

## QuickStart Guide 1 - Overview

Document Version: v1.2

Product Version: v2.9

Date: 9<sup>th</sup> September 2017

This document provides an overview for the key MDM Operations supported by clearMDM; document references are made to the QuickStart Guides provided for each MDM Operation.

The logical sequence required to orchestrate the individual MDM operations into an end-to-end MDM process is also provided, with an example implementation approach for reference purposes.

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

<b>QuickStart Guide 1 - Overview</b> .....	<b>1</b>
<b>Product Introduction</b> .....	<b>2</b>
What is clearMDM? .....	2
Key Features .....	2
Key Use Cases .....	2
<b>Key Concepts</b> .....	<b>3</b>
<b>MDM Operations</b> .....	<b>4</b>
<b>MDM Process Overview</b> .....	<b>9</b>
<b>Implementation Scenario – Daily Batch</b> .....	<b>10</b>
MDM Batch Processing Lifecycle .....	11

## Product Introduction

What is clearMDM?

clearMDM is the pragmatic solution to common Master Data Management challenges such as duplicated or fragmented customer data. Get a Single Customer View to improve operational efficiency alongside 360-degree insight critical to accurate segmentation and personalised customer relationships. The clearMDM solution offers lightweight, scalable MDM functions designed for the Salesforce platform that can be implemented quickly and supported by Salesforce administrators.

## Key Features

Feature	Customer Benefit
Single Customer View Automation	Operational Efficiency. End users view a single Master Record; duplicate records are hidden. Duplicate management is an administrative issue not a business concern.
360-degree Insight	Enhanced sales and service relationships, increased marketing effectiveness through accurate segmentation.
Data Profiling	Establish a clear understanding of data quality through Data Quality Rule Sets. Proactive and actionable analysis of quality metrics can be imperative to the success of data-centric projects.
Data Standardisation	Standardise data using configurable rules and custom reference data. Examples; Country name variations standardised to ISO Country Code, Surname suffixes removed, Special characters removed from field values.
Point-of-entry Protection	Protect data quality with point-of-entry protection that prevents duplication at source. A real-time API enables external systems to reference Salesforce as a MDM hub.

## Key Use Cases

Use Case	Description
External Data Consolidation	Manage the ongoing flow of data onto the Salesforce platform from external sources (ERP, Website, E-commerce systems etc.) to extend or augment the Master Data-set.
Internal Data Management	De-duplicate and consolidate existing Salesforce records with point-of-entry protection.
MDM Hub	Extend MDM processing to external systems via REST API operations.
Data Quality	Clear visibility of data quality challenges can be key to a successful Salesforce implementation. clearMDM provides flexible data profiling tools that deliver data quality metrics and actionable insights.

## Key Concepts

**Target Object.** A compatible object that is defined as the target for Normalisation operations and where Master Records will be created or updated.

**Data Source.** A compatible object that provides data to MDM operations. Partition data sources enable a single object to support multiple data sources each with distinct configurations. 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) to allow different behaviour to be applied.

A broad range of **Standard Objects** (including Person Accounts) are supported as both Target Objects and Data Sources. **Custom Objects** are also supported.

**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 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.

**Source Record.** 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 completely hidden from view for Salesforce end-users or deleted entirely once processed.

**Master Record Update Logic - Attribute Group.** Attribute Groups are collections of fields on a Target Object that must be merged together from a single Source Record only. In determining how the Master Record fields (in the Attribute Group) are populated each related Source Record is evaluated for validity (typically completeness) and priority. Attribute Group priority can be defined on a Newest, Oldest or Dynamic Priority basis allowing custom business rules to direct which Source Record the field values are taken from. Attribute Group processing performs a full re-evaluation of all Source Records for a given Master Record whenever a new or existing Source Record modification is identified. Merge and Synchronisation MDM operations perform Attribute Group processing.

**Master Record Update Logic – Field Priority.** Fields that are not mapped to an Attribute Group are processed using Data Source specific merge priorities. Master Records retain a memory of the Source Record (and Data Source) that last updated each field. Subsequent Merge and Synchronisation operations will only update fields from a Source Record with a higher merge priority for the field, or where the record is more recently updated should the priority be equal. Manual updates, where Master Record fields are directly updated via the Salesforce UI or APIs, are assigned a merge priority that can be used in the relative scale evaluation applied to each field. This allows manual updates to be retained irrespective of the occurrence of Source Record updates on a per-field selective basis.

## MDM Operations

The table below outlines the complete set of MDM operations supported by clearMDM in the logical order of application.

#	MDM Operation	Description	Inputs	Outputs
1	<p><b>Normalisation</b></p> <p><i>QuickStart Guide 2 - Normalisation</i></p>	<p>The Normalisation MDM operation has 2 functions;</p> <p><b>1) Field Standardisation.</b> Matching accuracy is improved through the pre-standardisation of key input fields. Examples being country name standardisation to the ISO Country Code, or removal of spurious special characters or irregular suffixes.</p> <p>Normalisation Rules take an input field, apply a rule and populate an output field with the product. The output fields are then used in the composition of the Blocking Key and within Matching Rules.</p> <p><b>2) Blocking Key Population.</b> The first-pass 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. Each unique key should cover at most 250 Source Records. The <b>Blocking Key Statistics</b> Job can be used to evaluate the distribution of records based on the current Blocking Key structure - statistics data is output to the Blocking Key Statistics object which is visible via tab or reporting.</p> <p>Note, the Blocking Key Match Value setting determines the actual length of the Blocking Key that is initially used in Matching. For example, the full Blocking Key may be 9 characters, with the first 6 used for matching, when larger record groupings are encountered the system may adjust the length to 7, 8 or 9 to enable processing to proceed within Salesforce limits.</p>	NA	<p><b>Standardised Fields</b> for use by Matching Rules.</p> <p><b>Blocking Key</b> used to group record for Matching.</p>

		<p>Note, Records without a complete Blocking Key are ignored by the MDM Matching operation.</p> <p>The relevant MDM field is: <b>[Is Blocking Key Complete?]</b></p> <p>A typical Blocking Key for a Business Account is constructed using the;</p> <p>[Country ISO Code] and [Normalised Name] fields.</p> <p>e.g. GBFABRIKA</p> <p>For Matching purposes, the grouping may be GBFABR to allow for spelling deviations in the Name.</p>		
2	<b>Data Quality</b>	<p>The Data Quality MDM operation enables records to be evaluated by quality rules that can calculate a quality score and apply quality related actions (corrective, preventative etc.).</p> <p>Quality Rule Sets can be used to intelligently and automatically set the correct Data Source partition on a per-record basis, to set Dynamic Priority values for Attribute Groups and to calculate the Master Record Priority which governs the selection of the Master Record within a Matching Record Group.</p> <p>Quality Rule Sets can also apply corrective actions to update field values based on conditional (and time-based) logic.</p>	<b>Standardised Fields</b> may be subjected to Corrective Actions.	<b>Standardised Fields</b> for use by Matching Rules.
3	<b>Synchronisation</b>	<p>The Synchronisation MDM operation provides an efficient method of promoting Source Record changes to the Master Record across a pre-existing relationship. This operation negates the need to Match and Merge records that have previously been matched.</p>		

4	<p><b>Matching</b></p> <p><i>QuickStart Guide 3 - Matching</i></p>	<p>The Matching MDM operation applies a 2-pass approach;</p> <p><b>1) Blocking Key.</b> Records are first grouped by the Blocking Key Match Value (first x characters of the Blocking Key as defined by the Blocking Key Match Value setting). Records that have an incomplete Blocking Key, or the length of the key is less than the Blocking Key Match Value are skipped.</p> <p><b>2) Matching Rules.</b> For each Blocking Key Match Value grouping (e.g. records that share the grouping value GBFABR) defined Matching Rules (Key, Deterministic and Fuzzy) are evaluated against each unique pairing of records in the group to identify whether the records match or not. In cases where matches are identified, transitive <b>Matching Record Groups</b> are constructed where each record matches to one or more records in the group. Records do not have to match to every other record.</p> <p>Matched Record Groups are recorded in the <b>Matched Record Pair</b> object, the [Record Grouping Id] field groups together matched record pairs within a Matched Record Group.</p>	<p><b>Standardised Fields</b> for use by Matching Rules.</p> <p><b>Blocking Key</b> used to group record for Matching.</p>	<p><b>Matched Record Groups</b> to be merged.</p>
5	<p><b>Merge</b></p> <p><i>QuickStart Guide 4 - Merge</i></p>	<p>The Merge MDM operation is tasked with processing each Matched Record Group into either a Master Record Create or Update operation. Source Records are never deleted.</p> <p>Each <b>Matched Record Pair</b> record has an individual Match Status field value. Depending on the level of the calculated match score this status may automatically be set to 'Accepted' at lower scores the default value may be 'Candidate' which would require a Data Steward to manually review the</p>	<p><b>Matched Record Groups</b> to be merged.</p>	<p><b>Master Records</b></p>

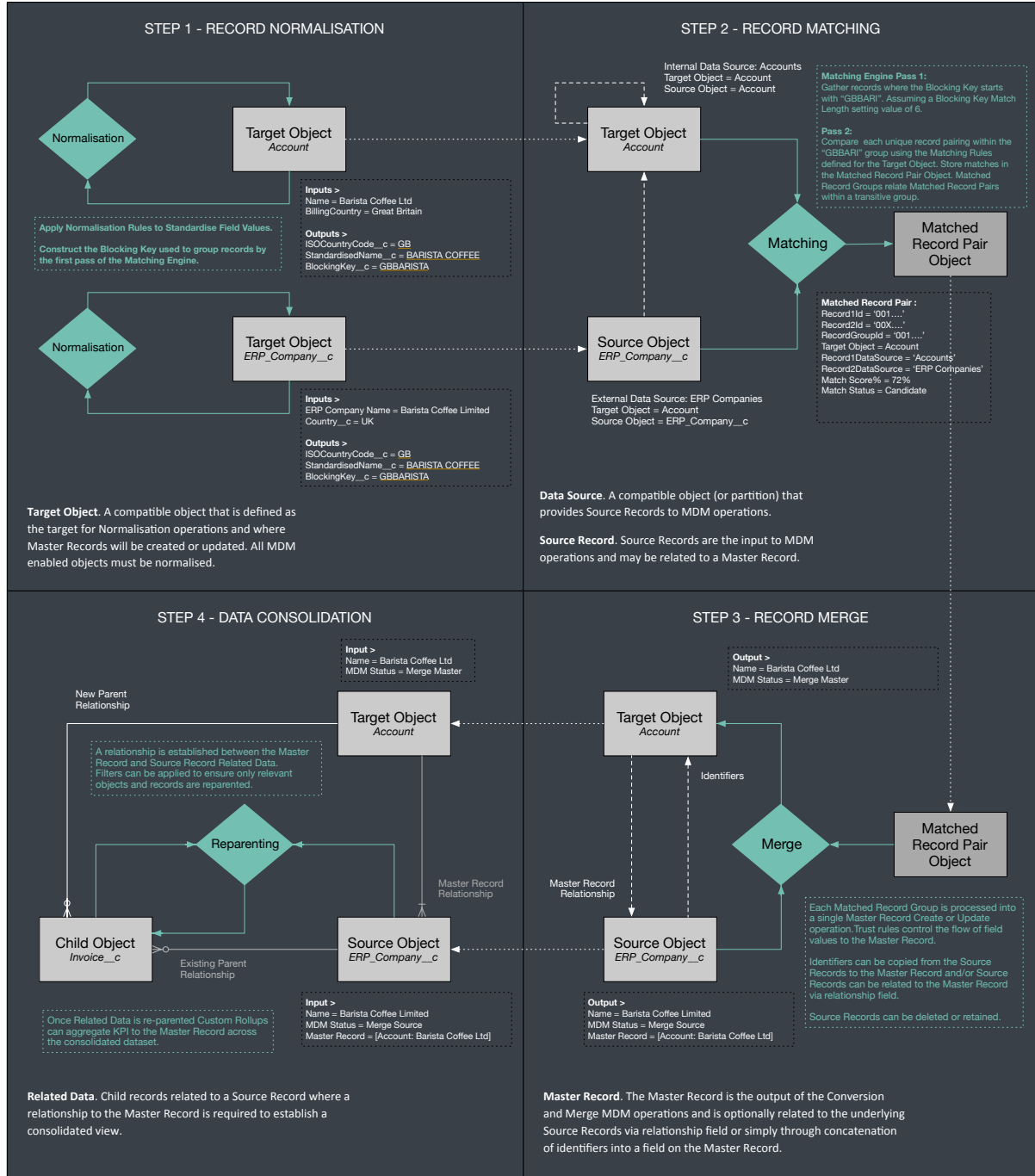
		<p>pairing and set the status to 'Accepted' or 'Rejected'. The score levels that apply to auto-acceptance are configurable enabling the level of manual intervention to be controlled.</p> <p>Matched Record Groups that contain one or more 'Candidate' status pairing are considered to be partial and will be skipped by the Merge operation. It is therefore imperative that all pairings are accepted or rejected in advance.</p> <p>Matched Record Pairs that are rejected are recorded in the <b>Rejected Record Pair</b> object which is referenced by the Matching operation to avoid repeated matching of previously rejected matches.</p> <p>The Merge operation deletes the <b>Matched Record Pair</b> records for processed <b>Matched Record Groups</b>.</p>		
6	<p><b>Conversion</b></p>	<p>The Conversion MDM operation provides a means to create Master Records from standalone (or unmatched) Source Records on a one-to-one basis.</p> <p>Standalone records can occur where the Blocking Key Match Value is unique, or where the record simply doesn't match any records that share the same Blocking Key Match Value.</p>		
7	<p><b>Re-parenting</b></p> <p><i>QuickStart Guide 5 – Data Consolidation</i></p>	<p>The Re-parenting MDM operation enables Child Records (e.g. Tasks, Opportunities, Cases) to be re-parenting from the Source Record to the Master Record as a key part of delivering a consolidated view.</p> <p>For Standard Object to Child Object relationships (e.g. Account to Case), the existing relationship field is directly updated. It is best practice to use Field History</p>	<p><b>Re-parenting Rules</b></p>	

		<p>Tracking to track the Owner field in this case.</p> <p>For Custom Object to Child Object relationships (e.g. EcomCust__c to Case), a distinct Lookup type relationship field must be selected that references the Master Record object. In this model the original parent relationship (to the Custom Object) is retained; this allows the child records to follow the Source Record, should a new Master Record relationship be established at a later stage.</p> <p>The specific relationships which are re-parented are configurable.</p> <p>Selective filters can be applied to ensure that only relevant data is re-parented to the Master Record. For example, time-expired Opportunities or Cases may create data noise that is undesirable.</p> <p>Note, Master Detail relationships must be configured to allow re-parenting.</p>		
8	<p><b>Custom Rollup</b></p> <p><i>QuickStart Guide 5 – Data Consolidation</i></p>	<p>The Custom Rollup MDM operation enables aggregate calculations to be applied to Child Records related to Master Records.</p> <p>This function is comparable to the standard Rollup Summary Field capability which is limited to Master Detail relationships. The Custom Rollup operation supports lookup relationships and also Text concatenation.</p>	<p><b>Custom Rollup Rules</b></p>	



## MDM Process Overview

The diagram below provides an overview of the core MDM operations in sequence.



## Implementation Scenario – Daily Batch

clearMDM can be implemented to support MDM processing for multiple records on a scheduled batch basis or per-record on a transactional basis. This section provides an example implementation scenario where a batch MDM processing model is applied.

In this example the clearMDM implementation is tasked with consolidation of customer records from an external ecommerce system (Affinity) with an existing and evolving customer dataset managed within Salesforce (as Person Accounts). Affinity customer records are integrated to a Custom Object (Affinity\_Customer\_\_c) via an ETL tool on a daily basis. clearMDM processes new or changed Affinity records each day applying standardisation, matching and merge actions. Note, the scenario below covers the day-to-day operationalised processing cycle, additional considerations are required for the initial bulk processing of an existing dataset.

The following outcomes are key to understand; in all cases records are standardised by the **Normalisation MDM operation** as the first step.

1. New Source Record – match.  
New Source Records are matched by the **Matching MDM operation** against Master Customer Records. Where matching is successful the Affinity Customer is linked to the Master Customer (Person Account) by the **Merge MDM operation** and the MDM Status is set to “Merge Source”. Merge Trust Rules allow field data to update the Master Record on a controlled basis. Following MDM processing the new record is deactivated for subsequent MDM processing; activation will occur when the record is next updated from the original source via data integration.
2. New Source Record – no match.  
Where a new Source Record (i.e. Affinity Customer) is not matched, the record can be converted to a Master Customer Record on a one-to-one basis by the **Conversion MDM operation** or left unprocessed to allow future matching attempts.
3. Existing Source Record – previous match.  
Where a modified Source Record (i.e. Affinity Customer) has an existing link to a Master Customer Record (i.e. Person Account) this relationship can be retained and used to promote field changes (only) via the **Synchronisation MDM operation**, no attempt is made to rematch the record.
4. Existing Source Record – no previous match.  
If the modified Source Record (i.e. Affinity Customer) was previously converted to a Master Customer Record then this relationship can be retained and used to promote field changes only via the **Synchronisation MDM operation**, no attempt is made to rematch the record. Otherwise the record can be processed by the **Matching MDM operation** as per a new Source Record.

## MDM Batch Processing Lifecycle

The table below outlines the data processing lifecycle for this example scenario.

**Key Concept:** Record-level flags are used to control the selective activation of Source Records for MDM processing; this approach avoids re-processing the complete dataset where only a subset of the records has materially changed.

Time	Operation	Logic	MDM Flags	Invocation
06:00	Data Integration	The ETL tool runs data synchronisation tasks to transit new and modified records from Affinity to the Salesforce Affinity_Customer__c custom object.	NA	Scheduled
06:00	Data Integration MDM Pre-processing	New and modified Affinity_Customer__c records are Normalised on create/update - as such a batch job is not required.  The data integration task sets the record-level flags that activate the records for MDM processing,	[Is Active for Matching?] = <b>True</b>  [Is Active for Conversion?] = <b>True</b> (new records only)	NA
6:00	Data Integration	A Post-Processing step invokes the MDM Normalisation operation via the REST API.	NA	NA
6:05	Normalisation	The Normalisation job runs for <b>Person Account</b> .  This ensures that any new/modified Person Accounts are normalised before matching.	[Is Normalised?]= <b>True</b>	BatchJobRunAction API
06:19	Synchronisation	Source Records with an existing link to a Master Record are processed.	[Is Active for Matching?] = <b>False</b>	Job Chaining.
07:00	Matching	New Source Records or those without an existing link to a Master Record are processed.	[Is Active for Matching?] = <b>False</b>	Job Chaining.
07:00	Merge	Matched Record Groups are processing into Master Record Create and/or Update operations.	[Is Active for Reparenting?] = <b>True</b>	Job Chaining.
07:00	Conversion	New Source Records with no match are processed to a new Master Record (1-2-1 basis).	[Is Active for Conversion?] = <b>False</b>  [Is Active for Reparenting?] = <b>True</b>	Job Chaining.
07:00	Re-parenting	Child Records are selectively re-parented from Source Record to the linked Master Record.	[Is Active for Reparenting?] = <b>False</b>	Job Chaining.