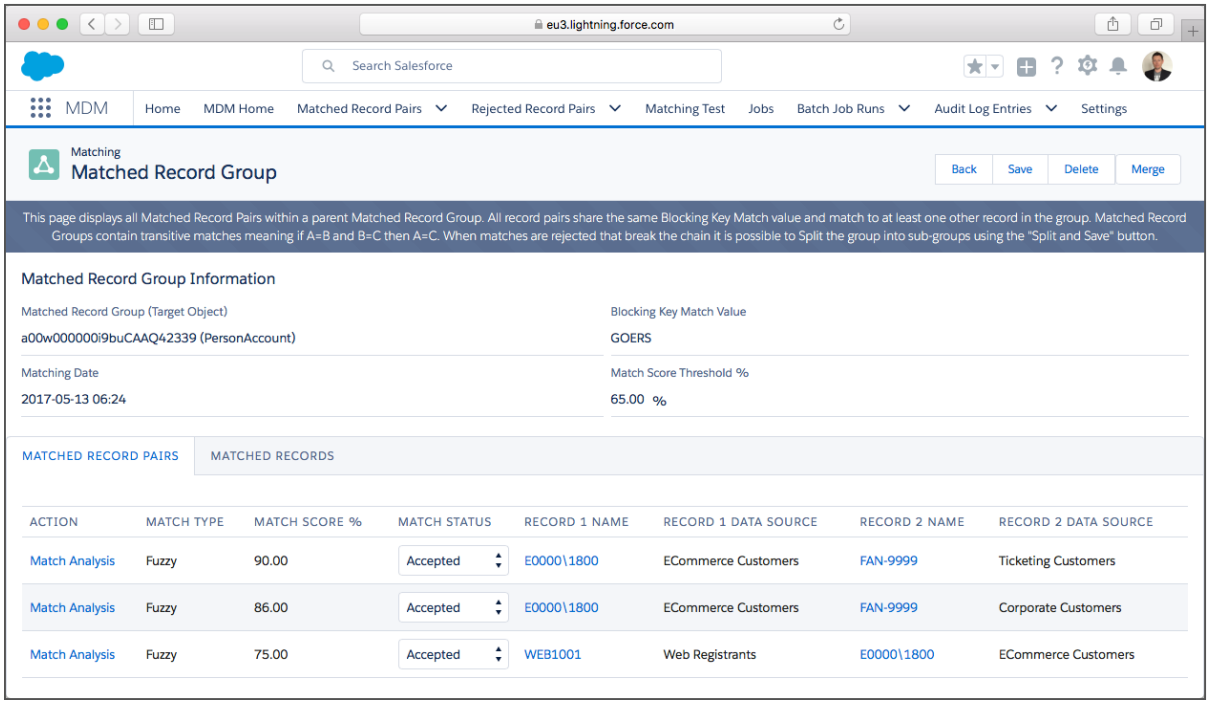


Next click the Save button to create a Matched Record Group for pairings that have the [Is Match?] column ticked. Note, this value can be overridden manually.

clearMDM® is a registered trademark of Audit9 Ltd a company registered in the United Kingdom with company number 08088394.



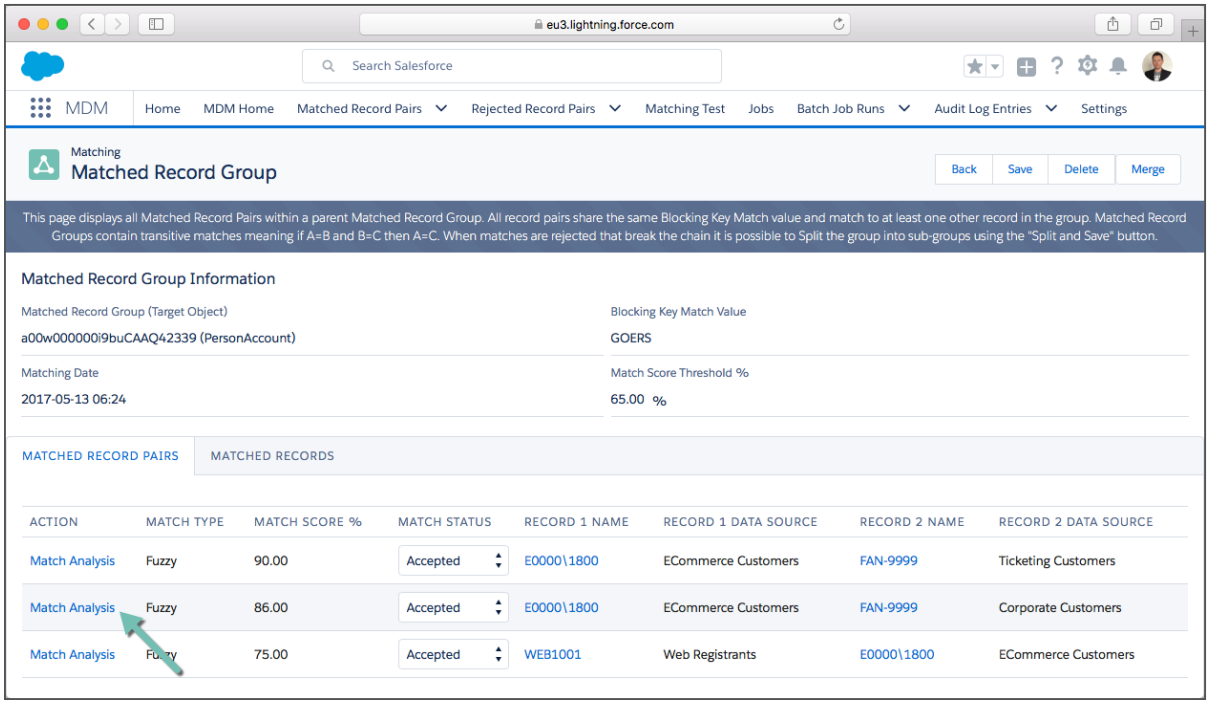
Next open the Matched Record Group via the hyperlink shown above.

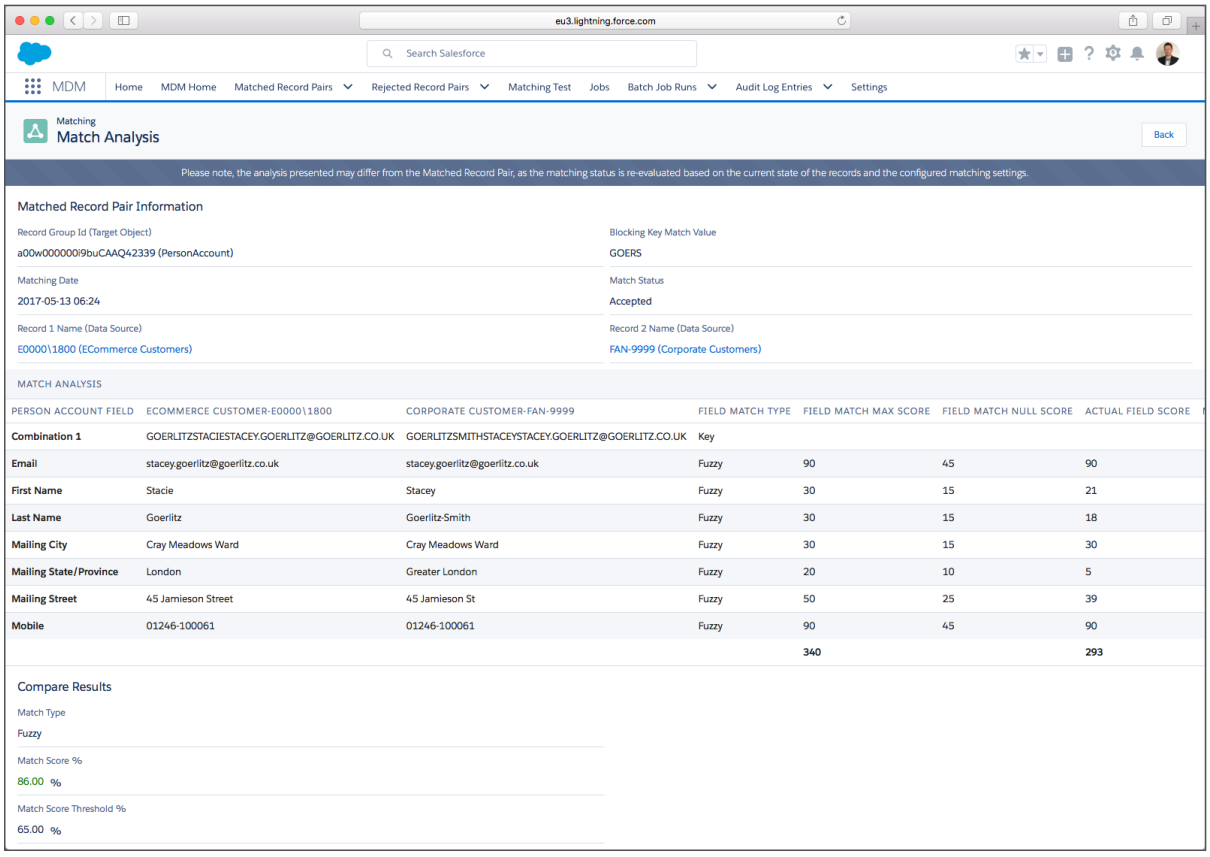


Finally the Matched Record Group enables Match Analysis of individual pairing and the option to manually merge the group.

Step 8 – Manual Matching – Match Analysis

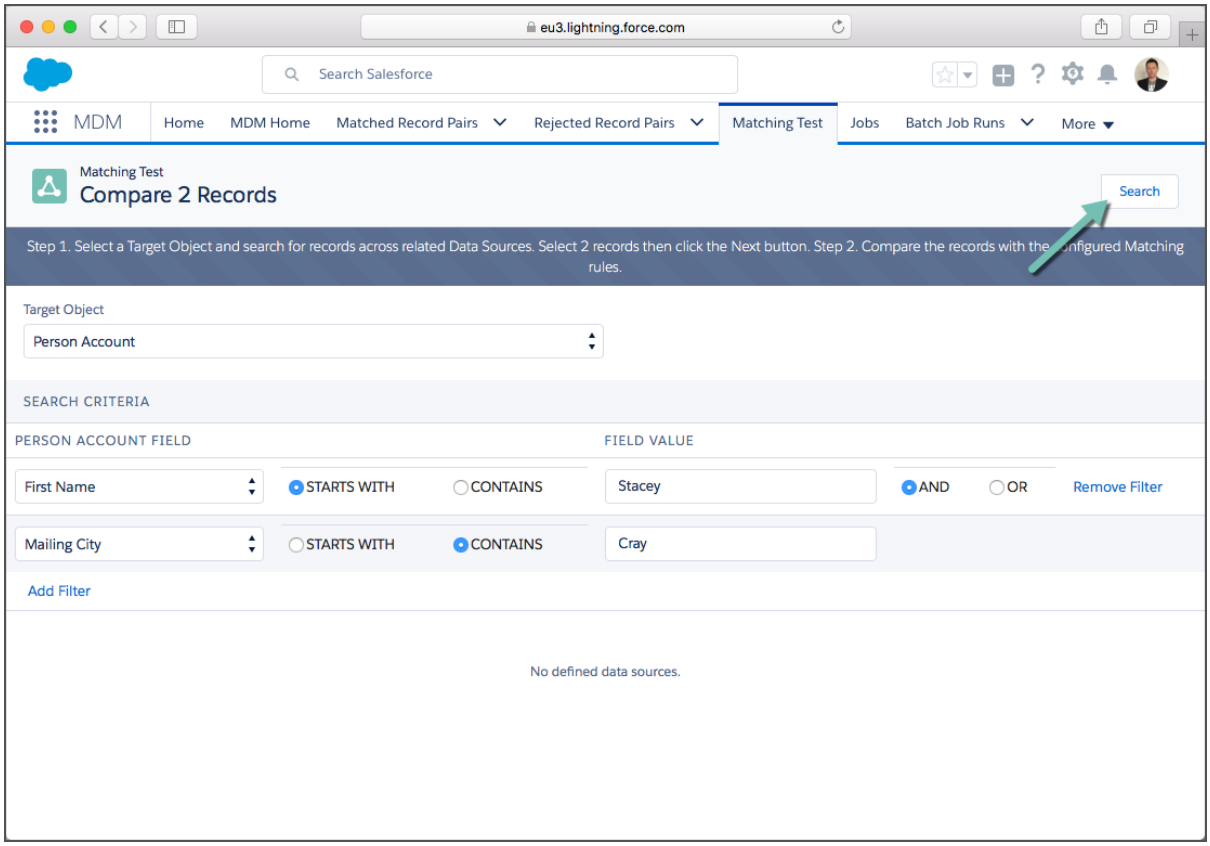
The screenshots below show the “Matched Record Group” and “Match Analysis” features. Note, the features are applicable to all Matched Record Groups irrespective of whether the grouping was created by manual, batch job or API matching.



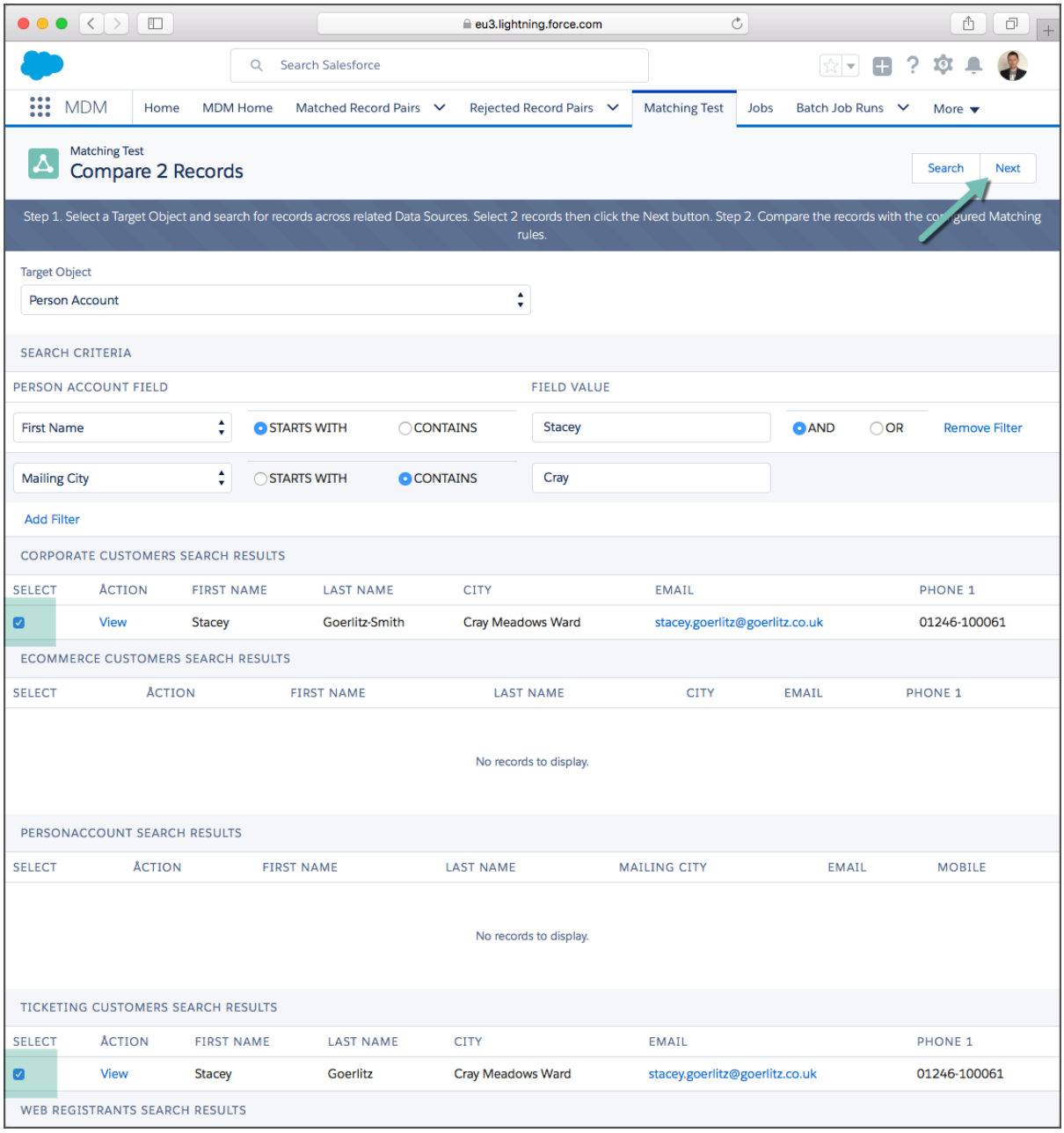


Step 9 – Manual Matching – Matching Test

The screenshots below show the “Matching Test” feature that allows two arbitrary records to found, compared and potentially merged. The feature is intended to enable an exploratory approach to the definition of the optimal set of Matching Rules.

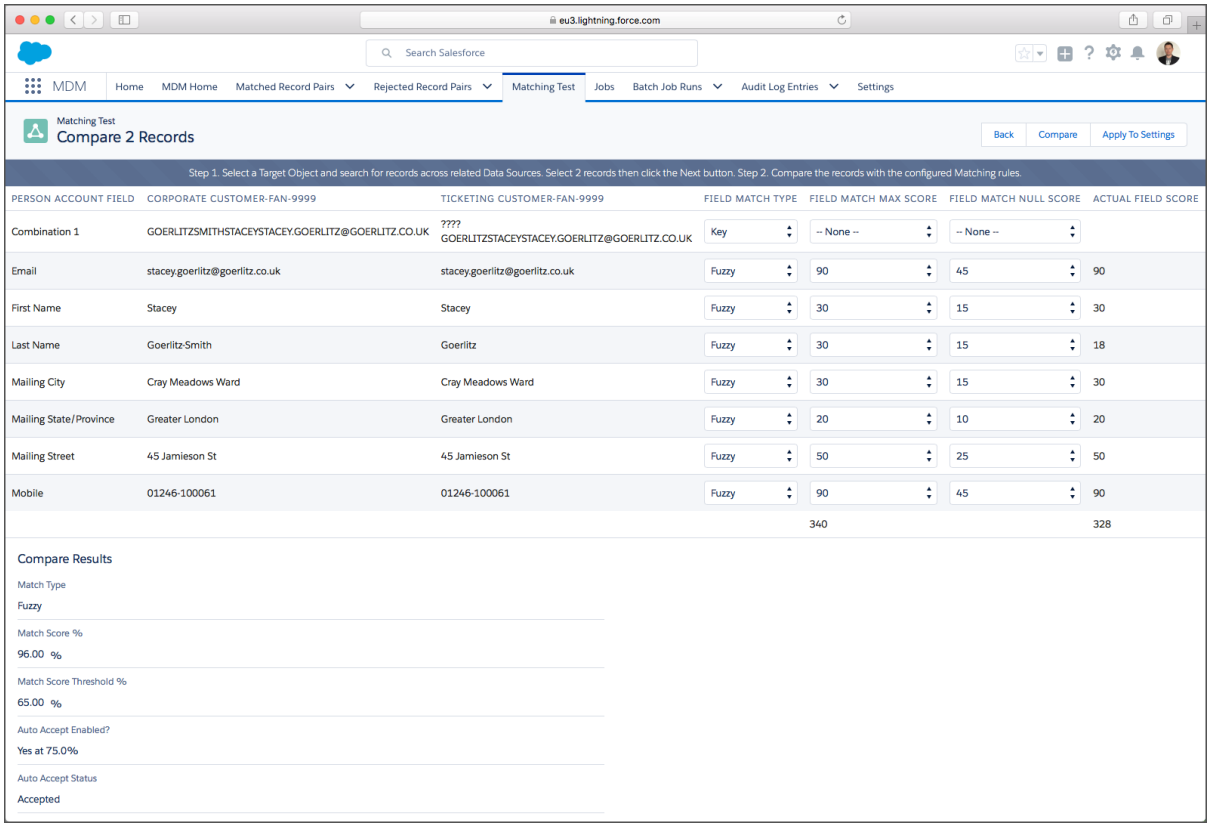


A compound filter can be defined using Starts With or Contains predicates. The Search buttons runs the defined query to return any matching records across all active Data Sources for the selected Target Object.



From the returned Search Results, 2 records can be selected and the Next button clicked to proceed to the Compare page.

clearMDM® is a registered trademark of Audit9 Ltd a company registered in the United Kingdom with company number 08088394.



The compare page allows Matching Rule changes to be applied and the Matching Score to be recalculated.

Appendix A - Matching Settings Reference

Target Object Matching Settings are configured 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.
Is Internal Matching Active?	If set to True the **Internal Matching** job is available for selection on the Jobs page.
Check for Matches on Record Creation?	If set to True then Matching will run for new records at the time of creation – where matches are found the record creation will be blocked.
Is Deterministic Rules First?	If set to True, deterministic rules are processed before key rules by the Matching engine.
Auto Adjust Blocking Key Match Value?	If set to True then Matching engine will increase the length of the **Blocking Key Match Value** to allow processing of groups that exceed the current settings. Auto-adjustment is only possible where the overall size of the Blocking Key (Input lengths 1,2 and 3 combined) exceeds the current Blocking Key Match Length.
Matching Check on Save Field Name	If set to True then Matching will run for modified records at the time of save – where matches are found the record modification will be blocked. This behaviour can prove too restrictive in certain circumstances (Admin profiles etc.) – as such the Matching Check can be bypassed using the Matching Check Override Field Name setting.
Matching Check Override Field Name	A checkbox formula field on the Target Object that returns true for a given record where the Matching Check on Save behaviour should be bypassed. The formula expression can encapsulate logical conditions (such as particular record types etc.) and access scenarios (User profile etc.).
Blocking Key Match Length	The length of the **Blocking Key Value** used for the initial grouping of records before **Matching Rules** are applied. It is best practice to define a Blocking Key length in the 8-10 range and use the first 6 or 7 characters for grouping. Where large groupings are encountered the Matching engine can add the 8th, 9th or 10th character to decompose the group into small subgroupings.
Date Matching Tolerance	The number of days difference between 2 compared Date or Datetime field values that constitutes a match. Where this value is Zero dates must be the same day, where the tolerance is 1 then the dates can be one day apart and still be considered a match.
Fuzzy Match Threshold %	The match score % that constitutes a Candidate match. Compared record pairs that score beneath this threshold are non-matches and are not recorded in the **Matched Record Pair** object.
Matching Manager Apex Class Name	The name of ApexClass to which the matching engine delegates the matching algorithm. The StandardMatchingManager default implements the Edit Distance or Levenshtein algorithm. Custom algorithms can be implemented using the Apex language.
Auto Accept Matches?	If set to True then compared record pairs may be set to Accepted status automatically if the score value is sufficiently high.

clearMDM® is a registered trademark of Audit9 Ltd a company registered in the United Kingdom with company number 08088394.

Auto Accept Match Threshold	The match score % that constitutes an Accepted match. Compared record pairs that score on or above this threshold are auto-accepted.
Is Invoked by Normalisation Job?	If set to True then Normalisation jobs will invoke a Matching job for the same Target Object upon completion.
Is Invoked by Synchronisation Job?	If set to True then Synchronisation jobs will invoke a Matching job for the same Target Object upon completion.

Data Source Settings are configured on the Data Source settings page.

Setting	Definition
Data Source Settings	
Is Active?	If set to False the Data Source will be excluded from all Matching MDM operations.
Is External?	If set to True then the Data Source is of the external type and a Source Object must be specified. If set to False then the Data Source is of the internal type and only the Target Object must be specified as the Source Object will be the same value.
Name	A unique name for the Data Source. Examples below. Accounts Master Accounts Excluded Contacts High Quality Leads SAP Companies Ecommerce Customers
Source SObject Type	**External Data Sources Only.** The name of the object this Data Source exposes records from.
Target SObject Type	The name of the Target Object to which this Data Source relates. The Matching MDM operation runs for a given Target Object; records are gathered from all Data Sources (internal and external) for the Target Object.
MDM Status Field Name	A text field on the Source Object into which the MDM status is set, e.g. Matched, No Match, Merge Source.
Is Active for Synchronisation?	If set to True then the Data Source will expose Source Records to the Synchronisation MDM operation. Synchronisation allows field value changes to efficiently flow across an existing linkage between a Source Record and a Master Record without the requirement to re-match and merge the modified Source Record. The Synchronisation MDM operation processes **Source Records** that have the [Is Active for Matching flag] set to true.
Is Active for Reparenting Field Name	A checkbox field on the Source Object that returns True where a Source Record has been merged and child records should be re-parented. The flag is set to True by the Merge operation and subsequently set to False by the Reparenting MDM operation. Where child records retain a relationship to the Source Record (Custom Objects) and are subject to ongoing change – the parent Source Record flag must be set to True to allows a relationship to the Master Record to be established. Process Builder provides an efficient means to address this requirement.

	The Reparenting MDM operation consolidates child records parented by merged Source Records to the related Master Record. This operation is key in providing a consolidated (360º) view.
Reset Reparenting Flag?	If set to True then the Matching MDM operation will set the [Is Active for Reparenting?] record level flag equal to false for all matched records in anticipation of the Merge MDM operation setting the same flag to true for merged records.
Is Partition?	If set to True then the Data Source exposes the subset of the Source Object records as defined by the partition filter.
Partition Field Name	**Is Partition = True only.** A custom text field on the Source Object upon which returns the partition filter value. **Best Practice:** A text formula field can be implemented to return the required partition value based on logic encapsulated in the expression. Alternatively the Data Quality MDM operation can populate the defined field with the partition value.
Partition Field Value	**Is Partition = True only.** A text value that defines the filter logic applied to the partition; external system name or region or quality grade etc.
Data Source Matching Settings	
Is Active?	If set to False the Data Source will be excluded from all Matching MDM operations.
Auto Match Records?	If set to True then all Source Records for the Source Object (subject to partition filtering) are exposed to the Matching MDM operation irrespective of the setting of the record-level [Is Active for Matching?] flag. Auto matching is typically used only when an existing dataset is matched for the first time – or during exploratory testing.
Master Records Active for Matching?	**Auto Match Records = False Only.** If set to True then all Master Records for the Source Object (subject to partition filtering) are exposed to the Matching MDM operation irrespective of the setting of the record-level [Is Active for Matching?] flag. This flag enables new or modified Source Records to be matched against the full set of existing Master Records for the same Blocking Key Match Value. If set to False then Source Records are matched against records that have the record-level [Is Active for Matching?] flag equal to true only.
Check Matching State on Save?	**Auto Match Records = False Only.** If set to True then record modifications will be evaluated to determine whether a matching significant field has changed, if this is the case then the [Is Active for Matching?] flag is set to true to expose the record to the next Synchronisation or Matching MDM operation.
Reset Source Record Matching Flag?	**Auto Match Records = False Only.** If set to True then the Matching MDM operation will set the record-level [Is Active for Matching?] flag equal to false once a record has been processed. This flag is key to ensuring that records are only processed once unless subsequent matching significant field changes are applied.
Is Active for Matching Field Name	**Auto Match Records = False Only.** A checkbox field on the Source Object that is populated with a true value when a record should be exposed to the Matching MDM operation.
Master Record Flag Field Name	**Auto Match Records = False &**

	Master Records Active for Matching = True Only. A checkbox field on the Source Object that is populated with a true value when a record is to considered a Master Record. If no field reference is set then the definition of a Master Record is taken from the MDM Status being equal to 'Merge Master' or 'Conversion Master'. **Best Practice:** In many cases standalone records that have no MDM Status are also considered to be a Master Record. To support this a formula field can be implemented that returns true when a record's MDM Status is not 'Merge Source' or 'Conversion Source'. The formula field will be set to control determination of Master Record state in this setting.
Last Matching Date Field Name	A datetime field on the Source Object that will be populated with the processing timestamp.
Source to Target Field Settings	
Target Field	A Target Object field that is active for Matching and Merge.
Source Field	**External Data Sources only.** A Source Object field that is mapped to the Target Field. MDM operations process data across object using a dynamic field map comprised of Source to Target field mappings.
Cross Field Matching	Additional fields on the Source Object where the Target Field value may exist. The Matching MDM operation will evaluate the best score for the Target Field across both the primary field and the secondary fields specified by this setting.

Appendix B – 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.

Matching Log Types

Rule Type	Definition
Group Size Limit Reached - Source Records Will Be Ignored [XXXXX] 7451:5000.0	The number of Source Records for the BKMV exceeds the setting value below. [System Settings] Max Records Per Iterable Cycle Where Data Sources exist that are configured with [Master Records Active for Matching?]=True then the actual limit applied to the Source Record count is reduced by the percentage specified in the setting below. [System Settings] Master Record Partition Allowance The above is necessary to limit the workload placed into each Batch Apex execute cycle such that CPU timeout errors are avoided. To mitigate this error, the BKMV can be made more selective by adding additional characters (increasing the BKMV length setting). Or, the System Settings below can be modified. **Max Records Per Group** : Controls the maximum BKMV group size before auto-adjustment takes place. *Recommended Setting: 500* **Max Records Per Iterable Cycle** : Controls the maximum number of records that can be placed into the workload for a single Batch Apex execute cycle. *Recommended Setting: 2000*

	Max Groups Per Iterable Cycle : Controls the maximum number of BKMV groups that can be placed into the workload for a single Batch Apex execute cycle. *Recommended Setting: 5 (this can be reduced down to 4,3,2 or 1 where limit issues are record as below)*
Blocking Key [XXXXX] Group Size Limit Reached - Source Records Will Be Ignored	The number of Source Records for the BKMV exceeds the setting value below – and auto adjustment is not enabled OR auto adjustment has been applied and the referenced BKMV is the full Blocking Key length and no further adjustment possible. [System Settings] Max Records Per Group
Blocking Key [XXXXX] Auto Adjustment - Source Records Will Be Ignored	The referenced BKMV has been auto-adjusted but not all Source Records could be assigned to sub groupings within the size limit defined by the setting below. [System Settings] Max Records Per Group
Cycle Processing Stopped - Processing Limit Exceeded (CpuTime, Heap or Dml Rows) bk:JOHNN cpu time:59708[59400.000] heap:689258[11880000.000] dml:isPersist=true:222[9999]	The workload placed into the Batch Apex execute cycle reached processing limits and to avoid a hard platform exception processing has been stopped. To mitigate this error please refer to the preceding row.
Missing or Inactive Target Object Matching settings	This error can occur where the Target Object Matching settings are inactive, deleted or the Salesforce User does not have permissions to the object or fields referenced.

clearMDM® is a registered trademark of Audit9 Ltd a company registered in the United Kingdom with company number 08088394.