Informatica

Informatica® Cloud MDM - Customer 360 for Salesforce

Winter 2020 Version 7.00

Administrator Guide

Informatica Cloud MDM - Customer 360 for Salesforce Administrator Guide
Winter 2020 Version 7.00
December 2019

© Copyright Informatica LLC 1998, 2019

This software and documentation are provided only under a separate license agreement containing restrictions on use and disclosure. No part of this document may be reproduced or transmitted in any form, by any means (electronic, photocopying, recording or otherwise) without prior consent of Informatica LLC.

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation is subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License.

Informatica, Informatica Intelligent Cloud Services, and the Informatica logo are trademarks or registered trademarks of Informatica LLC in the United States and many jurisdictions throughout the world. A current list of Informatica trademarks is available on the web at https://www.informatica.com/trademarks.html. Other company and product names may be trade names or trademarks of their respective owners.

The information in this documentation is subject to change without notice. If you find any problems in this documentation, report them to us at infa_documentation@informatica.com.

Informatica products are warranted according to the terms and conditions of the agreements under which they are provided. INFORMATICA PROVIDES THE INFORMATION IN THIS DOCUMENT "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING WITHOUT ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND ANY WARRANTY OR CONDITION OF NON-INFRINGEMENT.

Publication Date: 2019-12-12

Table of Contents

Preface 9

Informatica Resources. 9

Informatica Network. 9

Informatica Knowledge Base. 9

Informatica Documentation. 9

Informatica Product Availability Matrices. 10

Informatica Velocity. 10

Informatica Marketplace. 10

Informatica Global Customer Support. 10

Chapter 1: Introduction to Informatica Cloud MDM - Customer 360 for Salesforce. 11

Cloud MDM - Customer 360 for Salesforce Overview. 11

CC360 Architecture. 12

Cloud MDM - Customer 360 for Salesforce Processes. 13

Beans, Master Beans, and External Beans. 14

Life Cycle of Beans and Master Beans 14

Cloud MDM Tabs. 15

Chapter 2: Configuring the Data Cleansing Settings. 17

Configure the Data Cleansing Settings Overview. 17

Data Cleansing Options. 17

Legal Form Normalization. 18

Country Name Normalization. 18

Domain Normalization. 19

Enabling Data Cleansing Options. 19

Configuring Settings for a Picklist-Enabled Organization. 19

Configuring Legal Forms. 20

Configuring International Settings. 21

Chapter 3: Managing Batch Jobs. 22

Managing Batch Jobs Overview. 22

CC360 Admin Tab. 23

Batch Jobs. 23

Configuring a Custom Filter for a Batch Job. 24

Setting the Priorities for a Batch Job. 25

Synchronously Run Matching and Duplicate Contact Check Batch Jobs. 26

Running a Batch Job. 26

Pausing and Resuming Job Schedules. 27

Rescheduling Failed Jobs. 27

Managing the Batch Queue. 27
Monitoring the Running Batch Jobs. 28
Monitoring Queue Entries. 29
Batch Job Statuses. 30
Removing the Queue Entries from the Batch Queue. 31
Integrating External Batch Job. 31
Viewing Batch Statistics. 33

Chapter 4: Verifying Addresses, Email Addresses, and Phone Numbers. 35

Verifying Addresses, Email Addresses, and Phone Numbers Overview. 35
Address, Email Address, and Phone Number Verification Process. 36
Status of Records in the EventQueue Table. 36
Template for the DaaS Batch Job. 37
Settings for the Contact Data Verification Process. 38
Step 1. Create a Template for the DaaS Job. 38
Step 2. Configure the Address Verification Service Settings. 39
Step 3. Configure the Email Address Verification Service Settings. 41
Step 4. Configure the Phone Number Validation Service Settings. 41
Step 5. Configuring the DaaS Mapping. 42
Step 6. Configure the Account, Contact, and Lead Page Layout for Salesforce Classic. 43
Step 7. Configuring the Salesforce Web Service Timeout Interval (Optional). 43
Step 8. Running the DaaS Job 44

Chapter 5: Synchronizing Salesforce Records with CC360. 45

Synchronizing Salesforce Records with CC360 Overview. 45
Synchronization Process. 45
Settings for the Synchronization Process. 46
Configuring the Synchronization Settings. 46
Configuring the Performance Settings. 49
Migrate Batch Jobs. 50
Rerunning the Migrate Batch Jobs. 51
Running the HierarchyMigration Job in the Developer Console. 51

Chapter 6: Managing Duplicate Records. 52

Managing Duplicate Records Overview. 52
Duplicate Record Management Process. 52
Configuring Settings to Identify Duplicate Records in Real Time. 53
Setting Permissions for a User Profile to Identify Duplicate Records in Real Time. 53
Settings for the Duplicate Record Management Process. 54
Configuring the Batch Acceleration Level. 54
Configuring the Matching Settings. 55
Configuring the Match Score Weightings 58
Configuring the Null Scores. 59

Configuring the Matching and Duplicate Thresholds. 59
Comparing Records in the Fuzzy Matching Page. 60
Comparing Records in Salesforce Classic. 60
Comparing Records in Lightning Experience. 60
Duplicate Batch Jobs. 61
Rerunning the Duplicate Check Batch Job. 61
Disabling Duplicate Checking Between Person Accounts and Business Accounts. 62
Enabling Duplicate Checking Among Person Accounts, Leads, and Contacts. 63
Merge Batch Job. 64
Merging Contact Records. 64
Merging Person Accounts. 64
Identifying Duplicate Records for Custom Objects. 65
Step 1. Create an Object Class for the Custom Object. 65
Step 2. Create a Synchronization Setting for the Custom Object. 66
Step 3. Create a Data Source Setting for the Custom Object. 67
Step 4. Create a Matching Setting for the Custom Object. 67
Step 5. Create a Segment Setting for the Custom Object. 68
Step 6. Create an Apex Trigger for the Custom Object. 69
Step 7. Configure Field Sets for the Custom Object 70

Chapter 7: Consolidating Records. 71

Consolidating Records Overview. 71
Consolidation Process. 71
Settings for the Consolidation Process. 72
Configuring the Data Source Settings. 73
Configuring the Enrichment Settings. 74
Enrichment Settings for a Record Type. 77
Matching Batch Job. 77
Consolidation Batch Job. 78
Converting a Master Bean to a Salesforce Object. 78

Chapter 8: Converting External Records to Salesforce Records. 79

Converting External Records to Salesforce Records Overview. 79
Data Conversion Process. 80
Settings for the Data Conversion Process. 80
Configuring the Synchronization Settings. 81
Configuring the Matching Settings. 83
Comparing Records in the Fuzzy Matching Page. 87
Comparing Records in Salesforce Classic. 87
Comparing Records in Lightning Experience. 87
Data Conversion Batch Job. 88
Data Conversion after Consolidation Process. 88

Chapter 9: Managing Hierarchies. 89

Hierarchies Overview. 89
Adding Hierarchy Information to the Account Page Layout. 91
Manually Creating an Account Hierarchy. 91
Creating an Account Hierarchy During Initial Data Load. 92
Creating an Account Hierarchy After Initial Data Load. 94
Multidimensional Hierarchy Example. 97
Creating a Multidimensional Hierarchy in Salesforce Classic. 97
Creating a Multidimensional Hierarchy in Lightning Experience. 97
Creating a Hierarchy Type. 98
Enabling Alternate Hierarchy. 99
Asynchronous Hierarchy Processing. 99
Events that Trigger Hierarchy Processing. 100
Monitoring Asynchronous Hierarchy Processing. 100
Configuring Hierarchy Edit Search Fields. 101
Configuring Hierarchy Edit Search Results Fields. 102
Configuring the Hierarchy Processing Parameters. 102
Configuring the Customer 360 View. 102
Changing the Display Order of Objects. 103
Synchronize Reporting Data Batch Job. 104
Troubleshooting Hierarchies. 104

Chapter 10: Integrating Cloud MDM - Customer 360 for Salesforce with Multidomain MDM. 105

Integrating Cloud MDM - Customer 360 for Salesforce with Multidomain MDM Overview. 105
Step 1. Setting Up Informatica Cloud Application Integration. 106
Step 1. Create a Service Connector. 106
Step 2. Create a Connection. 109
Step 3. Create Processes. 110
Step 2. Configuring Remote Site Settings. 111
Step 3. Creating a Search Connection. 112

Chapter 11: Managing Multi-Org. 114

Managing Multi-Org Overview. 114
Real-Time Synchronization in a Multi-Org Implementation. 115
Synchronizing Accounts in a Multi-Org Implementation. 116
Informatica Cloud Services Workflow. 116
Real-Time and Batch Processing. 120
Enrichment Settings on the Hub Org. 120
Account Fields in the Spoke Org. 120
Authentication Parameters to Connect to the Master Org. 121
Configuring the Master Org. 121

Create a Data Source on the Master Org. . . . 121
Maintain Account Enrichment Settings on the Master Org. . . . 122
Create a Connected App on the Master Org. . . . 122
Limit the Number of Responses. . . . 123
Configuring the Spoke Org. . . . 123
Install the Unmanaged Package on the Spoke Org. . . . 124
Configure Connection to the Master Org. . . . 124
Verify the HubOrg Remote Site Setting. . . . 125
Verify the SFDC Remote Site Setting. . . . 125
Add the Visualforce Page for Real-Time Update. . . . 126
Getting Updates from the Master Org in Real-Time . . . 126
Getting Updates from the Master Org in Batch Mode. . . . 127

Chapter 12: Troubleshooting. . . . 129

Troubleshooting CC360 Configuration. . . . 129
Troubleshooting Hierarchies. . . . 131

Appendix A: Custom Settings. . . . 132

CC360 Profiles. . . . 132
Record Types. . . . 134
Default Record Type for Person Accounts. . . . 134
Setting the Default Record Type for Accounts. . . . 134
Resetting the Default Record Type of Accounts. . . . 135
CC360 Settings. . . . 135
Data Sources. . . . 137
Data Sources for a Record Type. . . . 138
Enrichment Settings. . . . 138
Enrichment Settings for a Record Type. . . . 141
Internal Settings. . . . 141
Configuring Internal Settings by Using the Setup Wizard. . . . 141
Configuring Internal Settings by Using the Salesforce Developer Console. . . . 142
Fuzzy Matching. . . . 153
Fuzzy Matching Overview. . . . 153
Matching Segment. . . . 153
Batch Acceleration Level. . . . 154
Matching Settings. . . . 155
Match Scores. . . . 155
Match Score Weighting. . . . 156
Null Scores. . . . 157
Matching and Duplicate Thresholds. . . . 157
Compare Records Page. . . . 158
International Settings. . . . 159
Job Monitor. . . . 159

Job Schedules. 160
Legal Forms. 160
Matching Settings. 161
Matching Settings for a Record Type. 164
Object Classes. 165
Performance Settings. 165
Segment Settings. 166
Segment Settings for a Record Type. 166
Synchronization Settings. 167
Synchronization Settings for a Record Type. 168
Visualforce Page Fields. 169
Configuring Custom Settings Manually. 169
Configuring Custom Settings Using the Setup Wizard. 170
Sharing the CC360 Settings. 170

Appendix B: Glossary. 171

Lightning Experience. 0
MDM Multidomain Edition. 0

Index. 174

Preface

Use the *Informatica Cloud MDM - Customer 360 for Salesforce Administrator Guide* to learn how to administer Cloud MDM - Customer 360 for Salesforce.

This guide is written for administrators who manage Cloud MDM - Customer 360 for Salesforce implementation. This guide assumes that you have administrator-level permissions to Cloud MDM - Customer 360 for Salesforce functionality and are familiar with the Salesforce environment.

Informatica Resources

Informatica provides you with a range of product resources through the Informatica Network and other online portals. Use the resources to get the most from your Informatica products and solutions and to learn from other Informatica users and subject matter experts.

Informatica Network

The Informatica Network is the gateway to many resources, including the Informatica Knowledge Base and Informatica Global Customer Support. To enter the Informatica Network, visit https://network.informatica.com.

As an Informatica Network member, you have the following options:

- Search the Knowledge Base for product resources.
- View product availability information.
- Create and review your support cases.
- Find your local Informatica User Group Network and collaborate with your peers.

Informatica Knowledge Base

Use the Informatica Knowledge Base to find product resources such as how-to articles, best practices, video tutorials, and answers to frequently asked questions.

To search the Knowledge Base, visit https://search.informatica.com. If you have questions, comments, or ideas about the Knowledge Base, contact the Informatica Knowledge Base team at KB_Feedback@informatica.com.

Informatica Documentation

Use the Informatica Documentation Portal to explore an extensive library of documentation for current and recent product releases. To explore the Documentation Portal, visit https://docs.informatica.com.

If you have questions, comments, or ideas about the product documentation, contact the Informatica Documentation team at infa_documentation@informatica.com.

Informatica Product Availability Matrices

Product Availability Matrices (PAMs) indicate the versions of the operating systems, databases, and types of data sources and targets that a product release supports. You can browse the Informatica PAMs at https://network.informatica.com/community/informatica-network/product-availability-matrices.

Informatica Velocity

Informatica Velocity is a collection of tips and best practices developed by Informatica Professional Services and based on real-world experiences from hundreds of data management projects. Informatica Velocity represents the collective knowledge of Informatica consultants who work with organizations around the world to plan, develop, deploy, and maintain successful data management solutions.

You can find Informatica Velocity resources at http://velocity.informatica.com. If you have questions, comments, or ideas about Informatica Velocity, contact Informatica Professional Services at ips@informatica.com.

Informatica Marketplace

The Informatica Marketplace is a forum where you can find solutions that extend and enhance your Informatica implementations. Leverage any of the hundreds of solutions from Informatica developers and partners on the Marketplace to improve your productivity and speed up time to implementation on your projects. You can find the Informatica Marketplace at https://marketplace.informatica.com.

Informatica Global Customer Support

You can contact a Global Support Center by telephone or through the Informatica Network.

To find your local Informatica Global Customer Support telephone number, visit the Informatica website at the following link:
https://www.informatica.com/services-and-training/customer-success-services/contact-us.html.

To find online support resources on the Informatica Network, visit https://network.informatica.com and select the eSupport option.

CHAPTER 1

Introduction to Informatica Cloud MDM - Customer 360 for Salesforce

This chapter includes the following topics:

- Cloud MDM - Customer 360 for Salesforce Overview, 11
- CC360 Architecture, 12
- Cloud MDM - Customer 360 for Salesforce Processes, 13
- Beans, Master Beans, and External Beans, 14
- Cloud MDM Tabs, 15

Cloud MDM - Customer 360 for Salesforce Overview

Cloud MDM - Customer 360 for Salesforce (CC360), is a master data management application that runs on your Salesforce org within Force.com. You can import Salesforce data and other external data into CC360. You can also create records such as accounts, contacts, and leads in CC360. You can cleanse, deduplicate, consolidate, and convert the data to get the best version of truth. You can consolidate and enrich data in Salesforce accounts to get a complete view of each customer in Salesforce.

CC360 can integrate Salesforce data with data from other enterprise applications and from third-party data providers. You can access CC360 functionality through Visualforce pages that override the standard Salesforce screens. Use CC360 to maintain a comprehensive view of each of your customers.

CC360 extends Salesforce functionality with probabilistic fuzzy matching that finds relationships between records and identifies records that are potential duplicate records. You can then merge the potential duplicate records to deduplicate your data.

You can create hierarchies among data to get a global view of an organization and its subsidiaries with CC360. You can view data in hierarchies to understand the relationships between records. You can view and manage multiple types of hierarchies.

You can run or schedule batch jobs in CC360 to perform CC360 tasks such as synchronization, contact data verification, duplicate records detection, merging the duplicates records, consolidation, and data conversion. CC360 has default custom settings but you can configure the default custom settings according to the specifications and requirements of your organization.

You can configure CC360 to integrate with Informatica Data as a Service and verify contact data. The contact data verification process identifies whether the contact details in your organization are valid or not valid.

You can integrate CC360 with Multidomain MDM and search for matching or duplicate records in the MDM Hub data sources. To integrate CC360 with Multidomain MDM, you must configure settings, such as Informatica Cloud Application Integration, remote site settings, and search connection.

CC360 Architecture

CC360 is an Apex-based Force.com cloud application for master data management. CC360 runs within a contained environment on your Salesforce org. You can access CC360 functionality through Visualforce pages that override the standard Salesforce screens. You can migrate and synchronize data into CC360.

CC360 is compiled, stored, and run entirely on the Force.com environment. CC360 is AppExchange certified which means CC360 complies with the Salesforce security and code guidelines. CC360 has Aloha App status, which means CC360 tabs and objects do not count against the limits within Salesforce. CC360 has separate resource limits and governor limits from the Salesforce application with the exception of storage and batch job limits. CC360 architecture has two layers, Salesforce.com and Force.com.

The following image shows the CC360 architecture:

Salesforce.com

Salesforce.com is a customer relationship management system. Use Salesforce.com to manage and track customer records. Salesforce.com consists of the following objects:

- Standard Salesforce objects, such as accounts, contacts, or leads.
- Custom objects, such as a project or an investment record.

Force.com

Force.com represents the Force.com environment upon which Salesforce is developed. Force.com consists of the following components:

Data storage

CC360 stores data in beans and master beans.

Beans are a staging area for data where data can be deduplicated, cleansed, and consolidated. Master beans maintain the best version of the truth and link together all the beans that relate to an account.

CC360 stores duplicate records in the duplicates table.

Custom Settings

Use the custom settings to change the default behavior of CC360 based on your Salesforce org. You can use the setup wizard to configure the custom settings.

For example, performance settings is a custom setting that optimizes the performance of data processing for CC360.

Batch jobs

You can schedule and run batch jobs in CC360 to perform tasks, such as checking for duplicates, data matching, and data migration. You can run batch jobs to asynchronously process hierarchy cloning and hierarchy edit for large hierarchies.

Multi-Org Capability

Multi-Org capability automates the cleansing, standardization, de-duplication, and master data management of data from multiple Orgs in a distributed Salesforce environment.

Batch Manager

You can use the Batch Manager to run batch jobs and manage pending, failed, and completed batch jobs. You can view the running, pending, failed, and completed jobs in the batch queue.

Hierarchy Manager

You can manage hierarchies and hierarchy-related configurations in CC360. Use hierarchies to view an organization and its subsidiaries. You can view the account hierarchy and the details of the records that belong to the account.

Cloud MDM - Customer 360 for Salesforce Processes

Batch jobs in CC360 cleanses, synchronizes, consolidate, and convert records. You can run batch jobs to manage duplicate records.

The following image shows the CC360 processes:

You can perform the following tasks to get the golden record:

Cleanse data

Cleanses data when you enter data or when you run the migrate batch jobs. Consistent data improves the quality of reporting and improves matching results and the accuracy of duplicate detection.

Synchronize data

Migrates Salesforce data into CC360. You can run the migrate batch jobs to create beans and master beans for each Salesforce object, and synchronize information from the records to the corresponding beans and master beans. You can also use an external data loader to synchronize the Salesforce records with CC360.

Verify contact data

Verifies contact data in your organization. You can run the DaaS batch job to asynchronously verify the contact data of the existing Salesforce records.

Deduplicate data

Prevents and detects duplicate records at the point of entry. You can run the merge batch job to merge the duplicate records.

Consolidate data

Identifies beans which are not associated to any master bean. CC360 associates the beans to a master bean. CC360 creates a master bean for the beans. CC360 also prioritizes the data in your Salesforce environment. The consolidation batch job considers the priority of the data sources and updates the information from the beans to their respective master beans.

Convert data

Converts external beans to Salesforce objects. You can convert master beans to accounts or leads. You can convert beans to accounts, contacts, or leads. You can also convert leads to accounts, contacts, opportunities, or tasks.

Beans, Master Beans, and External Beans

A bean record is a container for data within CC360. A master bean is closely tied to a Salesforce account record. Master beans link the related contact beans and lead beans to the account. Master beans maintain the best version of the truth. External beans store data from external data sources. CC360 creates external beans when you import data from external data sources.

Each synchronized record in CC360 has a corresponding bean record. CC360 uses beans during multiple processes, such as matching and deduplication. The use of beans minimizes the amount of time that the associated record is locked during CC360 processes.

CC360 can associate a bean with the following objects:

- A standard Salesforce record, such as an account, contact, or lead
- Records from sources outside Salesforce, such as Dun & Bradstreet
- Custom objects, such as a project or an investment record

CC360 creates a master bean when CC360 creates an account bean. The master bean maintains the relationship between all beans that are linked to an account.

Migrate your data into beans and master beans to synchronize CC360 with your source systems. Configure field sets to determine which fields you populate in the beans. CC360 creates a copy of the data in the beans and master beans.

Life Cycle of Beans and Master Beans

You must run batch jobs or use a data loader to import Salesforce records into CC360. CC360 creates a bean when you first synchronize and import a record into CC360.

When you import records, CC360 performs the following tasks:

1. Creates beans, if you flag objects for synchronization.
   If you do not flag objects for synchronization, the migrate batch job creates beans for those objects.

2. Normalizes the data for certain fields to ensure that the data adheres to the standard formats. Normalized data ensures that the data is consistent among the beans for certain fields to reduce the chance for the duplication of data and to improve the quality of matching results. CC360 standardizes fields, such as country code, legal form, and domain name.
3. Creates the fuzzy segment and the external ID.
   The fuzzy segment is a key that CC360 uses as a first step to identify matches. The external ID associates a bean with a source record.
4. Searches for potential duplicates.
   If you decide that the record is a duplicate, CC360 does not create a bean and the process ends. If CC360 does not identify any potential duplicates or the user indicates that the potential duplicates are not duplicates, CC360 creates the bean.
5. Sends the normalized data from the bean back to the CC360 Salesforce record.
6. Creates a master bean.
   For an account bean, CC360 makes a clone of the account bean to create the master bean and adds the bean to the related list of the master bean to link the bean to the master bean. For a contact or a lead bean, the matching batch job matches the bean to a master bean. CC360 adds the bean to the related list of the master bean to link the bean to the master bean and the associated account bean.

Cloud MDM Tabs

The CC360 tabs provide access to a variety of CC360 features.

The following tabs appear in your Salesforce environment when you run CC360:

Account Scout

Create an account or search for existing accounts. The Account Scout tab displays the same Cloud MDM Visualforce page that launches when you select **New** on the Accounts home page.

Beans

Beans are staging areas where the data can be de-duplicated, cleansed, and consolidated. View all beans or a filtered list of the beans.

Cloud MDM

Start batch jobs and schedule batch jobs, view scheduled batch jobs, and view completed batch jobs. You can also launch the Setup Wizard.

Duplicate Accounts

View all duplicate accounts or a filtered list of duplicate accounts. This page displays the match score of the pairs. You can also choose to ignore duplicates. You can click the duplicate name to see more details about the pairs. Select the Master Bean IDs to see more information about the master beans.

Fuzzy Matching

Test the results of your fuzzy match settings. CC360 determines matches at the master bean level. All objects must be associated with a master bean to participate in the fuzzy matching process. However, you can use the Fuzzy Data Comparison page to test match results and examine match scores before you create beans or master beans. Compare two records to test match results so that you can adjust your match settings.

Master Beans

View all master beans or a filtered list of the master beans.

Other Duplicates

View all duplicate contacts or duplicate leads or a filtered list of duplicate contacts or duplicate leads. This page displays the match score of the pairs. You can also choose to ignore duplicates. You can click the duplicate name to see more details about the pairs. Select the Master Bean IDs to see more information about the master beans.

CHAPTER 2

Configuring the Data Cleansing Settings

This chapter includes the following topics:

- Configure the Data Cleansing Settings Overview, 17
- Data Cleansing Options, 17
- Enabling Data Cleansing Options, 19
- Configuring Settings for a Picklist-Enabled Organization, 19
- Configuring Legal Forms, 20
- Configuring International Settings, 21

Configure the Data Cleansing Settings Overview

You can configure CC360 to cleanse the data in the legal form field, country field, and domain field. You can also configure CC360 to prevent users from entering invalid country data. CC360 cleanses data at the point of entry or when you run the migrate accounts batch job.

You can configure CC360 to normalize legal forms. Legal form normalization improves matching performance. You can configure CC360 to ensure the domain field consistency and to improve match results.

CC360 uses the country names and ISO codes to standardize country information for all accounts, contacts, leads, and records from other systems. International settings are default settings within CC360. The default settings are for USA. You can configure international settings to provide lookup values and replacement values for the country field.

Data Cleansing Options

You can enable legal form normalization, country name normalization, and domain normalization in CC360.

Legal form normalization is the process of extracting the legal form from the company norm and populating the legal form field with normalized data. Country name normalization is the process of standardizing the data in the country name field. Standardized country data improves reporting because it ensures that all of your data for a specific country exists in a single country category. Domain normalization ensures domain field consistency and improves the match results. CC360 uses the domain field during fuzzy matching.

Legal Form Normalization

Legal form normalization is the process of extracting the legal form from the company norm and populating the legal form field with normalized data.

Extract the legal form from the company name to improve matching performance. When you enable legal form normalization, CC360 identifies the legal form in a company name and then populates a normalized legal form in a separate field from the company norm field. By default, CC360 combines the country ISO code and the country norm to create a fuzzy matching segment in the master bean. To optimize matching results, eliminate the legal form from the fuzzy matching segment.

For example, a user might enter company name information in any of the following ways:

- Acme Limited
- Acme Ltd.
- Acme L.T.D.

If a user enters any of these names in the company name field, CC360 can populate the company norm field with Acme and populate the legal form field with LTD. CC360 uses the company norm field when it creates the fuzzy matching segment.

When you extract the legal form from the company norm field, CC360 eliminates nonunique character strings from the fuzzy matching segment, which improves match performance.

For example, if an organization has many customers with frequently occurring elements in the company names such as ABC Plumbing and Acme Plumbing. If you do not specify a legal form, CC360 creates the fuzzy matching segments as USABCPLUMBING and USACMEPLUMBING. If you add Plumbing to the legal form settings, CC360 creates the fuzzy matching segments as USABC and USACME. Fuzzy matching performance improves because the character strings in the fuzzy matching segments are more unique and distinct.

Country Name Normalization

Country name normalization is the process of standardizing the data in the country name field. Standardized country data improves reporting because it ensures that all of your data for a specific country exists in a single country category.

Normalized country data improves reporting because it ensures that all of the data for a specific country exists in a single country category. You can also enable CC360 to add the country ISO code to the record. When you enable country name normalization, CC360 replaces multiple nonstandard names against a single standard name. When you enable country name validation, you cannot enter a country name that does not appear in the international settings. You can also enable CC360 to validate the country data that a user enters.

For example, if a user enters U.S.A. or United States of America, you can configure CC360 to replace these terms with USA.

The country ISO code is important for fuzzy matching for the following reasons:

- CC360 uses the ISO code as the first two characters of the fuzzy matching segment by default.
- You can configure CC360 to disregard matches if the country codes are not identical.

Domain Normalization

Normalized domain ensures domain field consistency and improves the match results. CC360 uses the domain field during fuzzy matching.

When you enable domain normalization, CC360 populates the domain field with the domain extracted from the website field.

For example, if a user enters `http://www.informaticacloud.com/products/cloud-customer-360` or `www.informaticacloud.com` in the website field, CC360 can populate the domain field as `informaticacloud.com`.

Enabling Data Cleansing Options

You can enable legal form, country name, and domain name normalization in CC360. You can also enable country name validation.

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Salesforce Classic, perform the following tasks:
     1. In the upper-right corner of the page, select your name, and click **Setup**.
     2. Under **App Setup**, click **Develop** > **Custom Settings**.
   - In Lightning Experience, perform the following tasks:
     1. Click the quick access menu ( ), and then click **Setup**.
        The **Setup Home** page appears.
     2. Under **PLATFORM TOOLS**, click **Custom Code** > **Custom Settings**.

   The **Custom Settings** page appears.
2. Select **Manage** for CC360 settings.

   The **CC360 Settings** page appears.
3. Select **Edit** for the **Active** settings.
4. In the CC360 settings information, select one or more of the following options:
   - Extract Domain. CC360 populates the domain field with the domain extracted from the website field.
   - Extract Legal Form. Improves the matching performance.
   - Standardize Country. CC360 replaces nonstandard country names with a standard country name.
   - Validate Country. Prevents you from entering country name that does not appear in the international settings.
5. Click **Save**.

Configuring Settings for a Picklist-Enabled Organization

If you have enabled state and country picklists in your organization, you must update the records based on the countries that you extract from the picklists. The predefined field sets do not include state and country

code fields. You must create field sets to include state and country code fields and override the predefined field set configuration with the custom field set configuration that you create. You must ensure that you use the standard Salesforce state and country codes in the international settings.

1. From the **Setup** section of the **CC360 Admin** tab, click **Run the Setup Wizard**.
2. Click **Standardization Rules**.
3. In the **Country Standardization** section, clear **Standardize Country**.
4. Click **Save**.
5. Click **Done**.
6. To add the state and country code fields to a predefined field set, click **Field Set**.
7. Click **Edit Lead Field Sets**, **Edit Opportunity Field Sets**, **Edit Account Field Sets**, **Edit Contact Field Sets**, **Edit Bean Field Sets**, or **Edit Master Bean Field Sets** based on your requirement.
8. Click **New**.
9. Type a label, name, and short description for the field set.
10. Click **Save**.
11. Drag the fields including the state and country code fields from the **Field Set Properties** section to the **In the Field Set** section.
12. Click **Save**.
13. In the **Field Set Settings** page, click the **Override** icon of the field set to which you want to add the state and country code fields.
14. Select the field set that you created from the list, and click **Save**.

    The field set that you override contains the state and country code fields.
15. Set the value of the `Extract Country from Picklist` internal setting to `true`.

    The `Extract Country from Picklist` internal setting extracts the country from the picklists and updates the records.

    The following sample code sets the value of the internal setting to `true`:

    ```
    API_SettingsManager.updateInternalSetting('Extract Country from Picklist','true');
    ```

Configuring Legal Forms

You can configure the legal form field in Cloud MDM - Customer 360 for Salesforce to contain normalized data for business entity designations. You can also configure the legal form to contain words or phrases that frequently occur in company names. You can configure legal forms by editing the legal form custom settings.

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Salesforce Classic, perform the following tasks:
     1. In the upper-right corner of the page, select your name, and click **Setup**.
     2. Under **App Setup**, click **Develop** > **Custom Settings**.
   - In Lightning Experience, perform the following tasks:
     1. Click the quick access menu ( ), and then click **Setup**.
        The **Setup Home** page appears.

2. Under **PLATFORM TOOLS**, click **Custom Code** > **Custom Settings**.

The **Custom Settings** page appears.

2. Select **Manage** for **Legal Forms**.

   The **Custom Setting Legal Forms** page appears.

3. To create a legal form, click **New**.
4. In the **Name** field, enter the text that you want CC360 to extract from the company norm field.
5. In the **Legal Form** field, enter the text that you want CC360 to use to populate the legal form field.
6. Click **Save**.

Configuring International Settings

Use the international settings to define the available billing countries and their ISO codes by configuring the international settings in CC360.

If you have state and country picklists enabled in your organization, ensure that you use the standard Salesforce state and country codes in the international settings. If you need to change country ISO codes, you must get a .csv file from Informatica and use a data loading tool such as Informatica Cloud to import the .csv file into Salesforce.

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Salesforce Classic, perform the following tasks:
     1. In the upper-right corner of the page, select your name, and click **Setup**.
     2. Under **App Setup**, click **Develop** > **Custom Settings**.
   - In Lightning Experience, perform the following tasks:
     1. Click the quick access menu ( ), and then click **Setup**.
        The **Setup Home** page appears.
     2. Under **PLATFORM TOOLS**, click **Custom Code** > **Custom Settings**.

   The **Custom Settings** page appears.

2. Select **Manage** for International Settings.

   The **Custom Setting International Settings** page appears.

3. To edit international settings for a country, find the country in the list and click **Edit**.

   By default, the names of the international settings are the same as the two-letter ISO code for the country.

4. In the **Country** field, enter the country name to appear in the billing country field.
5. In the **Old Names** field, enter the country names that you want CC360 to replace.

   Use commas with no spaces on either side of a comma to separate the country entries. For example, "U.S.A.,the States,the United States."

6. Click **Save**.

CHAPTER 3

Managing Batch Jobs

This chapter includes the following topics:

- Managing Batch Jobs Overview, 22
- CC360 Admin Tab, 23
- Batch Jobs, 23
- Configuring a Custom Filter for a Batch Job, 24
- Setting the Priorities for a Batch Job, 25
- Synchronously Run Matching and Duplicate Contact Check Batch Jobs, 26
- Running a Batch Job, 26
- Pausing and Resuming Job Schedules, 27
- Rescheduling Failed Jobs, 27
- Managing the Batch Queue, 27
- Integrating External Batch Job, 31
- Viewing Batch Statistics, 33

Managing Batch Jobs Overview

Use the batch jobs in Cloud MDM - Customer 360 for Salesforce (CC360) to synchronize the Salesforce records with CC360, identify and merge the duplicate records, consolidate records to create golden records, and convert external records to Salesforce records.

The batch jobs perform specific operations on the Salesforce records and external records. You can schedule to run the batch jobs at specific time, or you can run the batch jobs in real-time.

Use the batch jobs to perform the following tasks:

- Synchronize the Salesforce records with CC360
- Identify and merge the duplicate records
- Consolidate the records to create golden records
- Convert an external record to a Salesforce record
- Process changes in large hierarchies

You can use the batch queue to monitor pending batch jobs, running batch jobs, failed batch jobs, and completed batch jobs. The batch queue displays the information about each job, such as the status of the job, the start time, and the end time. You can delete any queued or completed jobs from the batch queue.

You can view statistics for each batch job. Batch statistics report includes information such as the number of records processed during a batch job and the total number of errors.

CC360 Admin Tab

You can schedule or run batch jobs from the Informatica CC360 Admin tab.

Click the **CC360 Admin** tab to run the batch jobs. The CC360 Admin tab contains the following sections:

Jobs

Run batch jobs or schedule batch jobs.

Schedules

Lists how frequently the batch jobs are scheduled to run.

Running Jobs

Lists any batch job that are running. CC360 displays information such as the status of the batch jobs, the number of items processed, and the number of unused batch slots.

Queue Entries

Lists the pending, failed, and completed batch jobs.

Setup

Run the setup wizard.

Batch Jobs

You can schedule or manually launch batch jobs to perform multiple tasks in Cloud MDM - Customer 360 for Salesforce (CC360).

You can perform the following tasks in CC360:

Verify Addresses, email addresses, and phone numbers

You can run the Data as a Service batch job to verify addresses, email addresses, and phone number of Salesforce records.

Synchronize the Salesforce records with CC360

You can run the following batch jobs to synchronize Salesforce records with CC360:

- Migrate Accounts. Creates a bean and master bean for each Salesforce account record, and synchronizes information from account records to the corresponding beans and master beans.
- Migrate Contacts. Creates a bean and master bean for each Salesforce contact record, and then synchronizes information from the contact records to each bean and master bean.
- Migrate Leads. Creates a bean and master bean for each Salesforce lead record, and then synchronizes information from the lead records to each bean and master bean.
- Migrate Hierarchy. Establishes parent-child relationships between the beans based on the Salesforce account hierarchy.

Identify and merge duplicate records

You can run the following batch jobs to identify and merge the duplicate records:

- Duplicate Check. Detects duplicate accounts.
- Duplicate Contact Check. Detects duplicate contacts, leads, and custom objects.
- Merge Accounts. Merges duplicate accounts.
- Merge Contacts. Merges duplicate contacts.
- Merge Leads. Merges duplicate leads.

Consolidate records to create golden records

You can run the following batch jobs to consolidate the records to create golden records:

- Matching. Finds the master bean that is the best match for beans without master beans.
- Consolidation. Consolidates data from multiple data sources and creates golden records.

Convert an external record to a Salesforce record

You can run the Data Conversion batch job to convert an external record to a Salesforce record.

Process changes in large hierarchies and update the hierarchy information

CC360 runs the following batch jobs asynchronously:

- Asynchronous Hierarchy Processing. Processes hierarchy changes for hierarchies with more than 500 records.
- Asynchronous Cloning Process. Process hierarchy cloning for hierarchies with more than 500 records.
- Synchronise Reporting Data. Updates the hierarchy information for all records in an account hierarchy when account hierarchy enrichment is disabled.

Configuring a Custom Filter for a Batch Job

You can configure custom filters for batch jobs. Use custom filters to filter the records that you want to process. For example, you can configure a filter to identify records that belong to a particular company.

Note: You can configure custom filters for all batch jobs except the Data Conversion batch job and the Data as a Service batch job.

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Lightning Experience, on the navigation bar, click **CC360 Admin**.
   - In Salesforce Classic, click the **CC360 Admin** tab.

   The **Manage CC360** page appears.
2. From the **Jobs** section, select a job for which you want to configure the filter.
3. Click **Configure**.

   The **Job Template Configuration** page appears.
4. Enter a custom filter in the following format:

```
(<Filter 1> [<Operator> <Filter 2>] )
```

The following table describes the parameters used in the custom filter format:

Parameter	Description
Filter 1	Filter criteria to identify the records that you want the batch job to process. For example, `DS_Last_Updated__c > 2014-12-314T05:42:14+0000`
Filter 2	Optional. Additional filter criteria to identify the records that you want the batch job to process. For example, `DS_company__c='INFA'`.
Operator	Required if you use more than one filter criteria. Logical operator `AND`. For example, `(DS_Last_Updated__c > 2014-12-314T05:42:14+0000 AND DS_company__c='INFA')`

5. Click **Save**.

Setting the Priorities for a Batch Job

CC360 runs batch jobs based on priorities. If you enable the Run Automatically option for all batch jobs, CC360 runs the batch jobs automatically based on user-defined priorities in the performance settings.

If a user does not define priorities or defines incorrect priorities for the batch jobs in the performance settings, CC360 runs the batch jobs automatically with default priorities.

CC360 assigns the following default priorities to the batch jobs:

Batch Job Name	Priority
Matching	1
Duplicate Contact Check	2
Duplicate Check	3
Consolidation	4
Conversion	5

To run the batch jobs automatically with default priorities instead of user-defined priorities, set the value of the `Run Automatic Default Job Priority` internal setting to `True`.

The following sample code sets the value of the internal setting to `True`:

```
API_SettingsManager.enableDefaultJobPriorityForRunAutomatic(True);
```

Note: If performance settings do not exist, then CC360 creates the performance settings for the batch jobs that you enable to run automatically.

Synchronously Run Matching and Duplicate Contact Check Batch Jobs

CC360 can synchronously run the matching and the duplicate contact check batch jobs for a record whenever a user adds a record. Enable the `Synchronous Run Automatic` setting to synchronously run the matching and the duplicate contact check batch jobs.

By default, the `Synchronous Run Automatic` internal setting is disabled. Use the setup wizard to enable the setting.

Note: When the `Synchronous Run Automatic` setting is enabled, if the record being processed is part of a large segment, the duplicate contact check job and the matching job do not process the record. You have to manually run the job to process the record.

Running a Batch Job

You can run a batch job in Cloud MDM - Customer 360 for Salesforce (CC360) immediately or schedule it to run later. The batch jobs can use a single process or multiple concurrent processes to run.

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Lightning Experience, on the navigation bar, click **CC360 Admin**.
   - In Salesforce Classic, click the **CC360 Admin** tab.

   The **Manage CC360** page appears.
2. From the **Jobs** section, select a job.
3. Perform one of the following tasks:
   - To run the batch job manually, click **Run Now**.
   - To run the batch job automatically, click **Run automatically**.
     CC360 disables the **Run Now** button and hides the schedule list and the **Update Schedule** buttons. As a result, you cannot manually run the batch job or schedule the batch job to run later.
   - To schedule the batch job, select a schedule from the list and click **Update Schedule**.
4. To remove a schedule for the batch job, select Not Scheduled from the schedule list and click **Update Schedule**.
5. If the batch job can run in multiple processes, select the number of concurrent processes that you want to run for the batch job.

   You can run the following batch jobs in multiple concurrent processes:
   - Consolidation
   - Duplicate Check
   - Duplicate Contact Check
   - Matching

   All the other batch jobs run as single processes.

Pausing and Resuming Job Schedules

You can pause all scheduled jobs in the Setup Wizard. You can also resume all paused jobs.

1. From the **Setup** section of the **CC360 Admin** tab, click **Run the Setup Wizard**.
2. In the **Tools** sections, click **Pause Job Schedules** to pause all the scheduled jobs.

3. To resume all paused jobs, click **Resume Job Schedules**.

Rescheduling Failed Jobs

You can configure the CC360 scheduler to reschedule the failed jobs. By default, the job scheduler reschedules only the successfully completed jobs.

1. Based on the Salesforce environment that you use, perform the following tasks:
   - In Salesforce Classic, from the **Setup** section of the **CC360 Admin** tab, click **Run the Setup Wizard**.
   - In Lightning Experience, perform the following tasks:
     1. On the navigation menu, click **CC360 Admin**.
        The **Manage CC360** page appears.
     2. From the **Setup** section, click **Run the Setup Wizard**.
2. Click **Internal Settings**.
   The **Step 9: Internal Settings** page appears.
3. Enable **Reschedule Failed Batch Jobs**.
4. Click **Save**.

Managing the Batch Queue

You can use the batch queue to monitor pending batch jobs, running batch jobs, failed batch jobs, and completed batch jobs. In the batch queue you can enable and disable automatic refresh and you can synchronize the queue.

You can perform the following tasks in the batch queue:

- View the running jobs and the queue entries.
- Disable and enable automatic refresh.
- Synchronize the batch queue to reactivate Cloud MDM - Customer 360 for Salesforce (CC360) batch queuing after you abort all the running batch jobs.

You can view the running jobs and the queue entries. The queue entries section displays information for the batch job that you have selected in the jobs list of the **Jobs** section.

You can disable and enable automatic refresh. When you select **Enable Automatic Refresh**, the information in the Running Jobs table and the Queue Entries table refreshes every five seconds.

You must synchronize the batch queue to reactivate CC360 batch queuing after you abort all the running batch jobs. If you abort all the running batch jobs in the batch queue, CC360 stops adding batch jobs to the available batch slots. Select **Synchronize Queue** in the Batch Queue section of the **CC360 Admin** tab to reactivate CC360 batch queuing.

The following image shows the batch queue:

Batch Queue — Disable Automatic Refresh | Synchronize Queue

Running Jobs

Number	Batch Name	Loop	Run	Job Size	Processed	Queue Status	Last Phase	Last Updated	Batch Apex Id	Batch Apex Status	Created By	Actions
0000000033	Matching	2	1		20	Running	execute (6 of 24)	11/28/2016 1:32 AM	707i000006DbVny	Processing	Sarath Kasetti	Abort
0000000034	Matching	2	1		20	Running	execute (2 of 24)	11/28/2016 1:32 AM	707i000006DbVo0	Processing	Sarath Kasetti	Abort
0000000037	Matching	2	1		20	Running	execute (13 of 24)	11/28/2016 1:32 AM	707i000006DbVis	Processing	Sarath Kasetti	Abort
0000000038	DuplicateBeanCheck	1	1		0	Running	execute (8 of 50)	11/28/2016 1:32 AM	707i000006DbVnw	Processing	Sarath Kasetti	Abort
0000000041	DuplicateBeanCheck	1	1		0	Running	execute (10 of 50)	11/28/2016 1:32 AM	707i000006DbViu	Processing	Sarath Kasetti	Abort
0000000023	DuplicateCheck	3	1		40	Running		11/28/2016 1:32 AM	707i000006DbVaX	Holding	Sarath Kasetti	
0000000024	DuplicateCheck	3	1		40	Running		11/28/2016 1:32 AM	707i000006DbVpj	Holding	Sarath Kasetti	
0000000025	DuplicateCheck	3	1		40	Running		11/28/2016 1:32 AM	707i000006DbVpl	Holding	Sarath Kasetti	
0000000026	DuplicateCheck	3	1		40	Running		11/28/2016 1:32 AM	707i000006DbVpn	Holding	Sarath Kasetti	
0000000027	DuplicateCheck	3	1		40	Running		11/28/2016 1:32 AM	707i000006DbVpp	Holding	Sarath Kasetti	
0000000028	DuplicateBeanCheck	2	1		20	Running		11/28/2016 1:32 AM	707i000006DbVoV	Holding	Sarath Kasetti	
0000000029	DuplicateBeanCheck	2	1		20	Running		11/28/2016 1:32 AM	707i000006DbVob	Holding	Sarath Kasetti	
0000000030	DuplicateBeanCheck	2	1		20	Running		11/28/2016 1:32 AM	707i000006DbVod	Holding	Sarath Kasetti	
0000000031	DuplicateBeanCheck	2	1		20	Running		11/28/2016 1:32 AM	707i000006DbVok	Holding	Sarath Kasetti	
0000000032	DuplicateBeanCheck	2	1		20	Running		11/28/2016 1:32 AM	707i000006DbVol	Holding	Sarath Kasetti	
0000000035	Matching	2	1		20	Running		11/28/2016 1:32 AM	707i000006DbVo2	Holding	Sarath Kasetti	
0000000036	Matching	2	1		20	Running		11/28/2016 1:32 AM	707i000006DbVo3	Holding	Sarath Kasetti	
0000000039	DuplicateBeanCheck	1	1		0	Running		11/28/2016 1:32 AM	707i000006DbVoX	Holding	Sarath Kasetti	
0000000040	DuplicateBeanCheck	1	1		0	Running		11/28/2016 1:32 AM	707i000006DbVoZ	Holding	Sarath Kasetti	

Unused Slots 0

Queue Entries

All Queue Entries

Number	Batch Name	Processed	Status	Start Time	End Time	Created By	Depends On	Actions	Errors
0000000042	DuplicateBeanCheck	0	Waiting	11/28/2016 1:31 AM		Sarath Kasetti	0000000041	Delete	

Page 1 (1-1) First Previous Next

CC360 processes only five jobs simultaneously. If you submit more than five jobs for execution, CC360 runs the first five jobs and assigns the Processing status to the running jobs. The remaining jobs are assigned the Holding status.

If you submit more than 100 jobs for execution, CC360 performs the following tasks:

1. Runs the first five jobs and assigns the Processing status.
2. Assigns the Holding status to the next 95 jobs.
3. Assigns the Waiting status to the remaining jobs.

You can also modify the order of the Holding jobs to control when the queued jobs get processed.

Monitoring the Running Batch Jobs

You can also monitor the running batch jobs in the batch queue. Cloud MDM - Customer 360 for Salesforce (CC360) displays the number of available batch slots. When you schedule a batch job, CC360 reserves one batch slot for each scheduled job until the scheduled job runs.

You can view the following information in the running jobs table:

Number

Queue entry sequence number.

Batch Name

Apex batch job name.

Loop

Loop number of the hierarchy batch job.

Job size

Total number of records for the batch job to process.

For the asynchronous hierarchy batch job for large hierarchies, the job size column displays 'Large Hierarchy'. The total number of records that the asynchronous hierarchy batch job processed appears after the batch job completes.

Processed

Number of record processed across the loop.

Queue status

Current batch job status.

Phase detail

Current job phase. Can be "Start," "Execute," or "Finish."

Last updated

Last updated date and time.

Batch Apex ID

Apex batch job ID.

Batch Apex status

Status of the Apex batch job.

Created by

User who launched the batch job.

Actions

The user can select **Abort** to abort a batch job.

Unused Slots

Displays the number of available batch slots.

Monitoring Queue Entries

You can enter queue entries in the **Queue Entries** section of the **CC360 Admin** tab.

The **Queue Entries** section displays information for the batch job selected in the jobs list of the **Jobs** section. To view queue entries for all batch jobs, select **All** from the Jobs menu. In the **Queue Entries** section, you can choose to view pending jobs, failed jobs, or completed jobs.

You can select the batch number in the **Number** column of the queue entries table to view the batch queue details for the particular batch job. Cloud MDM - Customer 360 for Salesforce (CC360) does not remove completed batch jobs from the batch queue until you run the maintenance batch job. Run the maintenance batch job to remove completed queue entries from the batch queue.

Batch Job Statuses

You can view the status of batch jobs in the Queue Status column of the Running Jobs table.

A batch job can have one of the following statuses:

Registered

The batch job is registered, but not queued.

Waiting

The batch job is waiting for an available batch slot.

Looping

The batch job has completed a batch transaction and is queued to process the next batch transaction.

Holding

The batch job is in the Apex Flex Queue.

Queued

The batch job is in the batch job queue.

Running

Cloud MDM - Customer 360 for Salesforce (CC360) is processing the batch job.

Completed

The batch job has successfully completed.

Failed

The batch job failed.

Aborted

A user aborted the batch job.

The following image shows the batch status diagram for submitted batch jobs:

Removing the Queue Entries from the Batch Queue

You can remove old queue entries from the batch queue. You can run the maintenance batch job to remove old queue entries from the batch queue.

The maintenance batch job performs the following tasks:

1. Removes old queue entries from the batch queue.
2. Removes uncommitted transactions older than two days.
3. Removes transaction records of converted leads.
4. Removes transaction records of deleted contacts.

Integrating External Batch Job

You can use Cloud MDM - Customer 360 for Salesforce (CC360) APIs to push external jobs to the CC360 batch queue.

You can integrate external Apex batch jobs with the batch queue to get batch statuses, and you can submit external Apex batch jobs to the CC360 batch queue.

Also register a loop batch job when you want a batch job to launch after a running batch completes. The batch context of the running batch job is passed to the subsequent batch job.

The following code example shows how you can use the QueueManager API to push jobs to the CC360 batch queue:

```
        global class SampleExternalBatch implements Database.Batchable, Database.Stateful{
            private Id cloudMDMJobQueueId = null;
            private Integer itemProcessed = 0;
            private Integer totalJobItem = 1000;

            global Database.QueryLocator start(Database.BatchableContext BC)
    {
                cloudMDMJobQueueId = API_QueueManager.start(BC);
                List<string> accountNames = API_QueueManager.getExtendedParameters
    (cloudMDMJobQueueId, 'AccountName');
                List<string> billingCountry = API_QueueManager.getExtendedParameters
    (cloudMDMJobQueueId, 'BillingCountry');
                API_QueueManager.UpdateStatus(cloudMDMJobQueueId, 'start', 0);

                return Database.getQueryLocator(<>);
            }

            global void execute(Database.BatchableContext BC, List scope) {
                itemProcessed = itemProcessed + scope.size();
                API_QueueManager.updateExecutionProgress(cloudMDMJobQueueId,  totalJobItem,
    itemProcessed);
            }

            global void finish(Database.BatchableContext BC) {
                API_QueueManager.finish(cloudMDMJobQueueId,
    BATCH_STATUS.SUCCESS);
            }
        }
```

You can use the QueueManager API to perform the following tasks:

- Submit a batch job to the CC360 batch queue.
  The following code example shows how to submit jobs into the CC360 batch queue:

```
        API_QueueManager qManager = new API_QueueManager ('CustomBatch',
    'ETL_CustomApexBatchJob');
        qManager.setQuickParameters (new List<string>{'PARAM1', 'PARAM2', 'PARAM3',
    'PARAM4', 'PARAM5', 'PARAM6', 'PARAM7', 'PARAM8'});
         qManager.pushExtendedParameters('AccountName', new List<string>{'Informatica',
    'EcoLab'});
         qManager.pushExtendedParameters('BillingCountry', new List<string>{'USA', 'UK'});
         Id batchJobQueueId = qManager.submitJob();
```

- Register loop batch jobs.
  The following code example shows how to register batch loop jobs:

```
    global void finish(Database.BatchableContext BC) {

        // If more record to process register and loop next chunk or batch
        if (moreRecordsToProcess ())
        {
            Id newLoopJobId = API_QueueManager.registerJobLoop(
                            cloudMDMJobQueueId);
            API_QueueManager.submitJob (newLoopJobId);
        }
        else
        {
        // no more pending records to process. Mark current job as complete
            API_QueueManager.finish(cloudMDMJobQueueId,
                        BATCH_STATUS.SUCCESS);
        }
    }
```

Viewing Batch Statistics

Cloud MDM - Customer 360 for Salesforce (CC360) generates statistical reports about batch jobs. Use the report to view details of the last batch job that is processed by CC360. You can view the batch statistics on the **Manage CC360** page.

The **Batch Statistics Edit** page contains fields that CC360 populates when a batch job runs. You must not change the batch statistics settings. The batch statistics report includes information such as the number of records that are processed during a batch job and the total number of errors. You need to configure the batch statistics settings to be able to view the statistics of batch processes.

CC360 displays the number of items processed by each batch job in the **Completed Jobs** table of the **Manage CC360** page. To view detailed statistics for a particular batch job, select the **# Errors** link from the batch job row in the **Completed Jobs** table.

The following statistical information appears for a batch job:

Name

Name of the data set used when the custom setting data is referenced by APEX.

Avg. Script Statements

The average number of script statements generated by all runs of the batch job.

Batches Processed

Total number of batches run by the batch job.

First Error

The first error message encountered by the first run of the batch job.

First Failed Record

The record ID of the first record that fails to update.

Items Created

The total number of records created by all runs of the batch job.

Items Failed During Update

Total number of records that fail to update in all runs of the batch job.

Items Fetched

The total number of records fetched by all runs of the batch job.

Items Processed

Total number of records that are processed in all runs of the batch job.

Items Updated

The total number of records updated by all runs of the batch job.

Items Skipped Before Processing

Total number of records that are skipped before CC360 processes all runs of the batch job.

Items Skipped During Processing

The total number of records skipped during processing of all runs of the batch job.

Last End Date

The last date on which the batch job completes.

Last Item Processed

The record ID of the last record updated by the batch job.

Last Start Date

The last date on which the batch job was started.

Throw Exception

Use for debugging.

Total No. Of Comparisons

The total number of comparisons made by all runs of the batch job.

Total No. Of Errors

Total number of errors encountered by all runs of the batch job.

Total Script Statements

The total number of script statements generated by all runs of the batch job.

Note: By default, CC360 does not log batch statistics. To log batch statistics, set the `Log Batch Statistics` internal setting value to true. The following sample code sets the value of the internal setting to true:

```
API_SettingsManager.updateInternalSetting('Log Batch Statistics','true');
```

CHAPTER 4

Verifying Addresses, Email Addresses, and Phone Numbers

This chapter includes the following topics:

- Verifying Addresses, Email Addresses, and Phone Numbers Overview, 35
- Address, Email Address, and Phone Number Verification Process, 36
- Template for the DaaS Batch Job, 37
- Settings for the Contact Data Verification Process, 38

Verifying Addresses, Email Addresses, and Phone Numbers Overview

You can verify addresses, email addresses, and phone numbers with Informatica Data as a Service (DaaS). You can use the address verification service to verify addresses. You can use the email address verification service to verify email addresses. You can use the phone number validation service to validate phone numbers. You can configure Cloud MDM - Customer 360 for Salesforce (CC360) to integrate with DaaS and verify addresses, email addresses, and phone numbers.

The address, email address, and phone number verification process identifies whether the contact information in your organization are valid or not valid. You can verify addresses, email addresses, and phone numbers for Salesforce standard and custom objects. You can verify whether the address is postal-verified, deliverable, or not deliverable. You can verify whether the email address is valid, not valid, or malicious. You can verify the phone numbers to ensure your ability to connect.

You can verify the contact information when you create records in the accounts, contacts, or leads creation page. You can also run the DaaS batch job to verify the contact information of the existing Salesforce records.

Use a template to configure the settings for a DaaS batch job. You can verify all the records in your organization or specific records based on the conditions that you configure in the template.

Note: If you want to use only DaaS, you do not need to configure CC360-related settings.

Address, Email Address, and Phone Number Verification Process

You can run the DaaS job to verify the contact information. The DaaS job uses address verification service to verify address, email address verification service to verify email address, and phone number validation service to validate phone number. The DaaS job also uses service mappings to map the fields between the Salesforce fields and the DaaS fields.

Use the template for the DaaS job to schedule the job, select the objects, and the records that you want to verify. You can create multiple instances of the job to verify records based on your requirements or specifications. Every record that the DaaS job processes is stored as a transaction entry in the EventQueue table.

In the Lightning Experience, for email address and phone number fields, transaction entry is created only after you click the **Click to Verify** button. In Salesforce Classic, transaction entries for all the fields are created as you enter the values.

The following image shows how the DaaS job verifies the contact information:

Status of Records in the EventQueue Table

The DaaS job creates a transaction entry for the records processed in the EventQueue table and sets the status to `Pending`. The external adapter changes the status based on the response from the address, email address, and phone number verification service.

The DaaS job receives the response from the verification service and updates the following transaction status:

Pending

Indicates that the DaaS job has not yet verified the records.

Failed

Indicates that the DaaS job has failed to verify the records. The job updates the status for the failed records as `Failed` in the EventQueue table.

Completed

Indicates that the DaaS job has successfully verified the records. The job deletes all records that have been verified successfully from the EventQueue table. The DaaS job also creates a chatter feed for all successfully validated records.

Uncommitted

Indicates that the record belongs to any operation that is not yet saved in the database.

Not Verified

Indicates that the subject fields of the transaction records have not been verified.

Valid

Indicates that the subject fields of the transaction records have been verified and they are valid.

Invalid

Indicates that the subject fields of the transaction records have been verified but they are not valid.

Undecided

Indicates that the subject fields of the transaction records have been verified but the validity is uncertain.

Template for the DaaS Batch Job

A template for the DaaS job includes the schedule details to run the job, a list of services to run, and the object on which you want to run the selected services. Use a template to configure the settings for the DaaS job. In the template for the DaaS job, you can specify which objects you want to verify. You can schedule the job to run at different intervals. You can select which service you want to use to verify the contact data.

You can create a template for the DaaS job before you run it. You can create multiple templates for multiple instances of the DaaS job. You can run each job at different schedules on different objects.

You can configure the filters to select the records on which you want to run the DaaS batch job. If you have person accounts enabled in your organization, you must configure the Is Person Account custom filter when you create a template for the DaaS job to differentiate between business accounts and person accounts.

For example, if you select Is Person Account as the field and EQUALS as the operator and enter `true` as the value, CC360 verifies only person accounts.

The following image shows a template for the DaaS batch job:

Settings for the Contact Data Verification Process

You can run the DaaS batch job based on the schedule that you configure in the template. The DaaS batch job verifies the contact details based on the DaaS mapping, the address verification service settings, the email verification service settings, and the phone validation service settings.

Before you run the DaaS batch job, you must configure the following settings:

- Create a template for the DaaS batch job.
- Configure the address verification service settings.
- Configure the email address verification service settings.
- Configure the phone number validation service settings.
- Configure the DaaS mapping.
- Configure the account, contact, and lead page layout.
- Optionally, configure the Salesforce web service timeout interval.

Step 1. Create a Template for the DaaS Job

Create a template for the DaaS job to schedule the job, select the objects, and select the services that you want to run.

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Lightning Experience, on the navigation bar, click **CC360 Admin**.

- In Salesforce Classic, click the **CC360 Admin** tab.

The **Manage CC360** page appears.

2. From the **Jobs** section, select the **Data as a Service** job.
3. Click **Add Template**.

   The **Job Template Configuration** page appears.
4. Enter a description for the template.
5. Schedule the frequency to run the job.

   You can schedule to run the job once, daily, hourly, weekly, monthly, and yearly.
   By default, the maximum number of schedules for the DaaS job is 30. To change the default value, change the `MaxSchedulesData as a Service` internal setting value.

   The following sample code sets the value of the internal setting to 50:

   ```
   API_SettingsManager.updateInternalSetting ('MaxSchedulesData as a Service', 50);
   ```
6. Select an object to verify. You can select a Salesforce object or a custom object.

   The column filter fields appear only after you select an object.
7. Select one or more of the following verification options:
   - Verify Phone Numbers.
   - Verify Emails.
   - Verify Addresses.

   Optionally, you can exclude the records that you previously verified for each verification service. For example, you can specify to exclude the records verified within the last two days for the email verification service.
8. To select the records on which you want to run the DaaS batch job, configure the filters.

   For example, if you select Accouny Name as the field and EQUALS as the operator and enter `Informatica` as the value, CC360 verifies only the records that have `Informatica` as the account name.

   Note: For the IN, NOT IN, INCLUDES, and EXCLUDES operators, you can enter multiple values separated by commas. For the other operators, do not enter values within single quotes. If you enter values within single quotes, the single quotes are treated as part of value.
9. Click **Save**. The template is created.

   After you create a template, you can clone the template to create a new template. You can use the new template to configure another job that run on a different schedule, run on different objects, or run a different service.

Step 2. Configure the Address Verification Service Settings

The DaaS job uses the address verification service settings to verify postal addresses.

1. From the **Setup** section of the **CC360 Admin** tab, click **Run the Setup Wizard**.
2. In the **Service Setup** section, click **Address Verification**.

   The **Service: Informatica Address Verification** page appears.
3. In the **Service Information** section, activate the Informatica Address Verification service.
4. In the **Informatica Address Verification Configuration** section, configure the following subsections:

Authentication

Enter the authentication credentials to access the address verification service.

Advanced Settings

Select one or more of the following options for the point of entry verification:

- Enable Point of Entry Suggestions. Suggests the address values when you start entering the address. You must enter at least two address fields to see the suggestions.
  For example, if you enter `United States` as the Billing Country and start entering values for the Billing Street, you get suggestions for addresses in `United States`.
- Enable Point of Entry Click to Verify. Verifies the complete address that you enter when you click the **Click to Verify** button.
- Enable Point of Entry Save to Verify. Verifies the address when you click **Save** in the **Account Edit** page or in the record details page.

  Note: When you select the Enable Point of Entry Save to Verify option, CC360 does not accept address suggestions and saves the entered values in the address fields with the correct verification status.

You can also add or remove optional parameters such as Default Country, Minimum Characters for suggestions to appear, and Time Delay for Suggestions to appear (milliseconds).

Note: Default value for Default Country is United States.

Service Parameters

Optional. Add or remove the Use Transaction Pool service parameter.

You can include the Use Transaction Pool parameter to specify the transaction pool that Informatica DaaS must use for a call.

For more information about the service parameters, see *Informatica AddressDoctor® Cloud User Guide*.

Validation Parameters

Add or remove validation parameters to specify which values to validate. By default, CC360 uses the default values of the parameters.

Note: If you have state and country picklists enabled in your organization, you must add the Country Type validation parameter and set `ISO2` as the value.

For more information about the validation parameters, see *Informatica AddressDoctor Cloud User Guide*.

Enrichment Parameters

Optional. Add or remove the enrichment parameters to receive additional information for an address.

For more information about the enrichment parameters, see *Informatica AddressDoctor Cloud User Guide*.

5. Click **Save**.
6. To test the connection with the Address Verification service, click **Test Connection**.

   The **Test Connection** button is enabled only after you save the settings.

Step 3. Configure the Email Address Verification Service Settings

The DaaS batch job uses the email address verification service settings to verify email addresses.

1. From the **Setup** section of the **CC360 Admin** tab, click **Run the Setup Wizard**.
2. In the **Service Setup** section, click **Email Verification**.

   The **Service : Informatica Email Verification** page appears.
3. In the **Service Information** section, activate Informatica Email Verification service.
4. In the **Informatica Email Verification Configuration** section, configure the following subsections:

   Authentication

   Enter the authentication credentials to access the email verification service.

   Advanced Settings

   Select one or more of the following options for the point of entry verification:

   - Enable Point of Entry Click to Verify. Verifies the email address at the point of entry.
   - Enable Point of Entry Save to Verify. Verifies the email address when you save a record.

   Additional Parameters

   Optional. Enter a value for **Timeout (seconds)**. Timeout is the total number of seconds that the service runs to verify the email address. The verification request fails after the specified time period. Default is 90 seconds.

   You can also add or remove optional parameters, such as Optional Source ID.
5. Click **Save**.
6. To test the connection with the Email Address Verification service, click **Test Connection**.

   The **Test Connection** button is enabled only after you save the settings.

Step 4. Configure the Phone Number Validation Service Settings

The DaaS job uses the phone validation service settings to validate the phone numbers.

1. From the **Setup** section of the **CC360 Admin** tab, click **Run the Setup Wizard**.
2. In the **Service Setup** section, click **Phone Number Validation**.

   The **Service : Informatica Phone Number Validation** page appears.
3. In the **Service Information** section, activate Informatica Phone Verification service.
4. In the **Informatica Phone Number Validation Configuration** section, configure the following subsections:

   Authentication

   Enter the authentication credentials to access the phone number validation service.

   Advanced Settings

   Select one or more of the following options for the point of entry verification:

   - Enable Point of Entry Click to Validate. Verifies the phone number at the point of entry.
   - Enable Point of Entry Save to Validate. Verifies the phone number when you save.

   You can also add optional parameters, such as Default Country Code. The default country code is +1.

   Note: If you do not add the country code when you enter a phone number, the phone number validation service adds the default country code to the phone number.
5. Click **Save**.

6. To test the connection with the Phone Number Validation service, click **Test Connection**.

   The **Test Connection** button is enabled only after you save the settings.

Step 5. Configuring the DaaS Mapping

You can create a DaaS mapping to create a mapping between the fields of the Salesforce objects and the address, email address, and phone number verification services. You can also customize the default mappings based on your organization. For example, you can map a country field of Salesforce to the country field of Address Verification service.

If you have state and country picklists enabled in your organization, you must customize the default DaaS mappings to update the state and country fields of the Salesforce objects in the request and response mappings.

Creating a DaaS Mapping

You can create a mapping if you do not want to use the default mappings.

1. From the **Setup** section of the **CC360 Admin** tab, click **Run the Setup Wizard**.
2. In the **Service Mapping** section, click **Mapping**.

   The **DaaS Mapping** page appears.
3. Click **New Mapping**.
4. In the **Field Group Mapping Detail** section, perform the following tasks:
   a. Enter a name for the mapping.
   b. Select a Salesforce object whose fields you want to map to the service fields.
   c. Select the service for which you want to map the fields of the Salesforce objects.
5. In the **Request Mapping** section, select a Salesforce object field, and map the field to a service field.

   You can map more than one Salesforce object field to a service field. If you want suggestions to appear for AddressComplete, you must map a country field with the service country field.

   Note: When you map Salesforce object custom fields, use only custom fields with Text data type as DaaS supports only custom fields of Text data type.
6. In the **Response Mapping** section, select a service field, and map the field to a Salesforce object field.

   You can map more than one service field to a Salesforce object field. You can configure a response mapping from a service field to a Salesforce object field that indicates the status of the verification.

   Note: When you map Salesforce object custom fields, use only custom fields with Text data type as DaaS supports only custom fields of Text data type.
7. Click **Save**.

   The mapping is created and listed under the appropriate objects for which you created the mapping in the **DaaS Mapping** page.
8. Activate the mappings.

   You can edit or delete the mappings. You can also change the display order of the mappings.
9. Click **Done**.

Customizing the Default DaaS Mapping

You can customize the DaaS mappings based on the organization's requirement. If you have state and country picklists enabled in the organization, you must customize the default DaaS mappings to update the state and country Salesforce object fields in the request and response mappings.

1. From the **Setup** section of the **CC360 Admin** tab, click **Run the Setup Wizard**.
2. In the **Service Mapping** section, click **Mapping**.

   The **DaaS Mapping** page appears.
3. If you want to customize a default mapping, select the mapping in the **Account**, **Contact**, or **Lead** section and edit the mapping based on your organization's requirements.
4. If you have state and country picklists enabled in your organization, perform the following tasks:
   a. In the **Account** section, click **Edit** next to the **Billing Address** field group.
   b. In the **Request Mapping** and **Response Mapping** sections, change the **Billing State/Province** Salesforce object field to **Billing State/Province Code** and the **Billing Country** Salesforce object field to **Billing Country Code**.
   c. Click **Save**.
   d. Update the state and country Salesforce object fields to the state and country code Salesforce object fields for the following field groups:
      - Shipping Address
      - Mailing Address
      - Other Address
      - Address

Step 6. Configure the Account, Contact, and Lead Page Layout for Salesforce Classic

If you use Salesforce Classic, you must configure the account, contact, and lead page layout to include the DaaS Editor Visualforce component. You can verify the contact details in the record details page only after you add the DaaS Editor Visualforce component to the page layout.

Note: If you use Lightning Experience, you do not need to configure the DaaS Editor component in the record details page.

For more information about configuring the account, contact, and lead page layout, see the *Informatica Cloud MDM - Customer 360 for Salesforce Setup Guide*.

Step 7. Configuring the Salesforce Web Service Timeout Interval (Optional)

The default value set for the Salesforce web service timeout interval for address, email address, and phone number verification services is 120 seconds. Use the `Sfdc Webservice Timeout in Milli Secs` internal setting to configure the timeout interval.

The following sample code sets the `Sfdc Webservice Timeout in Milli Secs` internal setting value to `10000` (10 seconds):

```
API_SettingsManager.updateInternalSetting('Sfdc Webservice Timeout in Milli Secs',
'10000');
```

Step 8. Running the DaaS Job

You can run the DaaS job to verify the contact information in the Salesforce standard and custom objects. The DaaS job runs based on the schedule that you configure in the template. You can also manually run the DaaS batch job.

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Lightning Experience, on the navigation bar, click **CC360 Admin**.
   - In Salesforce Classic, click the **CC360 Admin** tab.

   The **Manage CC360** page appears.
2. From the **Jobs** section, select the **Data as a Service** job.
3. Under the **Schedules** section, click the **Run** button.

   The following image shows the **Run** button in the **Schedules** section:

   After you click the **Run** button, the job runs immediately.

   Note: If the connection between Informatica Data as a Service and Salesforce records fail, the DaaS job logs an error and skips the records for which the connection failed. The DaaS job processes the skipped records when you run the job again.
4. To refresh the schedules of the DaaS jobs, click **Refresh**.

   If you update the schedule of any DaaS job, you can refresh the schedule table to reflect the changes.

CHAPTER 5

Synchronizing Salesforce Records with CC360

This chapter includes the following topics:

- Synchronizing Salesforce Records with CC360 Overview, 45
- Synchronization Process, 45
- Settings for the Synchronization Process, 46
- Migrate Batch Jobs, 50

Synchronizing Salesforce Records with CC360 Overview

You can synchronize CC360 with Salesforce to migrate Salesforce records to CC360. You can perform CC360 operations, such as duplicate records identification, consolidation, and data conversion on the migrated records.

When you synchronize the records for the first time, CC360 creates beans and master beans for accounts and beans for other Salesforce objects. You can run the migrate batch jobs to synchronize the Salesforce records. You can also use a data loader such as Apex Data Loader to synchronize the Salesforce records.

For example, if you have 1000 accounts, 1000 contacts, and 1000 leads in a Salesforce organization, run the migrate batch jobs to create 3000 beans for all the records and 1000 master beans for the account records.

The migrate batch jobs use custom settings such as synchronization settings and performance settings. The synchronization settings define the mapping of fields from the standard and custom Salesforce objects to the CC360 beans. If the record type is enabled in Salesforce, you can define synchronization settings specific to each record type of an object. The performance settings optimize the performance of data processing for CC360.

Synchronization Process

CC360 creates beans and master beans when you synchronize Salesforce records from an organization.

Run the migrate batch jobs to synchronize Salesforce records. The migrate batch jobs create a bean for each record and a master bean for each account record. The migrate batch jobs also create master beans for the

account records. You can also synchronize Salesforce account hierarchy The migrate batch job also Establishes parent-child relationships between the beans based on the Salesforce account hierarchy.

The migrate batch jobs use custom settings, such as synchronization settings and performance settings to synchronize the records. You can configure the default custom settings based on your Salesforce organization and requirement.

The following image shows how the migrate batch jobs synchronize Salesforce records and Salesforce account hierarchies with CC360:

Settings for the Synchronization Process

The migrate batch jobs uses the default synchronization settings and the performance settings to synchronize data into CC360. You can customize the default custom settings based on your Salesforce organization and requirement.

Before you run the migrate batch jobs, you can perform the following tasks:

- Configure the synchronization settings.
- Configure CC360 to cleanse data.
- Configure the performance settings.

Configuring the Synchronization Settings

The synchronization settings define the mappings of fields from standard Salesforce and custom objects to the CC360 beans. To customize the default synchronization settings, use the setup wizard.

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Salesforce Classic, perform the following tasks:
     1. In the upper-right corner of the page, select your name, and click **Setup**.
     2. Under **App Setup**, click **Develop** > **Custom Settings**.
   - In Lightning Experience, perform the following tasks:
     1. Click the quick access menu ( ), and then click **Setup**.
        The **Setup Home** page appears.

2. Under **PLATFORM TOOLS**, click **Custom Code** > **Custom Settings**.

The **Custom Settings** page appears.

2. Select **Manage** for **Synchronization Settings**.

   The **Custom Setting Synchronization Settings** page appears.

3. To create a Synchronization Settings, click **New**.

   The **Synchronization Settings Edit** page appears.

4. Configure the following settings:

Name

Name of the object or the record type for which you want to define the synchronization settings.

You can use one of the following formats to define synchronization settings:

- For an object type, `<Object Name>`
  For example, Account.
- For one or more record types, `<Object Name>:<Record Type Name1>,<Record Type Name2>,...<Record Type NameN>`
  For example, Case:High Severity,Medium Severity indicates that the synchronization settings are applicable to the High Severity and Medium Severity record types.
- For a person account, `PersonAccount`

City

Optional. Mapping for City field on the bean. Specify the API name of the corresponding field on the bean.

Country

Optional. Mapping for Country field on the bean. Specify the API name of the corresponding country text field on the bean.
For example, `BillingCountry`.

Note: If you have country picklists enabled in your organization, specify the API name of the standard Salesforce country code field on the bean.

For example, `BillingCountryCode`.

Postal Code

Optional. Mapping for Postal Code field on the bean. Specify the API name of the corresponding field on the bean.

State

Optional. Mapping for State field on the bean. Specify the API name of the corresponding text field on the bean.
For example, `BillingState`.

Note: If you have state picklists enabled in your organization, specify the name of the standard Salesforce state code field on the bean. For example, `BillingStateCode`.

Birthdate

Optional. Mapping for Birthdate field on the bean. Specify the API name of the corresponding field on the bean.

Company

Optional. Mapping for Company field on the bean. Specify the API name of the corresponding field on the bean.

Custom Fields 1 through 20

Optional. Mapping for custom fields 1 through 20 on the bean. Specify the API name of the corresponding field on the bean.

Quick Cleansing

Optional. Accelerates the data cleansing process and reduces the number of script statements in the Salesforce triggers. Disable only during the initial data load and during migration, or if you experience problems with the order of execution of custom triggers. Default is enabled.

Domain

Optional. Mapping for Domain field on the bean. Specify the API name of the corresponding field on the bean.

Email

Optional. Mapping for Email field on the bean. Specify the API name of the corresponding field on the bean.

First Name

Optional. Mapping for First Name field on the bean. Specify the API name of the corresponding field on the bean.

Keep Data after Object Deletion

Optional. Saves a copy of the data in the data pool after deletion of the object in Salesforce. Default is disabled.

Last Name

Optional. Mapping for Last Name field on the bean. Specify the API name of the corresponding field on the bean.

Mobile Phone

Optional. Mapping for Mobile Phone field on the bean. Specify the API name of the corresponding field on the bean.

Do Not Delete Bean after Unsynchronizing

Optional. Saves the bean even if the Synchronize flag is disabled.

Parent ID

Optional. Mapping for Parent ID field on the bean. Specify the API name of the corresponding field on the bean.

Phone

Optional. Mapping for Phone field on the bean. Specify the API name of the corresponding field on the bean.

Record Type ID

Optional. Mapping for Record Type ID field on the bean. Specify the API name of the corresponding field on the bean.

Synchronize

Optional. Synchronizes the corresponding Salesforce object such as Account, Lead, and Contact. Default is enabled.

Website

Optional. Mapping for Website.

5. Click **Save**.

Configuring the Performance Settings

The performance settings optimize the performance of data processing for CC360. Use the setup wizard to configure the performance settings.

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Salesforce Classic, perform the following tasks:
     1. In the upper-right corner of the page, select your name, and click **Setup**.
     2. Under **App Setup**, click **Develop** > **Custom Settings**.
   - In Lightning Experience, perform the following tasks:
     1. Click the quick access menu (), and then click **Setup**.
        The **Setup Home** page appears.
     2. Under **PLATFORM TOOLS**, click **Custom Code** > **Custom Settings**.

   The **Custom Settings** page appears.
2. Select **Manage** for **Performance Settings**.

   The **Custom Setting Performance Settings** page appears.
3. To create a Performance Settings, click **New**.

   The **Performance Settings Edit** page appears.
4. Configure the following settings:

   Name

   Name of the performance job. Do not change the default value. Default is DuplicateCheck.

   Batch Acceleration Level

   Optional. Determines the inclusiveness of the exact match on the matching segment. You can increase the batch acceleration level to increase fuzzy matching performance. The acceleration level must be equal to the number of digits in the record count of the batch. For example, if the batch record count is 100,000, the acceleration level is 6. Default is 3.

   Batch Size

   Size of the batch that is processed. Default is 10.

   Concurrent Jobs

   Optional. Specifies the number of batch apex jobs that can be processed in parallel. CC360 can process up to 5 concurrent jobs. Configure the number of concurrent jobs to 1.

   Fuzzy Limit

   Optional. The limit for fuzzy comparisons. Default is 200.

Job Priority

Optional. Determines the priority that a job is assigned when CC360 runs in automatic mode. Highest priority is 1. If set to 0, the job is skipped. Default is 1.

Limits Threshold

The threshold limit for the batch job. Default is 90.

Loop

Optional. Enables CC360 to repeat a job until all records are checked. Default is enabled.

Query Limit

The query limit for the batch job. Default is 1000.

Run Automatic

Optional. Schedules jobs to run whenever a user adds or modifies a record. Default is disabled.

Search Acceleration Level

Optional. Determines the acceleration level of a fuzzy search. The higher the value, the faster the fuzzy logic. Increase the acceleration to achieve faster search results. Default is 3.

Note: Higher acceleration levels may reduce the number of results returned.

Segment Limit

Optional. The segmentation limit for the batch job. Default is 20.

5. Click **Save**.

Migrate Batch Jobs

You can run the migrate batch jobs to synchronize Salesforce records and Salesforce account hierarchies with Cloud MDM - Customer 360 for Salesforce (CC360).

The migrate batch jobs perform the following tasks:

1. Creates beans and master beans for records.
2. Synchronizes the information from the records to the corresponding beans and master beans.
3. Establishes parent-child relationships between the beans based on the Salesforce account hierarchy.
4. Enables the synchronization flag in the record detail page.

You can run the following migrate batch jobs to synchronize the Salesforce records:

Migrate Accounts

Creates a bean and a master bean for each Salesforce account record, and synchronizes information from account records to the corresponding beans and master beans. After synchronizing information from account records, the migrate accounts job runs the migrate hierarchy job to establish the parent-child relationships between the beans based on the Salesforce account hierarchy.

Migrate Contacts

Creates a bean for each Salesforce contact record and the synchronizes information from the contact records to each bean.

Migrate Leads

Creates a bean for each Salesforce lead record, and then synchronizes information from the lead records to each bean.

You can run the batch jobs in the Jobs section of the **CC360 Admin** tab. For more information about running, scheduling, and managing the batch jobs, see Chapter 3, “Managing Batch Jobs” on page 22.

Rerunning the Migrate Batch Jobs

The synchronization flag synchronizes the data from the Salesforce records into the beans. The migrate batch job enables the synchronization flag after successful completion. If you want to synchronize the Salesforce records again, disable the synchronization flag in the record detail page. Run the migrate batch jobs.

1. Click and open an account.

   The account details page appears.

2. In the account details page, disable the Synchronization flag.

   The following image shows the Synchronization flag:

3. Click **Save**.
4. Run the migrate batch jobs from the Jobs section of the **CC360 Admin** tab.

   For more information about running, scheduling, and managing batch jobs, see Chapter 3, “Managing Batch Jobs” on page 22.

Running the HierarchyMigration Job in the Developer Console

The HierarchyMigration job reestablishes the parent-child relationships between the beans based on the Salesforce account hierarchy. If you want to fix any hierarchy-related issues in the beans, you can manually run a HierarchyMigration job in the Salesforce Developer Console.

If you have hierarchy enabled for accounts, during synchronization of the records, the migrate jobs create record entries in a Staging Cache table. The HierarchyMigration job processes only the record entries available in the Staging Cache table.

The following sample code runs the HierarchyMigration batch job in the Salesforce Developer Console:

```
API_QueueManager qManager = new API_QueueManager ('HierarchyMigration',
'DS_HierarchyMigrateBatchClass');
 Id batchJobQueueId = qManager.submitJob();
```

CHAPTER 6

Managing Duplicate Records

This chapter includes the following topics:

- Managing Duplicate Records Overview, 52
- Duplicate Record Management Process, 52
- Configuring Settings to Identify Duplicate Records in Real Time, 53
- Settings for the Duplicate Record Management Process, 54
- Comparing Records in the Fuzzy Matching Page, 60
- Duplicate Batch Jobs, 61
- Merge Batch Job, 64
- Identifying Duplicate Records for Custom Objects, 65

Managing Duplicate Records Overview

You can run the duplicate check batch jobs to detect duplicate records. When you create a record, Cloud MDM - Customer 360 for Salesforce (CC360) identifies potential duplicate records that exist in Salesforce before it creates the record. After you find the duplicate records, you can run the merge batch job to merge the duplicate records.

You can configure CC360 to prevent duplicate records at the point of entry into Salesforce. You can identify duplicate records among the existing records in your organization. You can merge the duplicate records after you identify them.

Duplicate Record Management Process

The duplicate records management process includes identifying and merging duplicate records. You can run the duplicate check batch jobs to identify duplicate records. You can then run the merge batch job to merge the duplicate records.

The duplicate check batch jobs use the following custom settings:

- Matching Settings
- Match Score Weighting
- Null Scores

- Matching and Duplicate Thresholds
- Performance Settings

The duplicate check jobs compare the beans and the master bean of accounts, contacts, and leads to identify the duplicate records. The job stores the duplicate records in the Duplicates table.

After you identify the duplicate records, you can merge the records. Run the merge job to merge the duplicate records. The merge job eliminates the duplicate records and retains the master record. You can toggle the master record to select which record in the pair the merge job eliminates.

The following image shows how the duplicate check and the merge jobs detect and merge the duplicate records in CC360:

Configuring Settings to Identify Duplicate Records in Real Time

When the users create a record, Cloud MDM - Customer 360 for Salesforce (CC360) matches the record with the existing records in Salesforce and searches for the duplicate records.

You can configure CC360 to prevent users from creating a record when a potential duplicate record exists.

Setting Permissions for a User Profile to Identify Duplicate Records in Real Time

You can edit the Cloud MDM - Customer 360 for Salesforce (CC360) profile to prevent duplicate record creation.

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Salesforce Classic, perform the following tasks:
     1. In the upper-right corner of the page, select your name, and click **Setup**.
     2. Under **App Setup**, click **Develop** > **Custom Settings**.
   - In Lightning Experience, perform the following tasks:
     1. Click the quick access menu ( ), and then click **Setup**.
        The **Setup Home** page appears.

2. Under **PLATFORM TOOLS**, click **Custom Code** > **Custom Settings**.

The **Custom Settings** page appears.

2. Select **Manage** for **CC360 Profiles** settings.

   The **CC360 Profiles** page appears.

3. Select **Edit** for the user profile that you want to edit.
4. To prevent the creation of duplicate account records, clear **Allow Creation of Duplicate Accounts**.
5. To prevent the creation of duplicate contact records, clear **Allow Creation of Duplicate Contacts**.
6. To prevent the creation of duplicate lead records, clear **Allow Creation of Duplicate Leads**.
7. Click **Save**.

Settings for the Duplicate Record Management Process

The duplicate check jobs use the default custom settings. You can customize the default custom settings based on your Salesforce organization and requirement.

The duplicate check batch jobs use the following custom settings:

- Batch Acceleration Level
- Matching Settings
- Match Score Weighting
- Null Scores
- Matching and Duplicate Thresholds
- Performance Settings

You can configure the custom settings before you run the duplicate check jobs.

Cloud MDM - Customer 360 for Salesforce (CC360) identifies duplicate records more accurately with valid, standardized data. You can configure data cleansing settings including the fuzzy match segment, to prevent duplicate data and improve data quality. For more information about configuring data cleansing settings, see Chapter 2, “Configuring the Data Cleansing Settings” on page 17.

For optimal results, after you configure the custom settings, compare two records and test the settings.

Configuring the Batch Acceleration Level

The batch acceleration level determines the inclusiveness of the exact match on the matching segment. You can increase the batch acceleration to increase fuzzy matching performance.

1. Perform one of the following tasks based on the Salesforce environment that you use:
   - In Salesforce Classic, perform the following tasks:
     1. Select your name and click **Setup**.
     2. Under **App Setup**, select **Develop** > **Custom Settings**.

- In Lightning Experience, perform the following tasks:
  1. Click the quick access menu ( ) and then **Setup Home**.
     The **Setup Home** page appears.
  2. Under **PLATFORM TOOLS**, select **Custom Code** > **Custom Settings**.

  The **Custom Settings** page appears.

2. Select **Manage** for **Performance** settings.

   The **Performance Settings** page appears.

3. From the **Performance Setting** page, select **Edit** for the performance setting of a batch job.
4. In the **Batch Acceleration Level** field of the **Performance Settings Edit** page, enter the batch acceleration level.
5. Click **Save**.

Configuring the Matching Settings

The matching settings define weighting for each field that the matching and duplicate check processes use. If you do not change the matching settings, CC360 uses the default settings. If you want to update the matching settings, you can define matching settings for each object class or record type. CC360 returns null scores when a record has a null value for a field.

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Salesforce Classic, perform the following tasks:
     1. In the upper-right corner of the page, select your name, and click **Setup**.
     2. Under **App Setup**, click **Develop** > **Custom Settings**.
   - In Lightning Experience, perform the following tasks:
     1. Click the quick access menu ( ), and then click **Setup**.
        The **Setup Home** page appears.
     2. Under **PLATFORM TOOLS**, click **Custom Code** > **Custom Settings**.

   The **Custom Settings** page appears.

2. Select **Manage** for **Matching Settings**.

   The **Custom Setting Matching Settings** page appears.

3. To create a Matching Settings, click **New**.

   The **Matching Settings Edit** page appears.

4. Configure the following components:

   Name

   Name of the object class or the record type for which you want to define matching settings. You can use one of the following name formats to define matching settings:

   - For an object class, `<Object Class Name>`
     For example, Account.
   - For one or more record types, `<Object Class Name>:<Record Type Name1>,<Record Type Name2>,...<Record Type NameN>`

For example, Case:High Severity,Medium Severity indicates that the matching settings are applicable to the High Severity and Medium Severity record types.

- If you want to include the records of an object class that do not belong to any record type, `<Object Class>:,<Record Type Name1>,<Record Type Name2>,...<Record Type NameN>` The comma (,) after colon (:) in the name format represents the records of the object class that do not belong to any record type. For example, Account:,R1 indicates that the matching settings are applicable to the R1 record type and the account records that do not belong to any record type.

Duplicate Threshold

Optional. Threshold for being detected as a duplicate record. When the duplicate check batch job and duplicate contact check batch job run, records must have a match score that meets or exceeds the duplicate threshold. If the total match score achieves the duplicate threshold, CC360 considers the records as duplicates. Default is 90.

High Matching Threshold

Optional. The highest threshold for matching records. When the matching batch job runs, the external data record and the master bean record must have a match score that meets or exceeds the high matching threshold. If the total match score achieves the high matching threshold, CC360 considers the external data record and the master bean record a match. Default is 90.

Keep Duplicates when Re-Processing

Optional. Retains associated duplicate records even when the duplicate checked flag is reset. Default is disabled.

Low Matching Threshold

Optional. Not used.

Maximum Score

Optional. The maximum possible score. Do not populate this field. Default is no value.

Null Score Billing City

Optional. Null score for the Billing City field. Default is 100.

Null Score Billing Country

Optional. Null score for the Billing Country field. Default is 100.

Null Score Billing Postal Code

Optional. Null score for the Billing Postal Code field. Default is 100.

Null Score Billing State

Optional. Null score for the Billing State field. Default is 100.

Null Score Billing Street

Optional. Null score for the Billing Street field. Default is 100.

Null Score Birthdate

Optional. Null score for the Birthdate field. Default is 0.

Null Score Country ISO Code

Optional. Null score for the Country ISO Code field. Default is 100.

Null Score Custom Fields 1 through 6

Optional. Null score for custom fields 1 through 6.

Null Score Domain

Optional. Null score for the Domain field. Default is 0.

Null Score Email

Optional. Null score for the Email field. Default is 0.

Null Score First Name

Optional. Null score for the First Name field. Default is 100.

Null Score Last Name

Optional. Null score for the Last Name field. Default is 100.

Null Score Legal Form

Optional. Null score for the Legal Form field. Default is 100.

Null Score Mobile Phone

Optional. Null score for the Mobile Phone field. Default is 0.

Null Score Company Norm

Optional. Null score for the Company Norm field. Default is 100.

Null Score Phone

Optional. Null score for the Phone field. Default is 0.

Weighting Billing City

Optional. Weighting for the Billing City field. Default is 10.

Weighting Billing Country

Optional. Weighting for the Billing Country field. Default is 0.

Weighting Billing Postal Code

Optional. Weighting for the Billing Postal Code field. Default is 5.

Weighting Billing State

Optional. Weighting for the Billing State field. Default is 5.

Weighting Billing Street

Optional. Weighting for the Billing Street field. Default is 5.

Weighting Birthdate

Optional. Weighting for the Birthdate field. Default is 0.

Weighting Country ISO Code

Optional. Weighting of the Country ISO Code field. Weighting should be set to 200 to ensure that only accounts with the same country ISO code are detected as duplicates. Default is 200.

Weighting Custom Fields 1 through 6

Optional. Weighting for custom fields 1 through 6. The custom fields have the following behavior:

- If custom fields 1 and 2 have numeric values, CC360 performs an exact match.
- If custom fields1 and 2 have alphanumeric values, CC360 performs a fuzzy match.
- CC360 performs a fuzzy match for custom fields 3 and 4 if the weight is <=100. If the weight is >100, CC360 performs an exact match.

- CC360 performs an exact match for custom fields 5 and 6.

Weighting Domain

Optional. Weighting for the Domain field. Default is 10.

Weighting Email

Optional. Weighting for the Email field. Default is 0.

Weighting First Name

Optional. Weighting for the First Name field. Default is 0.

Weighting Last Name

Optional. Weighting for the Last Name field. Default is 0.

Weighting Legal Form

Optional. Weighting for the Legal Form field. Default is 0.

Weighting Mobile Phone

Optional. Weighting for the Mobile Phone field. Default is 0.

Weighting Company Norm

Optional. Weighting for the Company Norm field. Default is 75.

Weighting Phone

Optional. Weighting for the Phone field. Default is 10.

Enable Native Salesforce Matching

Optional. Enables CC360 to use the Salesforce Levenshtein matching algorithm.

5. Click **Save**

Configuring the Match Score Weightings

The match score weightings determines the impact a field score has on the total match score. CC360 calculates a total match score based on the total weighted field scores. The duplicate check batch jobs use the weighted field scores to identify duplicate records.

To configure the match score weightings, edit the matching settings.

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Salesforce Classic, perform the following tasks:
     1. In the upper-right corner of the page, select your name, and click **Setup**.
     2. Under **App Setup**, click **Develop** > **Custom Settings**.
   - In Lightning Experience, perform the following tasks:
     1. Click the quick access menu ( ), and then click **Setup**.
        The **Setup Home** page appears.
     2. Under **PLATFORM TOOLS**, click **Custom Code** > **Custom Settings**.

   The **Custom Settings** page appears.
2. Select **Manage** for **Matching Settings**.

   The **Custom Setting Matching Settings** page appears.
3. Click **Edit** for the object settings you want to edit or click **New** to create object settings.

4. On the **Matching Settings Edit** page, enter a weighting for each of the weighting fields.
5. Click **Save**.

Configuring the Null Scores

The null score determines how CC360 calculates the field score when one field for one of the records is null. You can configure CC360 to assign a score when one field is null so the null field does not penalize the total match score.

To configure the score, CC360 assigns a field if one of the records contains null values in the field, edit the matching settings.

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Salesforce Classic, perform the following tasks:
     1. In the upper-right corner of the page, select your name, and click **Setup**.
     2. Under **App Setup**, click **Develop** > **Custom Settings**.
   - In Lightning Experience, perform the following tasks:
     1. Click the quick access menu (), and then click **Setup**.
        The **Setup Home** page appears.
     2. Under **PLATFORM TOOLS**, click **Custom Code** > **Custom Settings**.

   The **Custom Settings** page appears.
2. Select **Manage** for **Matching Settings**.

   The **Custom Setting Matching Settings** page appears.
3. Click **Edit** for the object settings you want to edit or click **New** to create object settings.
4. On the **Matching Settings Edit** page, enter a null score value for each of the null score fields.
5. Click **Save**.

Configuring the Matching and Duplicate Thresholds

CC360 identifies duplicate records based on the matching and duplicate thresholds. You can configure the threshold that a total match score must achieve for CC360 to consider records a match or a duplicate.

To configure the matching and duplicate thresholds, edit the matching settings.

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Salesforce Classic, perform the following tasks:
     1. In the upper-right corner of the page, select your name, and click **Setup**.
     2. Under **App Setup**, click **Develop** > **Custom Settings**.
   - In Lightning Experience, perform the following tasks:
     1. Click the quick access menu (), and then click **Setup**.
        The **Setup Home** page appears.
     2. Under **PLATFORM TOOLS**, click **Custom Code** > **Custom Settings**.

   The **Custom Settings** page appears.
2. Select **Manage** for **Matching Settings**.

   The **Custom Setting Matching Settings** page appears.

3. Click **Edit** for the object settings you want to edit or click **New** to create object settings.
4. On the **Matching Settings Edit** page, enter a duplicate threshold in the duplicate threshold field.
5. Enter a matching threshold in the high matching threshold field.
6. Click **Save**.

Comparing Records in the Fuzzy Matching Page

After you configure the custom settings, you can test the settings to check the match results and determine whether the configured settings are ideal for your organization.

When you compare two records, you can see the following information in the comparison report:

- Summary. Displays the summary of the match settings.
- Segment Information. Displays the segment information for both the records.
- Details. Displays the match score, weighting, and weighted score for each field and the comparison type.

You can compare the records in Salesforce Classic or Lightning Experience.

Comparing Records in Salesforce Classic

Use the Compare Records page to compare records in Salesforce Classic.

1. Click the **Fuzzy Matching** tab.

   The **Compare Records** page appears.
2. In the **Input Parameters for Fuzzy Matching** section, select the object types of the records that you want to compare and the custom matching settings that you want to use for the comparison.
3. If you want to use the default Salesforce matching settings instead of the custom matching settings, select **Use Native Salesforce Matching**.
4. In the **Data for Fuzzy Matching** section, enter the name of the records that you want to compare.

   You can also search for the records by clicking the search icon.
5. Click **Compare**.

   The comparison report is displayed.
6. To reset the values in the fields, click **Reset**.

Comparing Records in Lightning Experience

You must open the Fuzzy Matching page from the Cloud MDM - Customer 360 for Salesforce app.

1. Click the **App Launcher** icon ( ).

   The **App Launcher** page appears.
2. Click **MDM - Customer 360 for Salesforce**.

   The **MDM - Customer 360 for Salesforce** page appears.
3. On the navigation bar, click **Fuzzy Matching**.

   The **Fuzzy Matching** page appears.
4. Select the object types of the records that you want to compare.

5. If you want to use the custom matching settings, select the appropriate custom matching settings.
6. If you want to use the default Salesforce matching algorithm instead of the custom matching settings, select **Use Native Salesforce Matching**.
7. Enter the name of the records that you want to compare.

   Note: Suggestions appear as you enter the name of the records. You can select one of the suggestions.
8. Click **Compare**.

   The comparison report is displayed.
9. To reset the values in the fields, click **Reset**.

Duplicate Batch Jobs

You can run the duplicate check batch jobs to detect duplicate accounts, contacts, leads, and custom objects in the database. The duplicate check batch jobs compile a list of duplicate pairs. Each duplicate pair consists of a master record and a duplicate record.

The duplicate check batch jobs perform the following tasks:

1. Compares beans against other beans and master beans against other master beans that have the duplicate checked flag set to false.
2. Stores the duplicate pairs in the Duplicates table.
3. Sets the duplicate checked flag for the beans and the master beans to true.

You can run the following batch jobs to detect the duplicate records:

Duplicate Check

Detects the duplicate accounts. The duplicate check batch job matches a master bean against other master beans to identify the duplicate accounts.

Duplicate Contact Check

Detects the duplicate contacts, leads, and custom objects. The duplicate contact check batch job matches a bean against other beans to identify the duplicate contacts, leads, or custom objects.

After you detect duplicates, you can view the duplicate pairs. You can select the **Duplicate Accounts** tab to view the duplicate account pairs. You can select the **Other Duplicates** tab to view duplicate contact pairs, duplicate lead pairs, and duplicate custom object pairs.

You can run the batch jobs in the Jobs section of the **CC360 Admin** tab. For more information about running, scheduling, and managing the batch jobs, see Chapter 3, “Managing Batch Jobs” on page 22.

Rerunning the Duplicate Check Batch Job

To rerun the duplicate check or duplicate contact check batch job after you adjust the batch acceleration level, clear the Duplicate Checked flags and enable Keep Duplicates when Re-Processing.

1. Use a tool such as Informatica Cloud to run an update task to clear the Duplicate Checked flag for all the records that you want to recheck for duplicates.

2. To retain the existing duplicate pairs, enable Keep Duplicates when Re-Processing.
   a. Based on the Salesforce environment that you use, perform one of the following tasks:
      - In Salesforce Classic, perform the following tasks:
        1. In the upper-right corner of the page, select your name, and click **Setup**.
        2. Under **App Setup**, click **Develop** > **Custom Settings**.
      - In Lightning Experience, perform the following tasks:
        1. Click the quick access menu (), and then click **Setup**.
           The **Setup Home** page appears.
        2. Under **PLATFORM TOOLS**, click **Custom Code** > **Custom Settings**.
   b. Select **Manage** for **Matching Settings**.

      The **Custom Setting Matching Settings** page appears.
   c. Click **Edit** for the Account settings.
   d. Enable **Keep Duplicates when Re-Processing**. Click **Save**.
3. Run the duplicate check or duplicate contact check batch job.

Disabling Duplicate Checking Between Person Accounts and Business Accounts

The duplicate check batch job searches among the person accounts and the business accounts to identify the duplicate person accounts. You cannot merge person accounts and business accounts. You can configure the duplicate check batch job to ignore potential duplicates between person accounts and business accounts. The performance of the merge batch job increases because the merge batch job does not spend time isolating duplicates between person accounts and business accounts.

To disable duplicate checking between person accounts and business accounts, edit the null score values in the matching settings.

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Salesforce Classic, perform the following tasks:
     1. In the upper-right corner of the page, select your name, and click **Setup**.
     2. Under **App Setup**, click **Develop** > **Custom Settings**.
   - In Lightning Experience, perform the following tasks:
     1. Click the quick access menu (), and then click **Setup**.
        The **Setup Home** page appears.
     2. Under **PLATFORM TOOLS**, click **Custom Code** > **Custom Settings**.

   The **Custom Settings** page appears.
2. Select **Manage** for **Synchronization Settings**.

   The **Custom Setting Synchronization Settings** page appears.
3. Select **Edit** for **Account** settings.
4. Enter "IsPersonAccount" in a **Custom Field <#>**. Note the custom field number and then click **Save**.
5. From the **Custom Setting Synchronization Settings** page, select **Edit** for **PersonAccount** settings.
6. Enter "IsPersonAccount" in the same **Custom Field <#>** as step 5. Click **Save**.
7. Repeat step 1.

8. Select **Manage** for **Matching Settings**.

   The **Custom Setting Matching Settings** page appears.

9. Select **Edit** for **Account** settings.

10. Enter "200" in the **Weighting Custom Field <#>** for the same custom field as step 5 and step 7.

    A weighting of 200 ensures an exact match for the field. An exact match for this field prevents Cloud MDM - Customer 360 for Salesforce (CC360) from identifying duplicates between person accounts and business accounts.

Enabling Duplicate Checking Among Person Accounts, Leads, and Contacts

The duplicate contact check batch job does not identify duplicates between person accounts and contacts or leads by default. To match person accounts against contacts or leads, convert the contacts or leads to person accounts., and then run the duplicate check batch job. To match person accounts against contacts or leads without converting them to person accounts, enable the lead person account comparison setting.

To enable duplicate checking among person accounts, leads, and contacts, you can set enableLeadPersonAccountComparison to true. By default, enableLeadPersonAccountComparison is false. Ensure you are aware of the impact to Cloud MDM - Customer 360 for Salesforce (CC360) batch processes when you enable duplicate checking among person accounts, leads, and contacts.

The following code sample enables duplicate checking among person accounts and leads and contacts:

```
DSE.API_SettingsManager.enableLeadPersonAccountComparison(true);
```

The following batch jobs are affected when you enable duplicate checking among person accounts, leads, and contacts:

Duplicate Contact Check

Checks for duplicates among person account contacts, leads, and business account contacts. You can see the duplicate pairs identified by this batch job in the Other Duplicates tab.

Duplicate Check

Checks for duplicates among person accounts. You can see the duplicate pairs identified by this batch job in the Duplicates tab.

Merge Accounts

Merges the person account duplicate pairs identified by the duplicate check, but does not merge duplicate pairs identified by the duplicate contact check.

Merge Contacts

Merges pairs of the same object types. For example, business account contacts are merged with business accounts contacts. Pairs that are of different object types are not merged. For example, person account contacts are not merged with business account contacts.

Merge Leads

Merges pairs of leads. Leads are not merged with objects that are not leads.

Merge Batch Job

You can run the merge accounts job, the merge contacts job, or the merge leads job to eliminate the duplicate record in a duplicate pair.

Before you run the merge accounts job, you must run the duplicate check job. The duplicate check job identifies duplicate pairs among account records. Before you run the merge contacts job or merge leads job, you must run the duplicate contact check job. The duplicate contact check job identifies duplicates among contact records, lead records, and custom object records.

Ensure each duplicate pair is a valid duplicate. You can enable the Ignore Duplicate flag to prevent Cloud MDM - Customer 360 for Salesforce (CC360) from merging a duplicate pair.

The merge job eliminates the duplicate record and keeps the master record. You can toggle the master record to select which record in the pair the merge job eliminates. If a contact is associated with a duplicate record, the merge accounts batch job associates the contact with the master record. If a duplicate record has child records, notes, attachments, and activities associated with it, the batch merge job associates these with the master record. If the duplicate record contains data that is not in the master record, such as a phone number, the data is lost when you batch merge the records. You can perform a manual merge to preserve data in a duplicate record.

During the merge process, if the accounts to be merged have hierarchy enabled, CC360 asynchronously processes hierarchy operations for both large and small hierarchies. When you run the merge account job, in the performance settings, set the Batch Size to 5 for the job to run without any exceptions. Default is 10.

Merging Contact Records

The merge contacts job eliminates the duplicate contact record and retains the master contact record. If the duplicate contact record is associated with an account record, the **Allow users to relate a contact to multiple accounts** account setting determines whether to associate the account record with the master contact record.

If this account setting is disabled, the merge contacts job eliminates the duplicate contact record and its association with the account record. If you enable this account setting, the merge contacts job eliminates the duplicate contact record and then associates the account record with the master contact record. By default, this account setting is disabled.

Merging Person Accounts

Cloud MDM - Customer 360 for Salesforce (CC360) merges person accounts with duplicate person accounts. CC360 does not merge person accounts with any other type of record, such as business accounts or contacts.

The merge accounts batch job does not merge person accounts with business accounts. If CC360 identifies a duplicate between a person account and a business account, CC360 enables the Ignore Duplicate field for the duplicate. CC360 excludes duplicates with Ignore Duplicate enabled from the merge account batch job.

CC360 does not identify duplicates between person accounts and contacts. If you manually create a duplicate relationship between a person account and a contact, CC360 ignores the duplicate and does not merge the records.

Identifying Duplicate Records for Custom Objects

If you have custom objects in your organization, you must configure CC360 to enable point of entry and batch level detection of duplicate records.

To identify duplicate records for custom objects, perform the following tasks:

1. Create an object class for the custom object.
2. Create a synchronization setting for the custom object.
3. Create a data source setting for the custom object.
4. Create a matching setting for the custom object.
5. Create a segment setting for the custom object.
6. Create an Apex trigger for the custom object.
7. Configure field sets for the custom object.

Step 1. Create an Object Class for the Custom Object

An object class indicates the object type to which a record belongs. Object classes are used in the matching settings to compare records. You can compare two records only if they are of the same object class.

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Salesforce Classic, perform the following tasks:
     1. In the upper-right corner of the page, select your name, and click **Setup**.
     2. Under **App Setup**, click **Develop** > **Custom Settings**.
   - In Lightning Experience, perform the following tasks:
     1. Click the quick access menu (), and then click **Setup**.
        The **Setup Home** page appears.
     2. Under **PLATFORM TOOLS**, click **Custom Code** > **Custom Settings**.

   The **Custom Settings** page appears.
2. For **Object Classes**, click **Manage**.

   The **Custom Settings Object Classes** page appears.
3. Click **New**.

   The **Object Classes Edit** page appears.
4. In the **Object Classes Information** section, configure the following object class settings:

   Name

   Name of the object class.

   Note: If you want to enable batch level detection of duplicate records, you must enter `Contact` as the object class name.

   Detect Duplicates from all Sources

   Enables CC360 to detect duplicates during the synchronization process. Default is enabled.

   Object Family

   Family to which the object class belongs, such as Company.
5. Click **Save**.

Step 2. Create a Synchronization Setting for the Custom Object

The synchronization settings define the mappings of fields from the custom object to the CC360 bean.

1. From the **Setup** section of the **CC360 Admin** tab, click **Run the Setup Wizard**.
2. Under **Setup**, click **Sync Fields**.
3. In the **Salesforce Sync Setup (Field Mapping)** section, click **New**.
4. In the **New Sync Mapping** section, select the custom object.

   Note: The name of the synchronization setting should be the same as the API name of the custom object.

   You can filter the objects that are displayed in the **Select an Object** list.
   For example, if you enter `Employee` and click **Filter Object Names**, the objects that contain `Employee` in the object name appears in the **Select an Object** list.
5. In the **Record Type Information** section, enter the record type for the custom object.
6. Configure the mappings for the custom object fields on the bean in the following sections:
   - Contact Information
   - Company Information

     Note: Company field mapping is required.
   - Address Information
   - Custom Information
7. Optionally, in the **Other Information** section, configure the following settings:

   Keep Data after Object Deletion

   Saves a copy of the data in the data pool after deletion of the object in Salesforce. Default is disabled.

   Synchronize

   Synchronizes the corresponding Salesforce object such as Account, Lead, and Contact. Default is enabled. Synchronizes the corresponding Salesforce object such as Account, Lead, and Contact. Default is enabled.

   Quick Cleansing

   Accelerates the data cleansing process and reduces the number of script statements in the Salesforce triggers. Disable only during the initial data load and during migration, or if you experience problems with the order of execution of custom triggers. Default is enabled.

   Do Not Delete Bean after Unsynchronizing

   Saves the bean even if the Synchronize flag is disabled.

   Parent ID

   Mapping for Parent ID field on the bean. Specify the API name of the corresponding field on the bean.

   Record Type ID

   Mapping for Record Type ID field on the bean. Specify the API name of the corresponding field on the bean.
8. Click **Save**.

Step 3. Create a Data Source Setting for the Custom Object

CC360 uses the data source settings for the consolidation process. You must also set the priority of the fields for the data source in beans.

1. From the **Setup** section of the **CC360 Admin** tab, click **Run the Setup Wizard**.
2. Under **Setup**, click **Data Sources**.
3. In the **Data Sources** section, click **New**.
4. In the **Information** section, configure the following settings:

   Name

   Name of the data source setting.

   Note: The name of the data source setting should be the same as the API name of the custom object.

   Description

   A description for the data source.

   Object Class

   Object class that you created for the custom object.
5. Configure the priority of the source on the corresponding fields on the bean in the following sections:
   - Contact Information
   - Company Information
   - Address Information
   - Custom Information
6. Click **Save**.

Step 4. Create a Matching Setting for the Custom Object

The matching settings define weighting for each custom object field that the matching and duplicate check processes use.

1. From the **Setup** section of the **CC360 Admin** tab, click **Run the Setup Wizard**.
2. Under **Setup**, click **Deduplication**.
3. In the **Deduplication** section, click **New**.
4. Optionally, in the **Other Information** section, configure the following settings:

   Name

   Name of the object class that you created for the custom object.

   Maximum Score

   The maximum possible score. Do not populate this field. Default is no value.

   Keep Duplicates when Re-Processing

   Retains associated duplicate records even when the duplicate checked flag is reset. Default is disabled.

   Duplicate Threshold

   Threshold for being detected as a duplicate record. When the duplicate check batch job and duplicate contact check batch job run, records must have a match score that meets or exceeds the

duplicate threshold. If the total match score achieves the duplicate threshold, CC360 considers the records as duplicates. Default is 90.

High Matching Threshold

The highest threshold for matching records. When the matching batch job runs, the external data record and the master bean record must have a match score that meets or exceeds the high matching threshold. If the total match score achieves the high matching threshold, CC360 considers the external data record and the master bean record a match. Default is 90.

Low Matching Threshold

Not used.

Enable Native Salesforce Matching

Mapping for Record Type ID field on the bean. Specify the API name of the corresponding field on the bean.

Native Salesforce Match Threshold Level

Threshold for detecting a matching record when CC360 uses the Saleforce matching algorithm. The Saleforce algorithm might result in slightly different match scores than the CC360 matching algorithm.

5. In the **Company Information** and **Contact Information** sections, configure null scores and weightings for the custom object fields.
6. Click **Save**.

Step 5. Create a Segment Setting for the Custom Object

The segment settings define custom segment structures for the custom object class.

1. From the **Setup** section of the **CC360 Admin** tab, click **Run the Setup Wizard**.
2. Under **Setup**, click **Advanced**.

   The **Advanced** section appears.
3. Under **Advanced**, click **Segments**.

   The **Custom Setting Segment Settings** appears.
4. Click **New**.
5. In the **Segment Settings Information** section, configure the following settings:

   Name

   Name of the object class that you created for the custom object.

   Component Fields 1 through 5

   API names of the bean fields to use to generate the segment.

   Component Length 1 through 5

   Nnumber of characters to use from the API names of beans that you specify in the component fields.
6. Click **Save**.

Step 6. Create an Apex Trigger for the Custom Object

Apex triggers enable you to perform custom actions before or after changes to the Salesforce records, such as insertions, updates, or deletions. The trigger code is stored as metadata under the custom object.

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Salesforce Classic, in the upper-right corner of the page, select your name, and click **Developer Console**.
   - In Lightning Experience, click the quick access menu (), and then click **Developer Console**.
2. In the **Force.com Developer Console** window, under **File**, select **New** > **Apex Trigger**.

   The **New Apex Trigger** window appears.
3. Enter a name for the trigger.

   Note: The name of the trigger should be the same as the API name of the custom object.
4. Select the custom object.
5. Click **Submit**.
6. In the trigger editor, enter a sample Apex code in the following format to create a trigger for the custom object:

```
trigger <TriggerName> on <CustomObject_API_Name> (after delete, after insert, after
undelete,after update, before delete, before insert, before update)
{
DSE.API_CustomScoutClass cs =new
DSE.API_CustomScoutClass('<CustomObjectName_API_Name>');
List<<CustomObject_API_Name>> <Variable1> = Trigger.new;
List<<CustomObject_API_Name>> <Variable2> = Trigger.old;
if (Trigger.isAfter)
{
if (<Variable1> != null)
<Variable1> = <Variable1>.deepClone(true, true, true);
if (<Variable2> != null)
<Variable2> = <Variable1>.deepClone(true, true, true);
}
cs.triggerSynchronize((List<SObject>) <Variable1>, (List<SObject>) <Variable2>,
Trigger.isInsert, Trigger.isUpdate, Trigger.isUnDelete, Trigger.isBefore,
Trigger.isAfter);
}
```

   The Apex code sample uses the following parameters:

   - `TriggerName`. Name of the trigger that you create.
   - `CustomObject_API_Name`. API Name of the custom object for which you create the trigger.
   - `Variable1` and `Variable2`. Variables used in the Apex code.

   The following example Apex code creates a trigger named `EmployeeTrigger` for the custom object `Employee__c`:

```
trigger EmployeeTrigger on Employee__c (after delete, after insert, after undelete,
after update, before delete, before insert, before update)
{
DSE.API_CustomScoutClass cs = new DSE.API_CustomScoutClass('Employee__c');
List<Employee__c> newEmployee = Trigger.new;
List<Employee__c> oldEmployee = Trigger.old;
if (Trigger.isAfter)
{
if (newEmployee != null)
newEmployee = newEmployee.deepClone(true, true, true);
if (oldEmployee != null)
oldEmployee = oldEmployee.deepClone(true, true, true);
}
```

```
cs.triggerSynchronize((List<SObject>) newEmployee, (List<SObject>) oldEmployee,
Trigger.isInsert,
Trigger.isUpdate, Trigger.isDelete, Trigger.isUnDelete, Trigger.isBefore,
Trigger.isAfter);
}
```

7. Under **File**, select **Save**.

Step 7. Configure Field Sets for the Custom Object

You must configure the following field sets for the custom objects:

- Search field set. Field set that you want to use in the Lightning Scout page.
- Search result field set. Field set that the search results use in the Lightning Scout page.

  Note: Search result field set for all objects are created in the bean object.

For more information about configuring field sets for custom objects, see the *Informatica Cloud MDM - Customer 360 for Salesforce Setup Guide*.

CHAPTER 7

Consolidating Records

This chapter includes the following topics:

- Consolidating Records Overview, 71
- Consolidation Process, 71
- Settings for the Consolidation Process, 72
- Matching Batch Job, 77
- Consolidation Batch Job, 78
- Converting a Master Bean to a Salesforce Object, 78

Consolidating Records Overview

Consolidation is a process of merging data from all the beans linked to a master bean to create a golden record. You can run the consolidation batch job to consolidate the records.

Cloud MDM - Customer 360 for Salesforce (CC360) uses data from the Salesforce objects and from external sources, such as SAP, ERP, or Oracle. The consolidation batch job consolidates the data from multiple beans and updates the master bean with the consolidated data to create golden records. Before you run the consolidation batch job, you can run the matching batch job to find the master bean that is the best match for each bean.

The consolidation batch job uses the data source settings and the enrichment settings if enrichment is enabled. Enrichment is the process of enhancing Salesforce records with consolidated data from the master beans after you run the consolidation batch job. Enrichment is part of the consolidation process if you enable it.

Consolidation Process

You can consolidate data from multiple data sources and create golden records. You can also enrich Salesforce records with data from the consolidated master bean records.

Before you run the consolidation job, you must run the matching batch job to find the master beans for the beans from external data sources and create master beans for the beans that are not linked to any master bean.

The following image shows how the matching job creates master beans for beans:

You can run the consolidation job on beans from multiple data sources. The consolidation job uses the data source settings and the enrichment settings. You can also enrich the Salesforce records with data from the consolidated beans if the enrichment is enabled.

During the consolidation process, CC360 performs the following tasks:

1. Compares the priorities of the fields between the external beans based on the data source settings.
2. Updates the value of the higher priority field to the corresponding field in the master bean.

During comparison, if a field has the same priority for all the external beans, CC360 checks the value of the Last Updated custom field for the beans and selects the bean with the latest date and time value. If the conflict persists, CC360 compares the bean numbers and selects the bean with the higher bean number.

You can supplement records with information from external data sources. Configure the fields of the objects that Cloud MDM - Customer 360 for Salesforce (CC360) must override when the fields in the associated master bean are updated.

You can enrich the following objects:

- Accounts, person accounts, cases, and opportunities.
- Custom objects.

The following image shows how the consolidation job consolidates data and enriches records:

Settings for the Consolidation Process

The consolidation batch job uses the default data source and enrichment settings if you enable the enrichment option. Before you run the consolidation batch job, you must also run the matching batch job that uses the default matching and synchronization settings. You can customize the default custom settings based on your Salesforce organization and requirement.

Before you run the consolidation batch job, you can perform the following tasks:

- Configure the data source settings.
- Configure the enrichment settings if you enable enrichment option.

- Configure the matching settings.
- Configure the synchronization settings.

Configuring the Data Source Settings

You can define the available data sources and set the priority of the fields for the data source in beans.

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Salesforce Classic, perform the following tasks:
     1. In the upper-right corner of the page, select your name, and click **Setup**.
     2. Under **App Setup**, click **Develop** > **Custom Settings**.
   - In Lightning Experience, perform the following tasks:
     1. Click the quick access menu (), and then click **Setup**.
        The **Setup Home** page appears.
     2. Under **PLATFORM TOOLS**, click **Custom Code** > **Custom Settings**.

   The **Custom Settings** page appears.
2. Select **Manage** for **Data Source Settings**.

   The **Custom Setting Data Sources** page appears.
3. To create a Data Source, click **New**.

   The **Data Sources Edit** page appears.
4. Configure the following components:

   Name

   Name of the data source or record type.

   If the record type is enabled, use the following name format to define data source for one or more record types:

   ```
   <Data Source>:<Record Type Name1>,<Record Type Name2>,...<Record Type NameN>
   ```

   For example, SAP:Sales,Marketing indicates that data source definition is applicable to the Sales and Marketing record types.

   Check for Duplicates on Create

   Optional. Enables CC360 to check for duplicates when a new record is created. Default is disabled.

   Default User ID

   Optional. The user identified by this ID is set as the default owner of new account, lead, and contact beans.

   Description

   Optional. A description for the data source set.

   Dynamic Priority

   Optional. The API name of the numeric field where the dynamic priority value is stored.

   Enable for Search

   Optional. Displays the source as a search option in the Account, Lead, and Contact tabs. Default is enabled.

Object Class

Optional. Object class of the data source. For example, Account, Contact, or Product.

Priority Fields

Priority of the source on the corresponding field in beans in the consolidation process. You can set the priority of the fields ranging from 0 to 100. 100 is the highest priority.

If the priority is 0, CC360 does not update the corresponding field on the consolidated record and the master bean. If the priority is 100, CC360 updates the corresponding field on the master bean. If beans have an equal priority, the bean with the most current Last Updated Date is used to update the master bean. If the Last Updated is Null, the most current modification date is used to update the master bean.

5. Click **Save**

Data Sources for a Record Type

If the record type is enabled in Salesforce, you can define data sources for each record type.

When you run a consolidation batch job, the batch job looks for the data source definition specific to each record type. If the batch job cannot find a data source definition for a record type, the batch job ignores the record.

Configuring the Enrichment Settings

You can supplement records with information from external data sources. Configure the fields of the objects that Cloud MDM - Customer 360 for Salesforce (CC360) must override when the fields in the associated master bean are updated.

To define a mapping in the custom settings, enter the API name for the account field that you need to map to in the relevant enrichment setting field. Follow the API name with a colon (:) and then the enrichment operator that describes how you want the enrichment to behave.

Use the following enrichment operators:

- - (hyphen). Ignores the field.
- 0 (zero). Updates the field only if the value is null.
- + (plus). Updates the field if the present value is not null.
- ++ (double plus). Updates the field.

For example, if you enter `BillingCity:++` in the City field, CC360 updates the field. The default operator is 0.

To define a mapping in the setup wizard, select the account field and the enrichment option for that field.

Note: When you update a date or a time field, use the following format: For date. `<yyyy>-<mm>-<dd>`

For time. `<hh>:<mm>:<ss>`

For example, `1982-03-21 11:20:32`

1. To enable the enrichment option, enable the Override Account option in the **Edit CC360 Settings** page.
2. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Salesforce Classic, perform the following tasks:
     1. In the upper-right corner of the page, select your name, and click **Setup**.
     2. Under **App Setup**, click **Develop** > **Custom Settings**.

- In Lightning Experience, perform the following tasks:

  1. Click the quick access menu ( ), and then click **Setup**.
     The **Setup Home** page appears.
  2. Under **PLATFORM TOOLS**, click **Custom Code** > **Custom Settings**.

  The **Custom Settings** page appears.

3. Select **Manage** for **Enrichment Settings**.

   The **Custom Setting Enrichment Settings** page appears.

4. To create an enrichment setting, click **New**.

   The **Enrichment Settings Edit** page appears.

5. Configure the following fields:

   Name

   Name of the object type or the record type for which you want to define the enrichment settings. Enrichment settings must have the same name as the object to facilitate mapping of fields from the account to the master bean.

   You can use one of the following name formats to define enrichment settings:

   - For an object type, `<Object Name>`
     For example, Account.
   - For one or more record types, `<Object Name>:<Record Type Name1>,<Record Type Name2>,...<Record Type NameN>`
     For example, Account:Personal,Sales indicates that the enrichment settings are applicable to the Personal and Sales record types if the record type is enabled.

   City

   Optional. Mapping for the city field on the bean. Specify the API name of the corresponding field on the bean.

   Country

   Optional. Mapping for the country field on the bean. Specify the API name of the corresponding field on the bean.

   Postal Code

   Optional. Mapping for the postal code field on the bean. Specify the API name of the corresponding field on the bean.

   State

   Optional. Mapping for the state field on the bean. Specify the API name of the corresponding field on the bean.

   Street

   Optional. Mapping for the street field on the bean. Specify the API name of the corresponding field on the bean.

   Birthdate

   Optional. Mapping for the birthdate field on the bean. Specify the API name of the corresponding field on the bean.

Company

Optional. Mapping for the company field on the bean. Specify the API name of the corresponding field on the bean.

Custom Fields 1 through 20

Optional. Mapping for custom fields 1 through 20 on the bean. Specify the API name of the corresponding field on the bean.

Quick Cleansing

Optional. Accelerates the data cleansing process and reduces the number of script statements in the Salesforce.com triggers. Default is disabled.

Domain

Optional. Mapping for the domain field on the bean. Specify the API name of the corresponding field on the bean.

Email

Optional. Mapping for the email field on the bean. Specify the API name of the corresponding field on the bean.

First Name

Optional. Mapping for the first name field on the bean. Specify the API name of the corresponding field on the bean.

Keep Data after Object Deletion

Optional. Saves a copy of the data in the data pool after deletion of the object in Salesforce.com. Default is disabled.

Last Name

Optional. Mapping for the last name field on the bean. Specify the API name of the corresponding field on the bean.

Mobile Phone

Optional. Mapping for the mobile phone field on the bean. Specify the API name of the corresponding field on the bean.

Parent Account

Optional. Mapping for the parent account field on the bean. Specify the API name of the corresponding field on the bean.

Parent ID

Optional. Mapping for the parent ID field on the bean. Specify the API name of the corresponding field on the bean.

Phone

Optional. Mapping for the phone field on the bean. Specify the API name of the corresponding field on the bean.

Record Type ID

Optional. Mapping for the record type ID field on the bean. Specify the API name of the corresponding field on the bean.

Synchronize

Optional. Automatically synchronizes the corresponding Salesforce.com object such as Account, Lead, and Contact. Default is enabled.

Ultimate Parent Name

Optional. Mapping for the name of the ultimate parent in the hierarchy.

Ultimate Parent

Optional. Mapping for the ID of the ultimate parent in the hierarchy.

Website

Optional. Mapping for Website field on the bean. Specify the API name of the corresponding field on the bean.

6. Click **Save**

Enrichment Settings for a Record Type

If the record type is enabled in Salesforce, you can define enrichment settings specific to each record type of an object.

When you run a consolidation batch job, the batch job looks for the enrichment settings in the following order:

1. Enrichment settings specific to each record type.
2. If the enrichment settings are not available for a record type, the batch job looks for the enrichment settings defined for the object to which the record type belongs.

   For example, if you do not define enrichment settings for an account record type named Account:Personal, the batch job looks for the enrichment settings defined for the Account object.
3. If the enrichment settings are not available for the object to which the record type belongs, the batch job uses the default enrichment settings of the object.

Matching Batch Job

Before you run the consolidation batch job, you must run the matching batch job to match beans against master beans. The matching batch job finds the master bean that is the best match for beans that are not linked to any master bean. If no match is found, the matching batch job creates a master bean for the bean.

The matching batch job performs the following tasks:

1. Finds the master bean that is the best match for each bean.
2. Creates a master bean for the bean if no match is found.

You can run the batch jobs in the Jobs section of the **CC360 Admin** tab. For more information about running, scheduling, and managing the batch jobs, see Chapter 3, “Managing Batch Jobs” on page 22.

Consolidation Batch Job

After you run the matching batch job, you can run the consolidation batch job to merge data from all the beans linked to a master bean to create a golden record.

The consolidation batch job performs the following tasks:

1. Consolidates data from multiple beans linked to a master bean.

   Note: The consolidation batch job consolidates data based on the priority of the source on the corresponding field in the beans. You can configure the priority of the fields in the **Data Source Settings Edit** page.

2. Updates the master bean with the consolidated data to create golden records.
3. If enrichment is enabled, enriches Salesforce records with the data from the consolidated master bean.

Note: The beans that have the Account Deleted, Lead Deleted, or Contact Deleted flag enabled in the bean details page do not participate in consolidation. If you want the deleted beans to participate in the consolidation process, set the `Ignore Deleted Beans` internal setting value to false.

The following sample code sets the value of the internal setting to false:

```
API_SettingsManager.updateInternalSetting('Ignore Deleted Beans','false');
```

You can run the batch jobs in the Jobs section of the **CC360 Admin** tab. For more information about running, scheduling, and managing the batch jobs, see Chapter 3, “Managing Batch Jobs” on page 22.

Converting a Master Bean to a Salesforce Object

After you run the consolidation batch job, you can have master beans that are not linked to any account, person account, or lead. Based on your requirement, you can convert these master beans to accounts, person accounts, or leads.

1. On the **Master Beans** tab, open the master bean that you want to convert.
2. Click **Edit**.
3. In the **Actions** section, select one of the following options:
   - Convert to Account
   - Convert to Person Account
   - Convert to Lead
4. Optionally, in the **Convert to Record Type** field, specify the record type to which you want to assign the Salesforce object.
5. Click **Save**.

   The master bean is converted to the selected object, and a bean is created for the object and is linked to the master bean.

CHAPTER 8

Converting External Records to Salesforce Records

This chapter includes the following topics:

- Converting External Records to Salesforce Records Overview, 79
- Data Conversion Process, 80
- Settings for the Data Conversion Process, 80
- Configuring the Synchronization Settings, 81
- Configuring the Matching Settings, 83
- Comparing Records in the Fuzzy Matching Page, 87
- Data Conversion Batch Job, 88

Converting External Records to Salesforce Records Overview

If you import records from external data sources as beans, you can convert the beans to Salesforce records. Run the data conversion batch job to convert the external beans to account, contact, or lead beans.

When you import records from external data sources into Cloud MDM - Customer 360 for Salesforce (CC360), CC360 creates a bean for each record. The data conversion batch job converts the beans from external data sources to accounts, contacts, or leads. When you convert an external bean to an account, CC360 creates a bean and master bean for the external bean and links the bean and the external bean to the master bean. When you convert an external bean to a contact or lead, CC360 creates a bean and links the bean and the external bean together.

Data Conversion Process

You can convert beans from external data sources to Salesforce objects.

Run the data conversion batch job on the external beans and the beans that belong to Salesforce objects. The data conversion job performs the following tasks:

1. Compares between the following beans and identifies duplicate beans:
   - The external beans that you want to convert with the beans that belong to the Salesforce objects.
   - All the external beans that you want to convert.
2. Converts all the beans that do not have any duplicates to accounts, contacts, or leads.

The duplicate beans identification before conversion ensures that duplicate beans do not exist in the organization.

If you convert a lead to an opportunity, CC360 creates tasks for that opportunity. If you do not want to create tasks for the opportunities, set the `Create Opportunity Task` internal setting value to false.

The following sample code sets the value of the internal setting to false:

```
API_SettingsManager.updateInternalSetting('Create Opportunity Task','false');
```

The data conversion batch job uses the synchronization settings and the matching settings.

The following image shows how the data conversion batch job converts external beans to Salesforce objects:

Note: If you have the Require Validation for Converted Leads flag disabled in the **Lead Settings** page, set the `Disable Validation On Converted Lead` internal setting value to true.

The following sample code sets the value of the internal setting to true:

```
API_SettingsManager.updateInternalSetting('Disable Validation On Converted Lead','true');
```

Settings for the Data Conversion Process

The data conversion batch job uses the default synchronization settings and the match settings to convert beans from external data sources to Salesforce records. You can customize the default custom settings based on your Salesforce organization and requirement.

Before you run the data conversion batch job, you can perform the following tasks:

- Configure the synchronization settings.
- Configure the matching settings.

If a bean is from an external data source, if you have state and country picklists enabled in your organization, enter the correct state and country codes in the bean details page.

For example, if the bean is from an external data source with United States as the country and California as the state, enter `US` in the country field and `CA` in the state field.

For optimal results, after you configure the custom settings, compare two records and test the settings.

Configuring the Synchronization Settings

The synchronization settings define the mappings of fields from standard Salesforce and custom objects to the CC360 beans. To customize the default synchronization settings, use the setup wizard.

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Salesforce Classic, perform the following tasks:
     1. In the upper-right corner of the page, select your name, and click **Setup**.
     2. Under **App Setup**, click **Develop** > **Custom Settings**.
   - In Lightning Experience, perform the following tasks:
     1. Click the quick access menu ( ), and then click **Setup**.
        The **Setup Home** page appears.
     2. Under **PLATFORM TOOLS**, click **Custom Code** > **Custom Settings**.

   The **Custom Settings** page appears.
2. Select **Manage** for **Synchronization Settings**.

   The **Custom Setting Synchronization Settings** page appears.
3. To create a Synchronization Settings, click **New**.

   The **Synchronization Settings Edit** page appears.
4. Configure the following settings:

   Name

   Name of the object or the record type for which you want to define the synchronization settings.

   You can use one of the following formats to define synchronization settings:

   - For an object type, `<Object Name>`
     For example, Account.
   - For one or more record types, `<Object Name>:<Record Type Name1>,<Record Type Name2>,...<Record Type NameN>`
     For example, Case:High Severity,Medium Severity indicates that the synchronization settings are applicable to the High Severity and Medium Severity record types.
   - For a person account, `PersonAccount`

   City

   Optional. Mapping for City field on the bean. Specify the API name of the corresponding field on the bean.

Country

Optional. Mapping for Country field on the bean. Specify the API name of the corresponding country text field on the bean.
For example, `BillingCountry`.

Note: If you have country picklists enabled in your organization, specify the API name of the standard Salesforce country code field on the bean.

For example, `BillingCountryCode`.

Postal Code

Optional. Mapping for Postal Code field on the bean. Specify the API name of the corresponding field on the bean.

State

Optional. Mapping for State field on the bean. Specify the API name of the corresponding text field on the bean.
For example, `BillingState`.

Note: If you have state picklists enabled in your organization, specify the name of the standard Salesforce state code field on the bean. For example, `BillingStateCode`.

Birthdate

Optional. Mapping for Birthdate field on the bean. Specify the API name of the corresponding field on the bean.

Company

Optional. Mapping for Company field on the bean. Specify the API name of the corresponding field on the bean.

Custom Fields 1 through 20

Optional. Mapping for custom fields 1 through 20 on the bean. Specify the API name of the corresponding field on the bean.

Quick Cleansing

Optional. Accelerates the data cleansing process and reduces the number of script statements in the Salesforce triggers. Disable only during the initial data load and during migration, or if you experience problems with the order of execution of custom triggers. Default is enabled.

Domain

Optional. Mapping for Domain field on the bean. Specify the API name of the corresponding field on the bean.

Email

Optional. Mapping for Email field on the bean. Specify the API name of the corresponding field on the bean.

First Name

Optional. Mapping for First Name field on the bean. Specify the API name of the corresponding field on the bean.

Keep Data after Object Deletion

Optional. Saves a copy of the data in the data pool after deletion of the object in Salesforce. Default is disabled.

Last Name

Optional. Mapping for Last Name field on the bean. Specify the API name of the corresponding field on the bean.

Mobile Phone

Optional. Mapping for Mobile Phone field on the bean. Specify the API name of the corresponding field on the bean.

Do Not Delete Bean after Unsynchronizing

Optional. Saves the bean even if the Synchronize flag is disabled.

Parent ID

Optional. Mapping for Parent ID field on the bean. Specify the API name of the corresponding field on the bean.

Phone

Optional. Mapping for Phone field on the bean. Specify the API name of the corresponding field on the bean.

Record Type ID

Optional. Mapping for Record Type ID field on the bean. Specify the API name of the corresponding field on the bean.

Synchronize

Optional. Synchronizes the corresponding Salesforce object such as Account, Lead, and Contact. Default is enabled.

Website

Optional. Mapping for Website.

5. Click **Save**.

Configuring the Matching Settings

The matching settings define weighting for each field that the matching and duplicate check processes use. If you do not change the matching settings, CC360 uses the default settings. If you want to update the matching settings, you can define matching settings for each object class or record type. CC360 returns null scores when a record has a null value for a field.

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Salesforce Classic, perform the following tasks:
     1. In the upper-right corner of the page, select your name, and click **Setup**.
     2. Under **App Setup**, click **Develop** > **Custom Settings**.
   - In Lightning Experience, perform the following tasks:
     1. Click the quick access menu ( ), and then click **Setup**.
        The **Setup Home** page appears.
     2. Under **PLATFORM TOOLS**, click **Custom Code** > **Custom Settings**.

   The **Custom Settings** page appears.

2. Select **Manage** for **Matching Settings**.

   The **Custom Setting Matching Settings** page appears.

3. To create a Matching Settings, click **New**.

   The **Matching Settings Edit** page appears.

4. Configure the following components:

   Name

   Name of the object class or the record type for which you want to define matching settings. You can use one of the following name formats to define matching settings:

   - For an object class, `<Object Class Name>`
     For example, Account.
   - For one or more record types, `<Object Class Name>:<Record Type Name1>,<Record Type Name2>,...<Record Type NameN>`
     For example, Case:High Severity,Medium Severity indicates that the matching settings are applicable to the High Severity and Medium Severity record types.
   - If you want to include the records of an object class that do not belong to any record type, `<Object Class>:,<Record Type Name1>,<Record Type Name2>,...<Record Type NameN>`
     The comma (,) after colon (:) in the name format represents the records of the object class that do not belong to any record type. For example, Account:,R1 indicates that the matching settings are applicable to the R1 record type and the account records that do not belong to any record type.

   Duplicate Threshold

   Optional. Threshold for being detected as a duplicate record. When the duplicate check batch job and duplicate contact check batch job run, records must have a match score that meets or exceeds the duplicate threshold. If the total match score achieves the duplicate threshold, CC360 considers the records as duplicates. Default is 90.

   High Matching Threshold

   Optional. The highest threshold for matching records. When the matching batch job runs, the external data record and the master bean record must have a match score that meets or exceeds the high matching threshold. If the total match score achieves the high matching threshold, CC360 considers the external data record and the master bean record a match. Default is 90.

   Keep Duplicates when Re-Processing

   Optional. Retains associated duplicate records even when the duplicate checked flag is reset. Default is disabled.

   Low Matching Threshold

   Optional. Not used.

   Maximum Score

   Optional. The maximum possible score. Do not populate this field. Default is no value.

   Null Score Billing City

   Optional. Null score for the Billing City field. Default is 100.

   Null Score Billing Country

   Optional. Null score for the Billing Country field. Default is 100.

Null Score Billing Postal Code

Optional. Null score for the Billing Postal Code field. Default is 100.

Null Score Billing State

Optional. Null score for the Billing State field. Default is 100.

Null Score Billing Street

Optional. Null score for the Billing Street field. Default is 100.

Null Score Birthdate

Optional. Null score for the Birthdate field. Default is 0.

Null Score Country ISO Code

Optional. Null score for the Country ISO Code field. Default is 100.

Null Score Custom Fields 1 through 6

Optional. Null score for custom fields 1 through 6.

Null Score Domain

Optional. Null score for the Domain field. Default is 0.

Null Score Email

Optional. Null score for the Email field. Default is 0.

Null Score First Name

Optional. Null score for the First Name field. Default is 100.

Null Score Last Name

Optional. Null score for the Last Name field. Default is 100.

Null Score Legal Form

Optional. Null score for the Legal Form field. Default is 100.

Null Score Mobile Phone

Optional. Null score for the Mobile Phone field. Default is 0.

Null Score Company Norm

Optional. Null score for the Company Norm field. Default is 100.

Null Score Phone

Optional. Null score for the Phone field. Default is 0.

Weighting Billing City

Optional. Weighting for the Billing City field. Default is 10.

Weighting Billing Country

Optional. Weighting for the Billing Country field. Default is 0.

Weighting Billing Postal Code

Optional. Weighting for the Billing Postal Code field. Default is 5.

Weighting Billing State

Optional. Weighting for the Billing State field. Default is 5.

Weighting Billing Street

Optional. Weighting for the Billing Street field. Default is 5.

Weighting Birthdate

Optional. Weighting for the Birthdate field. Default is 0.

Weighting Country ISO Code

Optional. Weighting of the Country ISO Code field. Weighting should be set to 200 to ensure that only accounts with the same country ISO code are detected as duplicates. Default is 200.

Weighting Custom Fields 1 through 6

Optional. Weighting for custom fields 1 through 6. The custom fields have the following behavior:

- If custom fields 1 and 2 have numeric values, CC360 performs an exact match.
- If custom fields1 and 2 have alphanumeric values, CC360 performs a fuzzy match.
- CC360 performs a fuzzy match for custom fields 3 and 4 if the weight is <=100. If the weight is >100, CC360 performs an exact match.
- CC360 performs an exact match for custom fields 5 and 6.

Weighting Domain

Optional. Weighting for the Domain field. Default is 10.

Weighting Email

Optional. Weighting for the Email field. Default is 0.

Weighting First Name

Optional. Weighting for the First Name field. Default is 0.

Weighting Last Name

Optional. Weighting for the Last Name field. Default is 0.

Weighting Legal Form

Optional. Weighting for the Legal Form field. Default is 0.

Weighting Mobile Phone

Optional. Weighting for the Mobile Phone field. Default is 0.

Weighting Company Norm

Optional. Weighting for the Company Norm field. Default is 75.

Weighting Phone

Optional. Weighting for the Phone field. Default is 10.

Enable Native Salesforce Matching

Optional. Enables CC360 to use the Salesforce Levenshtein matching algorithm.

5. Click **Save**

Comparing Records in the Fuzzy Matching Page

After you configure the custom settings, you can test the settings to check the match results and determine whether the configured settings are ideal for your organization.

When you compare two records, you can see the following information in the comparison report:

- Summary. Displays the summary of the match settings.
- Segment Information. Displays the segment information for both the records.
- Details. Displays the match score, weighting, and weighted score for each field and the comparison type.

You can compare the records in Salesforce Classic or Lightning Experience.

Comparing Records in Salesforce Classic

Use the Compare Records page to compare records in Salesforce Classic.

1. Click the **Fuzzy Matching** tab.

   The **Compare Records** page appears.
2. In the **Input Parameters for Fuzzy Matching** section, select the object types of the records that you want to compare and the custom matching settings that you want to use for the comparison.
3. If you want to use the default Salesforce matching settings instead of the custom matching settings, select **Use Native Salesforce Matching**.
4. In the **Data for Fuzzy Matching** section, enter the name of the records that you want to compare.

   You can also search for the records by clicking the search icon.
5. Click **Compare**.

   The comparison report is displayed.
6. To reset the values in the fields, click **Reset**.

Comparing Records in Lightning Experience

You must open the Fuzzy Matching page from the Cloud MDM - Customer 360 for Salesforce app.

1. Click the **App Launcher** icon ( ).

   The **App Launcher** page appears.
2. Click **MDM - Customer 360 for Salesforce**.

   The **MDM - Customer 360 for Salesforce** page appears.
3. On the navigation bar, click **Fuzzy Matching**.

   The **Fuzzy Matching** page appears.
4. Select the object types of the records that you want to compare.
5. If you want to use the custom matching settings, select the appropriate custom matching settings.
6. If you want to use the default Salesforce matching algorithm instead of the custom matching settings, select **Use Native Salesforce Matching**.
7. Enter the name of the records that you want to compare.

   Note: Suggestions appear as you enter the name of the records. You can select one of the suggestions.
8. Click **Compare**.

The comparison report is displayed.

9. To reset the values in the fields, click **Reset**.

Data Conversion Batch Job

You can run the data conversion batch job to convert beans from external data sources to Salesforce records.

The data conversion batch job performs the following tasks:

1. If the Convert without Duplicate Check option is disabled in the beans detail page, checks for duplicate beans before converting the external beans.
2. Converts external beans to accounts, contacts, or leads.
3. If the Convert after Consolidation option is enabled in the bean details page, converts the consolidated master bean to accounts, contacts, or leads.
4. Converts the bean to the record type that you specify in the bean details page.

You can convert the records based on the following available options in the bean detail page:

- Convert without Duplicate Check. If enabled, converts the bean without checking for duplicate beans.
- Convert after Consolidation. If enabled, converts the consolidated master bean to accounts, contacts, or leads.

Data Conversion after Consolidation Process

You can enable data conversion after consolidation to convert the consolidated records.

If you have data conversion enabled after consolidation, the data conversion batch job converts the consolidated master beans to accounts, contacts, or leads. If the data conversion batch job is enabled, the batch job runs automatically after the consolidation batch job.

CHAPTER 9

Managing Hierarchies

This chapter includes the following topics:

- Hierarchies Overview, 89
- Adding Hierarchy Information to the Account Page Layout, 91
- Manually Creating an Account Hierarchy, 91
- Multidimensional Hierarchy Example, 97
- Creating a Hierarchy Type, 98
- Asynchronous Hierarchy Processing, 99
- Configuring Hierarchy Edit Search Fields, 101
- Configuring Hierarchy Edit Search Results Fields, 102
- Configuring the Hierarchy Processing Parameters, 102
- Configuring the Customer 360 View, 102
- Changing the Display Order of Objects, 103
- Synchronize Reporting Data Batch Job, 104
- Troubleshooting Hierarchies, 104

Hierarchies Overview

When you view hierarchies, you get a global view of an organization and its subsidiaries. You can view and manage multiple types of hierarchies. You can also view the contacts, leads, and opportunities associated with the records in the hierarchy.

Records exist as nodes in the hierarchy. You can use CC360 to view, navigate, and manage the different hierarchy types directly within Salesforce. You can add nodes to a hierarchy, remove nodes from a hierarchy, or drag-and-drop nodes to reorganize a hierarchy. You can move a node from one hierarchy to another, even when the hierarchies are of different hierarchy types.

You can use the Customer 360 view to get a detailed view of an organization and its subsidiaries. You can view the account hierarchy and the details of records that belong to the account. You can filter the records and alphabetically sort the records. You can search for accounts within a hierarchy.

CC360 synchronously processes changes to hierarchies with number of records that is equal to or less than the maximum trigger hierarchy limit plus one when you save your changes. CC360 asynchronously processes changes to hierarchies with number of records that is more than the maximum trigger hierarchy limit plus one in batch jobs.

A node in a hierarchy can have an amount of child records up to 75% of the Data Manipulation Language (DML) Salesforce limit.

The following image shows the hierarchy view in Salesforce Classic:

1. Hierarchy Type
2. Ultimate Parent
3. Record you are viewing on the Account page
4. Customer 360 View button
5. Expand All/Collapse All button

The following image shows the hierarchy view in Lightning Experience:

1. Hierarchy type
2. Ultimate parent
3. Expand/Collapse icon
4. View C360 icon

Note: Hierarchy functionality is compatible with any of the following browsers:

- Google Chrome
- Mozilla Firefox
- Microsoft Internet Explorer 8.0 onwards

Adding Hierarchy Information to the Account Page Layout

Configure the account page layout to provide access to CC360 hierarchy functionality.

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Salesforce Classic, perform the following tasks:
     1. In the upper-right corner of the page, select your name, and click **Setup**.
     2. Under **App Setup**, click **Customize** > **Accounts** > **Page Layouts**.
   - In Lightning Experience, perform the following tasks:
     1. Click the quick access menu ( ), and then click **Setup**.
        The **Setup Home** page appears.
     2. Under **PLATFORM TOOLS**, click **Objects and Fields** > **Object Manager**.
        The **Setup Object Manager** page appears.
     3. Click **Account**.
     4. Under the **Page Layouts** section, select **Account Layout**.

   The **Account Page Layout** page appears.
2. From the **Account Layout** palette, select **Fields** and then drag the **Section** field from the palette to the **Account** page layout.

   The **Section Properties** window appears.
3. Configure the Hierarchy Information section:
   a. Enter `Hierarchy Information` in the **Section Name** text box.
   b. Enable **Detail Page** and **Edit Page**.
   c. Select **1-Column** layout.
   d. Select **Left-Right** tab-key order.
   e. Click **OK**.

   The Hierarchy Information section is added to the Account page.
4. From the Account Layout palette, select **Visualforce Pages** and then drag the Account Hierarchy Visualforce component and the Account Hierarchy Status Message Visualforce component from the palette to the **Hierarchy Information** section.

   You can double-click the Account Hierarchy Visualforce component and the Account Hierarchy Status Message Visualforce component to change the display properties of the component.
5. From the Account Layout palette, select **Fields** and then drag the Ultimate Parent field and the Ultimate Parent Name field from the palette to the **Hierarchy Information** section.
6. Click **Save** from the **Account Layout** palette.

Manually Creating an Account Hierarchy

You can create a hierarchy during and after initial data load. To create a hierarchy during initial data load, assign the appropriate parent account value. You must start with the root node and then load the hierarchy

one level at a time. To create a hierarchy after initial data load, load all the records without parent account values.

Creating During Initial Data Load

CC360 uses the parent account value to maintain the structure of a hierarchy. You can populate these values to build your hierarchy when you use an external tool such as Informatica Cloud Services to initially load the data into CC360.

If you create the hierarchy during the initial data load, start with the top hierarchy level and then one hierarchy level at a time. Load one hierarchy level at a time until you reach the bottom hierarchy level. CC360 determines and then populates the approximate ultimate parent value and hierarchy value.

Creating After Initial Data Load

Load all the records without parent account values. After you load all the data, use an external tool such as Informatica Cloud Services to populate the parent account values.

Use this method to avoid asynchronous hierarchy processing when you create the hierarchy. Start with the root node at the top hierarchy level and then populate the values one hierarchy level at a time.

Creating an Account Hierarchy During Initial Data Load

To create a hierarchy during the initial data load, assign the appropriate parent account value. You must start with the root node and then load the hierarchy one level at time.

CC360 uses the parent account value to maintain the structure of a hierarchy. You can populate these values to build your hierarchy when you use an external tool such as Informatica Cloud Services to initially load the data into CC360. If you create the hierarchy during the initial data load, start with the top hierarchy level and then load one hierarchy level at a time. Load one hierarchy level at a time until you reach the bottom hierarchy level. CC360 determines and then populates the appropriate ultimate parent value and hierarchy value.

Creating an Account Hierarchy During Initial Data Load Example

You want to create a hierarchy with four levels during the initial data load.

The following figure shows the completed hierarchy for the example:

To create the hierarchy, first load hierarchy level 1. Hierarchy level 1 is the top hierarchy level. The top hierarchy consists of the root node. This root node is the ultimate parent for all the other nodes in the hierarchy. The following table shows the parent account value to populate when you load hierarchy level 1.

Node ID	Parent Account value
001	NULL

The following figure shows the hierarchy structure at this point in the process:

After you insert hierarchy level 1, insert the records for hierarchy level 2. The following table shows the parent account value to populate when you load hierarchy level 2.

Node ID	Parent Account value
002	001
003	001
004	001
005	001
006	001
007	001

The following figure shows the hierarchy structure at this point in the process:

After you insert hierarchy level 2, insert the records for hierarchy level 3. The following table shows the parent account value to populate when you load hierarchy level 3.

Node ID	Parent Account value
008	005
009	005
010	005

The following figure shows the hierarchy structure at this point in the process:

After you insert hierarchy level 3, insert the records for hierarchy level 4. The following table shows the parent account value to populate when you load hierarchy level 4.

Node ID	Parent Account value
011	010
012	010

The hierarchy is complete. The following figure shows the completed hierarchy:

Creating an Account Hierarchy After Initial Data Load

To create a hierarchy after initial data load, load all the records without parent account values. After you load all the data, use an external tool such as Informatica Cloud Services to populate the parent account values.

Use this method to avoid asynchronous hierarchy processing when you create the hierarchy. Start with the root node at the top hierarchy level and then populate the values one hierarchy level at a time.

Creating an Account Hierarchy After Initial Data Load Example

You want to create a hierarchy with four levels after the initial data load.

The following figure shows the completed hierarchy for the example:

Load all your records into CC360. Do not populate the parent account values. After you load the records into CC360, the records are not part of a hierarchy and the records do not have relationships.

The following figure shows the lack of hierarchy structure among the records at this point in the process:

After you load the records, use an external tool such as Informatica Cloud Services to populate the parent account values. First update the parent account value for the nodes in the second hierarchy level. You do not need to update the parent account value for the root node because the root node does not have a parent. The parent account value for the root node remains null. The following table shows the parent account values to populate when you update the values for hierarchy level 2:

Node ID	Parent Account value
002	001
003	001
004	001
005	001
006	001
007	001

The following figure shows the hierarchy structure at this point in the process:

After you update the values for hierarchy level 2, update the values for hierarchy level 3. The following table shows the parent account values to populate when you update the values for hierarchy level 3:

Node ID	Parent Account value
008	005
009	005
010	005

The following figure shows the hierarchy structure at this point in the process:

After you update the values for hierarchy level 3, update the values for the root node in hierarchy level 4. The following table shows the parent account value to populate when you update the values for hierarchy level 4:

Node ID	Parent Account value
011	010
012	010

The hierarchy is complete. The following figure shows the completed hierarchy:

Multidimensional Hierarchy Example

You need to create a multidimensional hierarchy that contains a franchise hierarchy within an account hierarchy.

The following figure shows an account hierarchy and a franchise hierarchy:

The franchise hierarchy is a separate hierarchy from the main corporate account hierarchy. However, you want the franchise hierarchy to also appear under the account hierarchy. To do this, clone the root node of the franchise hierarchy to the account hierarchy.

After you clone the franchise hierarchy to the account hierarchy, the records in the franchise hierarchy also appear in the account hierarchy. In this example, you want the root of the franchise hierarchy to appear under the root of the account hierarchy. The following figure shows the structure of the hierarchies after you clone the franchise hierarchy:

Creating a Multidimensional Hierarchy in Salesforce Classic

To create a multidimensional hierarchy, clone the node of a hierarchy of one type to a hierarchy of another type.

1. Edit the target hierarchy that you want to add a hierarchy clone to.
2. In the target hierarchy, select the parent node for the clone hierarchy.
3. Search for the record that you want to add to the hierarchy, and then select the record.
4. Click **Clone**.

   CC360 adds the record to the hierarchy.
5. Click **Save**.

Creating a Multidimensional Hierarchy in Lightning Experience

To create a multidimensional hierarchy, clone the node of a hierarchy of one type to a hierarchy of another type.

1. Edit the target hierarchy that you want to add a hierarchy clone to.

2. Select the parent node for the clone hierarchy and click the **Add** icon.

   The **Add account to the hierarchy** page appears.

3. Search for the record that you want to add to the hierarchy, and then select the record.
4. Click **Clone**.

   CC360 adds the record to the hierarchy.

5. Click **Review**.

   The **Review and Save** page appears.

6. Review the changes, and click **Save**.

Creating a Hierarchy Type

To create a hierarchy type in CC360, change the hierarchy type of the root node to the name of the hierarchy type you want to create.

1. In the hierarchy view, click the root node.

2. From the Bean section on the **Account** page, click **Edit**.

3. In the Hierarchy Informatica section on the **Bean Edit** page, enter the name of the hierarchy type you want to create in the **Hierarchy Type** field.

1. Hierarchy Type field
2. Ultimate Parent indicator

4. Click **Save**.

   The root record and all the children in the hierarchy belong to the hierarchy type that you created.

   Note: If you change the hierarchy type of a record that is not an ultimate parent, CC360 reverts the hierarchy type change.

Enabling Alternate Hierarchy

You must enable alternate hierarchy to view alternate hierarchies in the Hierarchy section of the account page.

To enable alternate hierarchy, set the `Enable Alternate Hierarchy Mode` internal setting value to true. You can disable alternate hierarchy if you set the `Enable Alternate Hierarchy Mode` internal setting value to false. By default, the `Enable Alternate Hierarchy Mode` internal setting value is set to false.
The following sample code enables alternate hierarchy:

```
API_SettingsManager.updateInternalSetting('Enable Alternate Hierarchy Mode', 'true');
```

Asynchronous Hierarchy Processing

CC360 asynchronously processes hierarchy operations for large hierarchies.

Large hierarchies are hierarchies with more nodes than the maximum trigger hierarchy limit. Hierarchy processes for large hierarchies would encounter Salesforce governor limits if CC360 processed hierarchy changes synchronously. To avoid these limits, CC360 uses asynchronous batch jobs to process changes to large hierarchies.

The following batch jobs process hierarchy changes asynchronously:

Asynchronous Hierarchy Process

Processes add, move, and remove operations for large hierarchies.

Asynchronous Cloning Process

Processes clone operations if the number of nodes to clone is larger than the maximum trigger hierarchy limit.

If you abort an asynchronous hierarchy batch job, you could get unexpected hierarchy results. Do not abort a hierarchy batch job.

When you delete a leaf node from a large hierarchy, CC360 processes the delete synchronously. CC360 processes all other changes to large hierarchies asynchronously.

Events that Trigger Hierarchy Processing

Certain operations can trigger hierarchy processing. Consider these operations to anticipate the launch of asynchronous hierarchy batch processes for large hierarchies.

The following table describes the changes that trigger processing for hierarchies:

Column Name	Affected Hierarchy	Description
Name	account and external	Company name changed for the ultimate parent node.
ParentID	account and external	Parent ID changed for any node in the hierarchy.
DSE__DS_Ultimate_Parent__c	account and external	Ultimate Parent ID changed for any node in the hierarchy.
DSE__DS_Hierarchy_Level__c	account and external	Hierarchy level changed for any node in the hierarchy.
DSE__DS_Ultimate_Parent_Name__c	account and external	Ultimate Parent Name changed for any node in the hierarchy.
DSE__DS_Parent__c	external	External Bean Parent ID changed for any node in the hierarchy.
DSE__DS_Parent_Id__c	external	External Bean Parent External ID changed for any node in the hierarchy.
DSE__DS_Hierarchy_Type__c	external	Hierarchy Type changed for any node in the hierarchy.
DS_Ultimate_Parent_ID__c	external	Ultimate Parent External ID changed for any node in the hierarchy.
DSE__DS_Is_Ultimate_Parent__c	external	Is Ultimate Parent state changed for any node in the hierarchy.

A change to non-hierarchy fields such as the billing city, billing country, or phone do not trigger hierarchy processing unless you modify the hierarchy fields and non-hierarchy fields in the same operation.

Monitoring Asynchronous Hierarchy Processing

To monitor asynchronous hierarchy processing, you can compare the total job items and the total items processed. You can view the items processed from the Job Logs tab or from the CC360 batch manager.

1. To monitor asynchronous hierarchy processing from the Job Logs:
   a. Select the **Job Logs** tab.
   b. Select the batch job from the jobs list.

c. Compare the total job items to the total items processed. If the hierarchy asynchronous batch job has not processed all the total job items, launch additional hierarchy asynchronous batch jobs as batch job slots become available.

The following image shows a hierarchy edit with 42,476 items to process completes in nine batch jobs:

Batch Queue
AsynchronousHierarchy

Batch Queue Detail

Batch Name	AsynchronousHierarchy	Owner	Vijay S [Change]
Concurrent Set	0	Loop Number	9
Job Class Name	DS_AsynchronousHMBatch	Run Number	1
Param 6	Ultimate Parent Name Changed	Job Phase	finish
Total Records	42476	Platform Job Id	707000000GBdi0
Processed Records Count	42,476	Status	Completed

Job Logs Job Logs Help

Action	Log Number	Job Description	Job Status	Job Type	Start Time	End Time	Total Items Processed	Total Job Items
Edit | Del	J-000525	Asynchronous Hierarchy Process	Completed	AsynchronousHierarchy	9/29/2013 6:13 AM	9/29/2013 6:21 AM	42,476	42,476
Edit | Del	J-000526	OPR=Ultimate Parent Name Changed, [redacted] Demo Data - 42.4K	Finished	AsynchronousHierarchy	9/29/2013 6:13 AM	9/29/2013 6:14 AM	5,000	42,476
Edit | Del	J-000527	OPR=Ultimate Parent Name Changed, [redacted] Demo Data - 42.4K	Finished	AsynchronousHierarchy	9/29/2013 6:14 AM	9/29/2013 6:15 AM	5,000	42,476
Edit | Del	J-000528	OPR=Ultimate Parent Name Changed, [redacted] Demo Data - 42.4K	Finished	AsynchronousHierarchy	9/29/2013 6:15 AM	9/29/2013 6:16 AM	5,000	42,476
Edit | Del	J-000529	OPR=Ultimate Parent Name Changed, [redacted] Demo Data - 42.4K	Finished	AsynchronousHierarchy	9/29/2013 6:16 AM	9/29/2013 6:17 AM	5,000	42,476
Edit | Del	J-000530	OPR=Ultimate Parent Name Changed, [redacted] Demo Data - 42.4K	Finished	AsynchronousHierarchy	9/29/2013 6:17 AM	9/29/2013 6:18 AM	5,000	42,476
Edit | Del	J-000531	OPR=Ultimate Parent Name Changed, [redacted] Demo Data - 42.4K	Finished	AsynchronousHierarchy	9/29/2013 6:18 AM	9/29/2013 6:19 AM	5,000	42,476
Edit | Del	J-000532	OPR=Ultimate Parent Name Changed, [redacted] Demo Data - 42.4K	Finished	AsynchronousHierarchy	9/29/2013 6:19 AM	9/29/2013 6:20 AM	5,000	42,476
Edit | Del	J-000533	OPR=Ultimate Parent Name Changed, [redacted] Demo Data - 42.4K	Finished	AsynchronousHierarchy	9/29/2013 6:20 AM	9/29/2013 6:20 AM	5,000	42,476
Edit | Del	J-000534	OPR=Ultimate Parent Name Changed, [redacted] Demo Data - 42.4K	Finished	AsynchronousHierarchy	9/29/2013 6:20 AM	9/29/2013 6:21 AM	2,476	42,476

2. To monitor asynchronous hierarchy processing from the CC360 batch manager:
   a. Select the **CC360 Admin** tab.
   b. Monitor the number of items processed in the Completed Jobs section for the asynchronous hierarchy processes. If necessary, launch additional hierarchy asynchronous batch jobs as batch job slots become available.

      A Running Jobs section appears for batch jobs that are running.

Configuring Hierarchy Edit Search Fields

To configure the search fields that are available when you search for a record to add to a hierarchy, configure the hierarchy edit search field set.

1. From the **Setup** section of the **CC360 Admin** tab, click **Run the Setup Wizard**.
2. Click **Field Set**.

   The **Field Set Settings** page appears.
3. In the **Account Field Sets** section, click **Edit Account Field Sets**.

   The **Account Field Sets** page appears.
4. From the **Field Sets** section, select **Edit** for the **Hierarchy Edit Search Fields**.
5. Drag the appropriate fields from the Hierarchy Edit Search Fields palette to the field set list.
6. Click **Save**.

Configuring Hierarchy Edit Search Results Fields

To configure the search result fields that appear when CC360 returns a list of records to add to a hierarchy, configure the hierarchy edit search results field set.

1. From the **Setup** section of the **CC360 Admin** tab, click **Run the Setup Wizard**.
2. Click **Field Set**.

   The **Field Set Settings** page appears.
3. In the **Bean Field Sets** section, click **Edit Bean Field Sets**.

   The **Bean Field Sets** page appears.
4. From the **Field Sets** section, select **Edit** for the **Hierarchy Edit Search Results Fields**.
5. Drag the appropriate fields from the Hierarchy Edit Search Results Fields palette to the field set list.
6. Click **Save**.

Configuring the Hierarchy Processing Parameters

To configure the hierarchy processing parameters edit the CC360 settings and performance settings.

1. From the **Setup** section of the **CC360 Admin** tab, click **Run the Setup Wizard**.
2. Select **Manage** for **CC360 Settings**.

   The **CC360 Settings**CC360 page appears.
3. Select **Edit** for the **Active** setting profile.
4. Enter a value for the hierarchy processing parameter you want to configure.

   You can configure the hierarchy batch size limit, the maximum hierarchy limit, the hierarchy scout node page size, and the maximum trigger hierarchy limit.
5. Click **Save**.
6. Repeat step 1.
7. Select **Manage** for **Performance Settings**.

   The **Performance Settings** page appears.
8. Select **Edit** for the asynchronous hierarchy batch setting profile.
9. To configure the batch size, enter a value in the **Batch Size** field.
10. Click **Save**.

Configuring the Customer 360 View

Use the setup wizard to configure the Customer 360. You can configure the objects that you can use in the **Customer 360** page. You can add standard objects and custom objects. Default object in Customer 360 is Contacts. You can set the default object by changing the display order.

1. From the **Setup** section of the **CC360 Admin** tab, click **Run the Setup Wizard**.

2. Click **Customer 360 Settings**.
3. In the **Related Object Setup** section, perform one of the following tasks:
   - Click **New** to add an object.
   - Click **Edit Contact**, **Edit Lead**, or **Edit Opportunity** to edit the configuration.

   The **Customer 360 Settings** page appears.
4. Configure the following components:

   Object Information

   Indicates the object that you want to add or configure. The Account Lookup field selects an account relationship field associated with the selected object.

   View Filter Information

   Indicates a filter logic in the filter criteria.

   For example, `MailingCountry='United States'` is a filter logic that searches for records with United States as MailingCountry.

   You can validate the filter logic. To validate the filter logic, click **Validate**.

   Display Column Information

   Indicates the fields that you can view in the Customer 360 page. You can use the predefined field sets or you can select the fields that you want to view in the Customer 360 page.

   Display Record Information

   Indicates the number of records that you want to display on the Customer 360 page.

   Visibility Information

   Indicates the custom permission that you want to assign to the object. You can create custom permissions and assign it to a user or to a profile. A user can view the object only if the user has the permission to access the object.
5. Click **Save**.

Changing the Display Order of Objects

You can change the display order of the objects that you view in the Customer 360 page. The object that you move to the top functions as the default object. The Customer 360 page displays the records based on the selected object.

To change the display order of the objects, perform the following tasks:

1. From the **Setup** section of the **CC360 Admin** tab, click **Run the Setup Wizard**.
2. Click **Customer 360 Settings**.

   The **Customer 360 Settings** page appears.
3. Click **Change Display Order**.
4. Click the up or down arrow to move the objects.

   The object that you move to the top functions as the default object.
5. Click **Save**.

Synchronize Reporting Data Batch Job

You can update the hierarchy information for all records in an account hierarchy when account hierarchy enrichment is disabled. Run the synchronize reporting data batch job to update the hierarchy information.

You must be a system administrator to run the synchronize reporting data batch job to ensure that you have sufficient privileges to update all the account records in the hierarchy.

Before you run the synchronize reporting data batch job, ensure that the asynchronous hierarchy process batch job and asynchronous cloning process batch job are not running. If you run the synchronize reporting data batch job when the asynchronous hierarchy process batch job and asynchronous cloning process batch job are running, you might encounter locking errors.

You can run the synchronize reporting data batch job in the Jobs section of the **CC360 Admin** tab. For more information about running, scheduling, and managing batch job, see the *Managing Batch Jobs* chapter.

Troubleshooting Hierarchies

If you encounter any issues when you configure hierarchies, use the following information to troubleshoot the issues.

When I edited hierarchy information for a record, I received an error message indicating that I do not have sufficient access rights.

When you update the hierarchy information of a child account, such as the parent account field, CC360 updates the hierarchy information for the parent account record. If you do not have sufficient privileges to edit the parent account record, you receive an error message.

If you receive an error message that indicates you do not have sufficient access rights, you need to disable account hierarchy enrichment. To disable account hierarchy enrichment, use the API_SettingsManager.enableHierarchyAccountEnrichment(false) API to set the DS_SETTING_HIERARCHY_ACCOUNT_ENRICHMENT internal CC360 parameter to `false`. DS_SETTING_HIERARCHY_ACCOUNT_ENRICHMENT is `true` by default.

If you update hierarchy information for a record when DS_SETTING_HIERARCHY_ACCOUNT_ENRICHMENT is `false`, CC360 only updates the records in the hierarchy for which you have sufficient privileges. To update hierarchy information for all records in the hierarchy, a system administrator must run the Synchronize Reporting Data batch job. Ensure that the Asynchronous Hierarchy Process batch job and Asynchronous Cloning Process batch job are not running when the Synchronize Reporting Data batch job runs.

CHAPTER 10

Integrating Cloud MDM - Customer 360 for Salesforce with Multidomain MDM

This chapter includes the following topics:

- Integrating Cloud MDM - Customer 360 for Salesforce with Multidomain MDM Overview, 105
- Step 1. Setting Up Informatica Cloud Application Integration, 106
- Step 2. Configuring Remote Site Settings, 111
- Step 3. Creating a Search Connection, 112

Integrating Cloud MDM - Customer 360 for Salesforce with Multidomain MDM Overview

You can integrate Cloud MDM - Customer 360 for Salesforce (CC360) with Multidomain MDM and search for matching or duplicate records in the MDM Hub data sources. If any duplicate or matching records exist, you can import the MDM Hub records into CC360. CC360 creates an external bean for each imported record.

To search for matching or duplicate records in MDM Hub, you must configure a search connection. A search connection defines settings required to connect CC360 with MDM Hub. You can use Informatica Cloud® Real Time as a connector between CC360 and the MDM Hub.

To integrate CC360 with MDM Hub, you must perform the following tasks:

1. Set up Informatica Cloud Application Integration. Not required if the Multidomain MDM is deployed on Amazon Web Services (AWS).
2. Configure remote site settings.
3. Create a search connection.

After you integrate CC360 with Multidomain MDM, you can use the Lightning Experience record creation page to search for duplicate records in the MDM Hub data sources. For more information about searching for duplicate records, see the *Informatica Cloud MDM - Customer 360 for Salesforce User Guide*.

Step 1. Setting Up Informatica Cloud Application Integration

Use Informatica Cloud Application Integration to connect CC360 with Multidomain MDM. Informatica Cloud Application Integration acts as a connector between the MDM Hub and CC360. Configure Informatica Cloud Application Integration and publish REST endpoints to access metadata of business entity view and to search records in the MDM Hub.

Note: If Multidomain MDM is deployed on Amazon Web Services (AWS), you do not require to set up Informatica Cloud Application Integration.

Use Informatica Cloud Application Integration to perform the following tasks:

1. Create a service connector.
2. Create a connection.
3. Create processes.

For more information about service connectors, connections, and processes, see the Informatica Cloud Application Integration help.

Step 1. Create a Service Connector

Use the Process Designer to create a service connector that specifies parameters to connect to services and data sources.

1. Log in to Informatica Intelligent Cloud Services, and select **Application Integration**.
2. Click **New**. The **New Asset** dialog box appears.
3. Click **Service Connectors**, and then click **Create**.
4. On the **Definition** tab, define the following basic properties for the service connector.
   - Name by which the service connector is made available for processes.
   - Project or folder to contain the service connector.
   - Description for the service connector.
   - Indicate whether the service connector must only be available to run on the Secure Agent.
   - Properties required to connect to this service.

The following image shows the **Definition** tab:

5. On the **Actions** tab, define the actions associated with the service connector.

   For more information about actions, see the Informatica Cloud Application Integration help.

6. On the **Input** tab, define the input variables that are unique to the service to which you are sending data.

   For more information about input variables, see the Informatica Cloud Application Integration help.

   The following image shows the **Input** tab:

7. On the **Binding** tab, define the following properties for each action in the service connector:

   - URL. URL of the Get Metadata or Search Business Entity REST web service.
     The Get Metadata URL has the following format: `http://<host>:<port>/<context>/<database ID>/<business entity>.json?action=meta`

     The following sample URL gets the metadata of a business entity or business entity view in the MDM Hub: `http://10.65.112.181:8080/cmx/cs/orcl-CMX_ORS/SA_Emp.json?action=meta`

     The Search Business Entity URL has the following format: `http://<host>:<port>/<context>/<database ID>/<business entity>.json?fq=<business entity field name>='<business entity field value>'`

     Note: In the Search Business Entity URL, you can use only the MDM Hub Smart Search fq query parameter and the AND logical operator.

The following sample URL searches for records that have the last name as Joe: `http://10.65.112.181:8080/cmx/cs/orcl-CMX_ORS/SA_Emp.json?fq=lastName=joe`

- Verb. Method to run the REST web service.
- Multi Using. Determines how query string parameters are generated when you need to specify multiple values for a parameter.
- Authentication. Authentication credentials to access the MDM Hub data sources.
- Other Parameters. Input parameters required by the service.
- HTTP Headers. If the service requires one or more HTTP headers, enter the name and the value of the parameter for each header.

The following image shows the **Binding** tab:

8. On the **Output** tab, define how the service connector parses data returned from a service and places it into variables. Specify the following properties for each output data item:
   - Name. Name of a variable into which a returned value is placed.
   - Type. Data type of the value written to the variable.
   - Description. Description of the variable.
   - Get From. Select one of the following:
     - Property. To enter a named value (which is the name used within the XML returned by the service) to be placed within the variable.
     - Expression. To parse the MDM Hub JSON response to the output variable.

Use the following sample code to parse the response:

```
{
$RESTResponse/rest:metadata/text()
}
```

- XQuery. To write an XSLT expression to parse the XML returned by the service.
- HTTP Response Header. To enter the part of the response header to assign to the field.
- HTTP Response Status Code. To check the HTTP response status code.
- Entire Response. To assign the complete contents of the response payload to the field.
- Simplified XML. To rearrange data that can be used by process objects.

The following image shows the **Output** tab:

For more information about the **Output** tab, see the Informatica Cloud Application Integration help.

9. Save the service connector.
10. On the **Test Results** tab, select a test server, and click **Test**.

    The test result appears.

Step 2. Create a Connection

A connection is used by processes to access the service. You can specify the values for the parameters in a connection to be used in your process.

1. In Application Integration, click **New**. The **New Asset** dialog box appears.
2. Click **Connections** > **Create**.

The following image shows the **Properties** tab:

3. On the **Properties** tab, enter the following connection details:
   - Name. Name of the connection.
   - Location. Location of the connection.
   - Description. Description of the connection.
   - Type. The service connector that you created.
   - Run On. The machine on which your secure agent resides.

   For more information about connection properties, see the Informatica Cloud Application Integration help.
4. Enter the following connection properties:
   - Name. Name of the property.
   - Value. IP address of the machine that runs Multidomain MDM.
   - Description. Description of the property.
5. Save and publish the connection.

Step 3. Create Processes

A process collects data from fields in the process and sends the output from its actions to other fields.

You must create the following two processes for each search service and meta service:

- Process. Process that runs on the Cloud Server.
- Subprocess. Process that runs on a Secure Agent.

1. In Application Integration, click **New**. The **New Asset** dialog box appears.
2. Click **Processes** > **Create**.
3. In the **Properties** panel, enter the required information in the following sections:

- General
- Start
- Input Fields
- Output Fields
- Temp Fields
- Messages
- Advanced
- Notes

For more information about the properties, see the Informatica Cloud Application Integration help.

4. In the **Design** panel, add the process steps and define the properties.

   You can add **Service** or **Assignment** steps. For more information about adding steps, see the Informatica Cloud Application Integration help.

5. Save and publish the process.
6. On the **My Processes** page, and click the process that you created.

   The properties of the process appears at the bottom of the screen. Use the Service URL to create a search connection in CC360.

Step 2. Configuring Remote Site Settings

After you publish the parent processes in Informatica Cloud Application Integration, you must register the service URL. To register the Service URL, configure the Salesforce remote site settings.

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Salesforce Classic, perform the following tasks:
     1. In the upper-right corner of the page, select your name, and click **Setup**.
     2. Under **Administration Setup**, click **Security Controls** > **Remote Site Settings**.
   - In Lightning Experience, perform the following tasks:
     1. Click the quick access menu ( ), and then click **Setup**.
        The **Setup Home** page appears.
     2. Under **SETTINGS**, click **Security** > **Remote Site Settings**.

   The **All Remote Site** page appears.
2. Click **New Remote Site**.

   The **Remote Site Edit** page appears.
3. Enter a name for the remote site.
4. Perform one of the following tasks based on how you deploy Multidomain MDM:
   - If the Multidomain MDM is deployed on AWS, enter the cloud server as the remote site URL.
     For example, enter `https://mdm-nonprod.com/` as the URL.

- If the Multidomain MDM is deployed locally, enter the service URL that you published in Informatica Cloud Application Integration as the remote site URL.
  For example, enter `https://ps1w2.rt.informaticacloud.com/` as the URL.

5. To allow access to the remote site regardless of whether the user's connection is over HTTP or HTTPS, select **Disable Protocol Security** .

   Select the option only if you understand the security implications.
6. Optionally, enter a description of the site.
7. Click **Save**.

Step 3. Creating a Search Connection

A search connection acts as a channel through which you can search for data or import data from MDM Hub. The search connection uses the search parameters to create a connection between CC360 and MDM Hub.

1. From the **Setup** section of the **CC360 Admin** tab, click **Run the Setup Wizard**.
2. In the **Service Setup** section, click **Multidomain MDM** .

   The **Service: Informatica Multidomain MDM** page appears.
3. To create a search connection, click **New Search Connection**.

   You can also edit an existing search connection.
4. In the **Search Service Information** section, perform the following tasks:
   a. Activate the search connection.
   b. Configure the following connection details:

      Name

      Name of the search connection.

      User ID and Password

      Authentication credentials to access Informatica Cloud.

      Business Entity/View

      Business entity type of the data in Multidomain MDM business entity or business entity views.

      Object Class

      Object class used in the matching settings to compare Multidomain MDM and CC360 records.

      Metadata URL

      URL to access the metadata of the MDM Hub.

      Use one of the following values based on how you deploy the MDM Hub:

      - If the Multidomain MDM is deployed on AWS, use URL of the Get Metadata REST web service.
        The following URL is an example service URL that gets the metadata of a business entity view in the MDM Hub:

        ```
        http://mdm-nonprod.informaticacloud.com/suenonprod/cmx/cs/orcl-CMX_ORS/
        SA_Emp.json?action=meta
        ```

- If the Multidomain MDM is deployed locally, use the service URL from the meta service parent process you published in Informatica Cloud Application Integration.
  The following URL is an example service URL that accesses the metadata of a business entity:

  ```
  https://ps1w2.rt.informaticacloud.com/active-bpel/rt/MDMDev003-Hybrid-ProcessCloud-Person-Meta
  ```

Search URL

URL to search for records in the MDM Hub.

Use one of the following values based on how you deploy the MDM Hub:

- If the Multidomain MDM is deployed on AWS, use the URL of the Search REST web service. The following URL is an example service URL to search for records in the MDM Hub data sources:

  ```
  http://mdm-nonprod.informaticacloud.com/suenonprod/cmx/cs/orcl-CMX_ORS/SA_Emp.json?fq=
  ```

  In the example, `fq` is a query parameter of the Search REST web service.

- If the Multidomain MDM is deployed locally, use the service URL from the service parent process that you published in Informatica Cloud Application Integration.
  The following URL is an example service URL to search for records in the MDM Hub data sources:

  ```
  https://ps1w2.rt.informaticacloud.com/active-bpel/rt/MDMDev003-Hybrid-ProcessCloud-Person-Search?icrtInput=
  ```

  In the example URL `icrtInput` is the input variable defined in the parent process that you created in Informatica Cloud Application Integration.

5. Click **Get Fields**. Gets the business entity service fields from Multidomain MDM.
6. In the **Field Mapping and Priority Information**, perform the following tasks:
   a. Map the bean fields to the business entity service fields.
   b. Enable search for the fields.
   c. Assign priorities for each field.

   Note: If you map date or time fields, ensure that you use the following format:

   For Date field. `<yyyy>-<mm>-<dd>T<hh>:<mm>:<ss>Z`

   For DateTime field. `<yyyy>-<mm>-<dd> <hh>:<mm>:<ss>Z`

   For example, `2014-12-26 10:10:10Z`
7. Click **Save**.

   Note: After you save the search connection, CC360 uses the field mapping and the priority information to create synchronization and data source settings for the Multidomain MDM business entity.

CHAPTER 11

Managing Multi-Org

This chapter includes the following topics:

- Managing Multi-Org Overview, 114
- Real-Time Synchronization in a Multi-Org Implementation, 115
- Synchronizing Accounts in a Multi-Org Implementation, 116
- Informatica Cloud Services Workflow, 116
- Real-Time and Batch Processing, 120
- Enrichment Settings on the Hub Org, 120
- Account Fields in the Spoke Org, 120
- Authentication Parameters to Connect to the Master Org, 121
- Configuring the Master Org, 121
- Configuring the Spoke Org, 123
- Getting Updates from the Master Org in Real-Time , 126
- Getting Updates from the Master Org in Batch Mode, 127

Managing Multi-Org Overview

Use the Multi-Org capability to automate the cleansing, standardization, deduplication, consolidation, data conversion, contact data verification, and master data management of data from multiple Orgs in a distributed Salesforce environment. You can perform real-time and batch synchronization of accounts and person accounts in a multi-org implementation.

In a Multi-Org implementation, you must identify one Salesforce Org as a master and the other related Orgs as connected Orgs. The Master Org is the Org on which you install CC360 and perform CC360 processes such as, cleansing, standardization, and deduplication to manage master data. The other connected Orgs are the Spoke Orgs on which you install a thin client for communication control. To get updated master data for accounts or person accounts from the Master Org, a Spoke Org uses the REST service of the Master Org to query the Master Org. Spoke Orgs can synchronize with the Master Org in real-time for a single account or by running a batch for all accounts. Before you implement the Multi-Org capability, create a workflow to migrate data from the Spoke Orgs to the Master Org. You can create this workflow by using Informatica Cloud Services or any other data integration tool.

Note: Multi-Org implementation supports synchronization of accounts and person accounts only.

Real-Time Synchronization in a Multi-Org Implementation

In a Multi-Org implementation, Spoke Orgs use the REST service of the Master Org to get an updated account or person account from the Master Org.

The following image shows the real-time synchronization of an account between the Master Org and a Spoke

Org:

The following process is a high-level overview of the real-time synchronization of an account between the Master Org and a Spoke Org:

Workflow creation using Informatica Cloud Services

Workflow creation using Informatica Cloud Services data integration tool for migration of accounts or person accounts from the Spoke Org to the Master Org. The workflow creates external beans on the Master Org.

Matching and consolidation check on the Master Org

Matching and consolidation check on the Master Org that creates master beans and convert master beans to accounts.

Real-time synchronization

Real-time synchronization during which a user opens an account in the Spoke Org, the Spoke Org uses the REST service to query the Master Org to get fields updated in the Master Org. The user can view the list of fields that are updated in the Master Org. The user can choose to apply or ignore the updates from the Master Org.

Synchronizing Accounts in a Multi-Org Implementation

In a Multi-Org implementation, Spoke Orgs use the REST service of the Master Org to get updated accounts or person accounts from the Master Org.

The following image shows the batch synchronization of accounts between the Master Org and a Spoke Org:

The following process is a high-level overview of the batch synchronization of accounts between the Master Org and a Spoke Org:

Workflow creation using Informatica Cloud Services

Workflow creation using Informatica Cloud Services data integration tool for migration of accounts or person accounts from the Spoke Org to the Master Org. The workflow creates external beans on the Master Org.

Matching and consolidation

Matching and consolidation check on the Master Org that creates master beans and convert master beans to accounts.

Batch job synchronization

Batch job synchronization during which the batch job from the Spoke Org uses the REST service to synchronize all records in the Spoke Org with the Master Org.

Informatica Cloud Services Workflow

You need to create a workflow to migrate data from the Spoke Orgs to the Hub Org. You can create this workflow by using Informatica Cloud Services or any other data integration tool.

The following image shows a sample mapping created in Informatica Cloud Services to migrate accounts from a Spoke Org to the Hub Org:

Target: DSE__DS_Bean__c

Status	Name	Expression/Lookup
	Record ID	
	Owner ID	
	Deleted	
	Bean No	
	Created Date	
	Created By ID	
	Last Modified Date	
	Last Modified By ID	
	System Modstamp	
	Custom Index	
	Account Deleted	
	Account	
	Assign existing Contact to C...	
	Assign existing Lead to Camp...	
✔	City	BillingCity
✔	Country	BillingCountry
✔	Postal Code	BillingPostalCode
✔	State	BillingState
✔	Street	BillingStreet
	Birthdate	
	Campaign	
	Check Hierarchy	
	Cleansed	
	Company Norm	
✔	Company	Name
	Consolidated	
	Contact Deleted	
	Contact	
	Conversion Status	

The following image shows the continuation of the sample mapping to migrate accounts from a Spoke Org to the Hub Org:

Convert after Consolidation	
Convert to Account	
Convert to Contact	
Convert to Lead	
Convert to Person Account	
Convert without Duplicate Check	
Country ISO Code	
Custom Field 10	
Custom Field 1	
Custom Field 2	
Custom Field 3	
Custom Field 4	
Custom Field 5	
Custom Field 6	
Custom Field 7	
Custom Field 8	
Custom Field 9	
Deleted	
Description	
Domain	
Duplicate Checked	
Email	
External ID	CONCAT('SFDCTransport#', Id)
First Name	
Hierarchy Level	
Hierarchy Type	
Internal Comment	
Is Ultimate Parent	
Last Duplicate Check	
Last Name	
Last Updated	
Lead Deleted	

The following image shows the continuation of the sample mapping to migrate accounts from a Spoke Org to the Hub Org:

	Lead	
	Legal Form	
	Master Account Name	
	Master Account	
	Master Bean Account Name	
	Master Bean Account	
	Master Bean	
	Override Master Bean	
	Mobile Phone	
	Object Class	
	Object Deleted	
	Parent External ID	
	Parent	
✔	Phone	Phone
	Prefix	
	Record Type Name	
	Score	
	Segment Key	
	Segment	
✔	Source No	Id
✔	Source	'SFDCTransport'
	Ultimate Parent ID	
	Ultimate Parent Name	
	Ultimate Parent	
	Updated	
✔	Website	Website
	XO_Obsolete_1	
	XO_Obsolete_2	
	Convert to Record Type	
	Converted from Bean	
	Converted from Master Bean	
	Recovery Info	
	Internal Ultimate Parent	
	X_Internal	
	XO_Obsolete_3	
	X_Reset_Actions	

Real-Time and Batch Processing

Spoke Orgs can get updated account details from the Master Org in real-time or by running a batch job.

Real-Time

When a user opens an account in the Spoke Org, the Spoke Org uses the REST service to check for updates to fields that are specified as part of enrichment settings in the Master Org. The list of fields that are updated in the Master Org after the previous synchronization appear on the page. The Spoke Org stores the previous synchronization timestamp in the `MultiOrg Last Update From Hub` field. The user can choose to apply or ignore the updates received from the Master Org.

Batch

To synchronize all records in a Spoke Org with the Master Org, you need to run a batch job from the Spoke Org. When you run the batch job, it uses the REST service to check for updates to fields that are specified as part of enrichment settings in the Master Org. The batch job gets updated field values for all records in the Spoke Org and performs field-level updates for each record. The batch job does not consider the value in the `MultiOrg Last Update From Hub` field. Instead, the batch job uses the batch number to track updates from the Master Org. After a batch job is complete, all records processed by the batch job must have the same batch number, which indicates all records in the Spoke Org are in sync with Master Org. The batch number is not updated for records that failed. The subsequent batch job picks up the failed records and if the records are processed successfully, the batch number is updated accordingly. The failed records are tracked by using Last Processed ID in the custom settings of the Spoke Org.

Note: Different batch numbers are maintained for synchronization of accounts and person accounts.

Enrichment Settings on the Hub Org

In a Multi-Org implementation, when an account is updated in the Master Org, the Master Org sends the updated account details to Spoke Orgs. However, the Master Org sends only the fields that you identify as part of enrichment settings to the Spoke Orgs. For example, you identify Billing Country, Billing City, Shipping Country, and Shipping City as part of enrichments settings. If any of these field values change in the Master Org, Spoke Orgs can get the updated field values in real-time or by running a batch job.

Account Fields in the Spoke Org

In addition to the standard fields, additional fields are added to the account object in the Spoke Org.

When you install the unmanaged package on a Spoke Org, the package adds the following fields to the account object:

MultiOrg Last Update From Master

Stores the timestamp of the last update from the Master Org to an account or person account in the Spoke Org. For example, this field shows that the last update from the Master Org is 3.00 P.M and a user opens an account at 3.45 P.M in the Spoke Org. The list of fields that are updated in the Master Org after the 3.00 P.M for that account appear on the page. The value for this field changes from 3.00 P.M to 3.45 P.M.

MultiOrg Batch Run Number

Stores the run number of a batch job. Separate batch jobs are available for synchronizing accounts and person accounts with the Master Org.

Authentication Parameters to Connect to the Master Org

When you create a Connected App on the Master Org, Cloud MDM generates the consumer key and consumer secret authentication parameters. To enable a Spoke Org to connect to the Master Org, you need to specify the consumer secret and consumer key as part of the custom setting on each Spoke Org.

The consumer key and consumer secret are explained as follows:

Consumer Key

A Spoke Org uses this value to identify with Salesforce.

Consumer Secret

A Spoke Org uses this secret to establish ownership of the consumer key.

Configuring the Master Org

In a Multi-Org implementation, the Master Org is the central org on which you install CC360.

To configure a Master Org, perform the following steps:

1. Create a data source for each Spoke Org that you want to integrate.
2. Maintain account enrichment settings for each Spoke Org.
3. Create a connected application for each Spoke Org that you want to integrate.
4. Set a value for the `setMasterDataResponseSize` internal setting.

Create a Data Source on the Master Org

In the Master Org, you must create a data source for each Spoke Org that you want to integrate with the Master Org.

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Salesforce Classic, perform the following tasks:
     1. In the upper-right corner of the page, select your name, and click **Setup**.
     2. Under **App Setup**, click **Develop** > **Custom Settings**.
   - In Lightning Experience, perform the following tasks:
     1. Click the quick access menu (), and then click **Setup**.
        The **Setup Home** page appears.
     2. Under **PLATFORM TOOLS**, click **Custom Code** > **Custom Settings**.

   The **Custom Settings** page appears.

2. Click **Manage** for **Data Sources**.
3. Click **New**.
4. Specify values for all relevant fields.
5. Click **Save**.

Maintain Account Enrichment Settings on the Master Org

In a Multi-Org implementation, when an account is updated in the Master Org, the Master Org sends the updated account details to Spoke Orgs. However, the Master Org sends only the fields that you identify as part of enrichment settings to the Spoke Orgs. For example, you identify Billing Country, Billing City, Shipping Country, and Shipping City as part of enrichments settings. If any of these field values change in the Master Org, Spoke Orgs can get the updated field values in real-time or by running a batch job.

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Salesforce Classic, perform the following tasks:
     1. In the upper-right corner of the page, select your name, and click **Setup**.
     2. Under **App Setup**, click **Develop** > **Custom Settings**.
   - In Lightning Experience, perform the following tasks:
     1. Click the quick access menu (), and then click **Setup**.
        The **Setup Home** page appears.
     2. Under **PLATFORM TOOLS**, click **Custom Code** > **Custom Settings**.

   The **Custom Settings** page appears.
2. Click **Manage** for **Enrichment Settings**.
3. Click **New**.
4. In the **Name** field, enter a name that conforms to the `sObject.SpokeOrgName` naming convention. For example, `Account.SpokeOrg1`.

   Note: The `SpokeOrgName` must match the data source name that you specified while creating a data source for the Spoke Org.
5. Specify the fields that you want to enrich.
6. Click **Save**.

Create a Connected App on the Master Org

In the Master Org, you must create a Connected App for each Spoke Org. Salesforce Connected Apps use a protocol `OAuth2` to authenticate with Salesforce. When Connected App users request access to a Salesforce Org, Salesforce passes a token to the Connected App if the authentication is successful.

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Salesforce Classic, perform the following tasks:
     1. In the upper-right corner of the page, select your name, and click **Setup**.
     2. Under **App Setup**, click **Create** > **Apps**
        The **Apps** page appears.

- In Lightning Experience, perform the following tasks:
    1. Click the quick access menu ( ), and then click **Setup**.
       The **Setup Home** page appears.
    2. Under **PLATFORM TOOLS**, select **Apps** > **Connected Apps**.

2. Under **Connected Apps**, click **New**.
3. Under **Basic Information**, enter a name for the application and an email ID.
4. In the **Callback URL** field, enter the following URL:

   `https://login.salesforce.com/services/oauth2/success`

5. Under **API (Enable OAuth Settings)**, select **Enable OAuth Settings**.
6. Under **Available OAuth Scopes**, select all the scopes and click **Add**.

   All the scopes appear under **Selected OAuth Scopes**.

7. Click **Save**.

The consumer secret and the consumer key appear under the **API (Enable OAuth Settings)** section. When you configure the custom setting on the Spoke Org, specify the consumer key and the consumer secret. A Spoke Org uses the consumer key value to identify with Salesforce and uses the consumer secret to establish ownership of the consumer key.

Limit the Number of Responses

Use the `setMasterDataResponseSize` internal setting to limit the number of responses from the Master Org to Spoke Orgs.

Use the `API_SettingsManager` API to set the value for the internal setting. The following sample code sets the value of the internal setting to 200:

```
API_SettingsManager.setMasterDataResponseSize(200);
```

Note: The default value of the `setMasterDataResponseSize` internal setting is 200. If you set the value of this setting to more than 200, the batch might report errors and not complete successfully.

Configuring the Spoke Org

In a Multi-Org implementation, a Spoke Org connects to the Master Org to get updated accounts and person accounts.

To configure a Spoke Org, perform the following steps:

1. Install the unmanaged package on the Spoke Org.
2. Configure the custom setting to specify credentials to connect to the Master Org.
3. Verify remote site settings.
4. Add the Visualforce page for real-time update.

Install the Unmanaged Package on the Spoke Org

To enable a Spoke Org to connect to the Master Org, install the unmanaged package for Spoke Org on each Spoke Org.

1. Click the unmanaged package installation link obtained from Informatica.
2. Follow the instructions in the wizard to complete the installation of the unmanaged package on the Spoke Org.

Configure Connection to the Master Org

On each Spoke Org, configure the custom setting and specify the credentials that enable the Spoke Org to connect to the Master Org.

1. Perform one of the following tasks based on the Salesforce environment that you use:
   - In Salesforce Classic, perform the following tasks:
     1. Select your name and click **Setup**.
     2. Under **App Setup**, select **Develop** > **Custom Settings**.
   - In Lightning Experience, perform the following tasks:
     1. Click the quick access menu ( ) and then **Setup Home**.
        The **Setup Home** page appears.
     2. Under **PLATFORM TOOLS**, select **Custom Code** > **Custom Settings**.

   The **Custom Settings** page appears.
2. Click **Manage** for **CMDMMultiOrgSettings**.
3. Click **New**.
4. In the **Name** field, enter a unique name for the setting.

   Note: The name must match the data source name defined for the Spoke Org in the Master Org.
5. In the **Consumer Key** field, enter the consumer key that was generated in the Salesforce connected app screen.
6. In the **Consumer secret** field, enter the consumer secret that was generated in the Salesforce connected app screen.
7. In the **Hub Org Password** field, enter the password for the Master Org.
8. In the **Hub Org Username** field, enter the user name for the Master Org.
9. In the **Hub Org REST API URL** field, enter the following URL that calls the REST API:

   `https://<ServerName>.salesforce.com/services/apexrest/DSE/MasterData/`
10. In the **OAuth_Token_URL** field, enter the following URL that generates the token to connect to the Master Org:

    `https://login.salesforce.com/services/oauth2/token`

    Note: This URL is the Salesforce `OAuth` endpoint for token requests.
11. Click **Save**.

Verify the HubOrg Remote Site Setting

When you install the unmanaged package on a Spoke Org, the package creates the **HubOrg** remote site setting. Verify that the **HubOrg** remote site setting is available and edit the setting to change the URL for the remote site.

1. Perform one of the following tasks based on the Salesforce environment that you use:
   - In Salesforce Classic, perform the following tasks:
     1. Select your name and click **Setup**.
     2. Under **Administration Setup**, select **Security Controls** > **Remote Site Settings**.
   - In Lightning Experience, perform the following tasks:
     1. Click the quick access menu ( ) and then **Setup Home**.
        The **Setup Home** page appears.
     2. Under **SETTINGS**, select **Security** > **Remote Site Settings**.

   The **All Remote Sites** page appears.
2. Click **Edit**.
3. In the **Remote Site URL** field, replace the server name in the following URL:

   ```
   https://<ServerName>.salesforce.com
   ```

   Note: The server name is the server on which you install CC360.
4. Click **Save**.

   Note: If the **HubOrg** remote site setting is not available, then you must create it and change the URL for the remote site.

Verify the SFDC Remote Site Setting

When you install the unmanaged package on a Spoke Org, the package creates the **SFDC** remote site setting. Verify that the **SFDC** remote site setting is available.

1. Perform one of the following tasks based on the Salesforce environment that you use:
   - In Salesforce Classic, perform the following tasks:
     1. Select your name and click **Setup**.
     2. Under **Administration Setup**, select **Security Controls** > **Remote Site Settings**.
   - In Lightning Experience, perform the following tasks:
     1. Click the quick access menu ( ) and then **Setup Home**.
        The **Setup Home** page appears.
     2. Under **SETTINGS**, select **Security** > **Remote Site Settings**.

   The **All Remote Sites** page appears.
2. Confirm that the SFDC remote site setting is available.

   Note: If the **SFDC** remote site setting is not available, then you must create it.

Add the Visualforce Page for Real-Time Update

To ensure users can view real-time updates to an account from the Master Org, you need to add a Visualforce page to the account and person account page layouts on each Spoke Org.

1. Perform one of the following tasks based on the Salesforce environment that you use:
   - In Salesforce Classic, perform the following tasks:
     1. Select your name and click **Setup**.
     2. Under **App Setup**, select **Customize** > **Accounts** > **Page Layouts**.
   - In Lightning Experience, perform the following tasks:
     1. Click the quick access menu ( ) and then **Setup Home**.
        The **Setup Home** page appears.
     2. Under **PLATFORM TOOLS**, select **Objects and Fields** > **Object Manager**.
        The **Setup Object Manager** page appears.
     3. Click **Account**.
     4. Under the **Page Layouts** section, select **Account Layout**.

   The **Account Page Layout** page appears.
2. Click **Edit** to edit the account page layout.
3. Under **Fields**, click **Visualforce Pages**.
4. Drag **CMDExternalUpdate** page to the layout and specify the properties.
5. Click **Save**.

   Repeat the preceding steps to add the Visualforce page to the person account page layout.

Getting Updates from the Master Org in Real-Time

When you open an account or a person account in a Spoke Org, the Spoke Org gets updated field values in real-time. You can choose to apply or ignore the updates from the Master Org.

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Salesforce Classic, click the **Accounts** tab.
   - In Lightning Experience, on the navigation bar, click **Accounts**.
2. Open an account that you want to view.

   If updates to any of the fields specified as part of enrichment settings are available, the following screen appears:

Updates from the Hub Org for Firstname Guardian Life Insurance Co Account

Field Name	Field Current Value	Field Value returned from Hub
Phone		626-357-0747
Demo_Total_Revenue_YTD__c		3445565
Name	Firstname Guardian Life Insurance Co	Guardian Life Insurance Co
demopkg01__Delivery_Street__c		the Delivery Street
Domain__c		kaylajesse.com
Domain__pc		TestingIndia.com

Apply changes | Ignore Changes

3. Click **Apply Changes** to update the account or person account.

4. Click **Ignore Changes** to ignore the updates.

   If you ignore the updates, the `MultiOrg Last Update From Hub` field is updated with the timestamp even though the account is not updated. The next time you open the same account, if there are updates from the Master Org and you choose to apply changes, the account is updated with all changes from the Master Org.

A notification is posted on the chatter feed to notify users about the updated account or person account. The following image shows the chatter feed notification of an updated account:

Getting Updates from the Master Org in Batch Mode

To get updates for multiple accounts or person accounts from the Master Org, you need to run a batch job from the Spoke Org.

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Salesforce Classic, in the upper-right corner of the page, select your name, and click **Developer Console**.
   - In Lightning Experience, click the quick access menu (), and then click **Developer Console**.
2. Click **Debug** > **Open Execute Anonymous Window**.
3. Enter the following command to run the batch job for person accounts:

   ```
   CMDMMultiOrgsObjectSyncBatch b = new CMDMMultiOrgsObjectSyncBatch('PersonAccount');
      database.executeBatch(b, batchSize);
   ```

   Note: If you do not specify a batch size, the method uses the default batch size of 200. It is recommended to set the batch size to less than or equal to 200.
4. Enter the following command to run the batch job for accounts:

   ```
   CMDMMultiOrgsObjectSyncBatch b = new CMDMMultiOrgsObjectSyncBatch('Account');
      database.executeBatch(b, batchSize);
   ```

   Note: If you do not specify a batch size, the method uses the default batch size of 200. It is recommended to set the batch size to less than or equal to 200.

   The batch jobs to synchronize accounts and person accounts with the Master Org cannot be run concurrently.

5. Click **Execute**.

   During batch processing, if errors occur because of incorrect field types or incorrect field values, the batch job processes other records and adds information about error records to the `External Field Update Failure Name` table. If the subsequent batch job identifies errors for an account for which an error record already exists in the error table, the batch job overwrites the existing error record. If the batch job encounters errors during an update to an account, only one error message is added to the `External Field Update Failure Name` table.

   Note: In the Master Org, if a boolean field is updated with any value other than `True`, the Apex API `Boolean.ValueOf` returns false. For example, in the Master Org, a boolean field is updated with a number `1234` and the value of that field in the Spoke Org is `False`. When you run the batch job, the Apex API `Boolean.ValueOf` returns the value `False`. As a result, the batch job does not update the boolean field in the Spoke Org because both the values are `False`.

A notification is posted on the chatter feed to notify users about updated accounts or person accounts.

CHAPTER 12

Troubleshooting

This chapter includes the following topics:

- Troubleshooting CC360 Configuration, 129
- Troubleshooting Hierarchies, 131

Troubleshooting CC360 Configuration

If you encounter any issues when you configure CC360, use the following information to troubleshoot the issues.

When I edited hierarchy information for a record, I received an error message indicating that I do not have sufficient access rights.

When you update the hierarchy information of a child account, such as the parent account field, CC360 updates the hierarchy information for the parent account record. If you do not have sufficient privileges to edit the parent account record, you receive an error message.

If you receive an error message that indicates you do not have sufficient access rights, you need to disable account hierarchy enrichment. To disable account hierarchy enrichment, use the API_SettingsManager.enableHierarchyAccountEnrichment(false) API to set the DS_SETTING_HIERARCHY_ACCOUNT_ENRICHMENT internal CC360 parameter to `false`. DS_SETTING_HIERARCHY_ACCOUNT_ENRICHMENT is `true` by default.

If you update hierarchy information for a record when DS_SETTING_HIERARCHY_ACCOUNT_ENRICHMENT is `false`, CC360 only updates the records in the hierarchy for which you have sufficient privileges. To update hierarchy information for all records in the hierarchy, a system administrator must run the Synchronize Reporting Data batch job. Ensure that the Asynchronous Hierarchy Process batch job and Asynchronous Cloning Process batch job are not running when the Synchronize Reporting Data batch job runs.

When I access the CC360 Accounts page or when I run a batch job, I receive an error message indicating that I do not have necessary permissions.

If you do not have sufficient permissions to access the CC360 Accounts page or to run the batch job, you receive an error message.

After you install or upgrade CC360, assign appropriate permissions to the users through user profiles and permission sets.

To grant access permission to the **Accounts** page, follow these steps.

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Salesforce Classic, perform the following tasks:
     1. In the upper-right corner of the page, select your name, and click **Setup**.
     2. Under **Administration Setup**, click **Manage Users** > **Profiles**.
   - In Lightning Experience, perform the following tasks:
     1. Click the quick access menu (), and then click **Setup**.
        The **Setup Home** page appears.
     2. Under **ADMINISTRATION**, click **Users** > **Profiles**.

     The **User Profiles** page appears.
2. Click a user profile to which you want to assign permissions.
   The profile details appear.
3. Under the **Enabled Visualforce Page Access** section, click **Edit**.
4. Under **Available Visualforce Pages**, select **DS_AccountDQPage** and click **Add**.
   The **DS_AccountDQPage** is added to **Enabled Visualforce Pages**.
5. Click **Save**.

When I convert a bean to an account, contact, or lead, I receive an error message indicating that the system cannot convert the record to the specified target object.

If you do not have sufficient permissions to create or edit a Salesforce object, you receive an error message. Assign appropriate permissions to the users through user profiles and permission sets.

In Lightning Experience, when I merge records, I receive an error message indicating that I do not have necessary permissions.

If you do not have sufficient permissions to access some fields or if the fields are hidden for your user profile, you receive an error message while merging records.

Assign appropriate permissions to the users through user profiles and permission sets, or display the fields that are hidden for a user profile.

To display the fields for a user profile, follow these steps:

1. In Lightning Experience, click the quick access menu (), and then click **Setup**.
2. Under **SETTINGS**, click **Security** > **Field Accessibility**.
3. Select the type of record for which you want to edit the field accessibility.

Troubleshooting Hierarchies

If you encounter any issues when you configure hierarchies, use the following information to troubleshoot the issues.

When I edited hierarchy information for a record, I received an error message indicating that I do not have sufficient access rights.

When you update the hierarchy information of a child account, such as the parent account field, CC360 updates the hierarchy information for the parent account record. If you do not have sufficient privileges to edit the parent account record, you receive an error message.

If you receive an error message that indicates you do not have sufficient access rights, you need to disable account hierarchy enrichment. To disable account hierarchy enrichment, use the API_SettingsManager.enableHierarchyAccountEnrichment(false) API to set the DS_SETTING_HIERARCHY_ACCOUNT_ENRICHMENT internal CC360 parameter to `false`. DS_SETTING_HIERARCHY_ACCOUNT_ENRICHMENT is `true` by default.

If you update hierarchy information for a record when DS_SETTING_HIERARCHY_ACCOUNT_ENRICHMENT is `false`, CC360 only updates the records in the hierarchy for which you have sufficient privileges. To update hierarchy information for all records in the hierarchy, a system administrator must run the Synchronize Reporting Data batch job. Ensure that the Asynchronous Hierarchy Process batch job and Asynchronous Cloning Process batch job are not running when the Synchronize Reporting Data batch job runs.

APPENDIX A

Custom Settings

This appendix includes the following topics:

- CC360 Profiles, 132
- Record Types, 134
- CC360 Settings, 135
- Data Sources, 137
- Enrichment Settings, 138
- Internal Settings, 141
- Fuzzy Matching, 153
- International Settings, 159
- Job Monitor, 159
- Job Schedules, 160
- Legal Forms, 160
- Matching Settings, 161
- Object Classes, 165
- Performance Settings, 165
- Segment Settings, 166
- Synchronization Settings, 167
- Visualforce Page Fields, 169
- Configuring Custom Settings Manually, 169
- Configuring Custom Settings Using the Setup Wizard, 170
- Sharing the CC360 Settings, 170

CC360 Profiles

CC360 profiles contain user permissions to access and perform tasks in CC360. You must create profiles named Admin and Default. You may also create additional profiles.

CC360 Enable all the CC360 profile settings to create an Admin profile. To create a Default profile, save the default CC360 profile settings. CC360 assigns the Default profile if you do not specify profile for a user. You can create custom profiles for specific users. You can access the user account to specify the custom profile that you create for the user.

You can configure the following CC360 profile settings:

Name

Name of the profile.

Allow Change Master Account

Enables you to change the master record for duplicate accounts. Default is disabled.

Allow Change Master Contact

Enables you to change the master record for duplicate contacts. Default is disabled.

Allow Change Master Lead

Enables you to change the master record for duplicate leads. Default is disabled.

Allow Consolidate

Allows you to consolidate data. Default is enabled.

Allow Convert Data

Allows you to convert data. For example, you might want to convert a lead to an account. Default is enabled.

Allow Create Contacts without Accounts

Enables you to create contacts without accounts. Default is enabled.

Allow Creation of Duplicate Accounts

Enables you to create duplicate accounts. Default is disabled.

Allow Creation of Duplicate Contacts

Enables you to create duplicate contacts. Default is disabled.

Allow Creation of Duplicate Leads

Enables you to create duplicate leads. Default is disabled.

Allow Edit Hierarchy

Allows you to add accounts to a hierarchy, deleted accounts in a hierarchy, or reorganize the hierarchy.

Allow Ignore Duplicate Accounts

Enables you to ignore duplicate accounts and not include duplicate accounts in the duplicate account list. Default is disabled.

Allow Ignore Duplicate Contacts

Enables you to ignore duplicate contacts and not include the duplicate contacts in the duplicate contacts list. Default is disabled.

Allow Ignore Duplicate Leads

Enables you to ignore duplicate leads and not include the duplicate leads in the duplicate leads list. Default is disabled.

Allow Merge Accounts

Enables you to merge accounts. You must also have delete permission on accounts to be able to merge accounts. Default is enabled.

Allow Merge Contacts

Enables you to merge contacts. You must also have delete permission on contacts to be able to merge contacts. Default is enabled.

Allow Merge Leads

Enables you to merge leads. You must also have delete permission on leads to be able to merge leads. Default is enabled.

View Consolidated Information

Allows you to view consolidated information of a record from different sources through the Data Manager. Default is disabled.

Record Types

You can use record types to define different custom settings for the Salesforce standard and custom objects.

For example, you can use record types to define different synchronization settings for customer accounts and partner accounts.

You can use the record types to define the following custom settings for CC360:

- Synchronization Settings
- Segment Settings
- Enrichment Settings
- Data Sources
- Matching Settings

Note: A user can access only the record types that are associated with the user profile.

Default Record Type for Person Accounts

Before you run a data conversion batch job to convert a record to a person account, you must set the default record type for person accounts if the record type is enabled. The data conversion job uses the matching settings defined for the default record type if the matching settings are not available for a record type and the object class to which the record type belongs.

You can set the default record type for person accounts in the Edit Record Type Settings page that you can open from the Record Type Settings section of the profile detail page. You can also explicitly set the default record type for person accounts in the Salesforce Developer Console.

The default record type that you explicitly set takes precedence over the default record type that you set in the Edit Record Type Settings page. When you change the default record type in the Edit Record Type Settings page, ensure that you explicitly update or reset the default record type.

Setting the Default Record Type for Accounts

You must configure the following internal settings that explicitly set the default record types of accounts in the Salesforce Developer Console:

- For person accounts, Default Record Type For Person Account
- For business accounts, Default Record Type For Account

Use the following sample code in the Salesforce Developer Console to configure the internal settings that set the default record type:

```
API_SettingsManager.updateInternalSetting('Default Record Type For Person Account|
Default Record Type For Account', '<Record Type ID>');
```

For example, the following sample code sets the default record type for person accounts:

```
API_SettingsManager.updateInternalSetting('Default Record Type For Person Account',
'012i000000084CgAAI');
```

Resetting the Default Record Type of Accounts

You can reset the explicitly set default record types to reflect the default record types that you set in the Edit Record Type Settings page.

Use the following sample code in the Salesforce Developer Console to reset the explicitly set default record types of person accounts:

```
API_SettingsManager.resetDefaultRecordTypesValues();
```

CC360 Settings

CC360 settings are basic settings that are required to activate and define the global features of CC360. The settings are stored in a record and only one settings record can be active at a time.

You can configure the following CC360 settings:

Name

Name of the CC360 settings record that must be used by CC360. Default is Active. Do not change the default value. The value is case sensitive.

Account Duplicate Warning if Lead Exists

Generates a warning for duplicates if an account matches an existing lead. Default is disabled.

Automatic Mode Job Delay

Launches a job when changes are made to the database that requires the data to be reprocessed. Specify the number of minutes that CC360 must wait before launching jobs. Default is 2.

Concurrent Jobs Limit

Enables CC360 to process jobs in parallel. CC360 can run up to five APEX jobs at a time.

Enable CC360 Self-Tests

Enables CC360 to run self-tests when all other APEX tests are run. Default is disabled.

Extract Domain

Enables CC360 to extract the domain such as informatica.com from the website and email fields. Default is enabled.

Extract Legal Form

Enables CC360 to extract legal forms from company names. For example, if the company name is XYZ Solutions Ltd., CC360 extracts the legal form "Limited." Default is enabled.

Hierarchy batch size limit

The maximum number of records that a hierarchy batch job can process. The default value is 5000. Do not increase the value beyond 5000.

Hierarchy Scout Index Name

API name of an external ID field that you create on the master bean object. CC360 uses the hierarchy index name if the master bean deletion returns an hierarchy error.

Hierarchy scout node page size

The maximum number of child nodes that can appear at one time in the hierarchy view. The default value is 10. If you set the hierarchy scout node page size to a value higher than 50, hierarchy viewer performance might decrease.

Maximum hierarchy limit

The default value is 45000. Do not increase the value beyond 45000. A hierarchy can be larger than the maximum hierarchy limit.

Maximum trigger hierarchy limit

The threshold to process hierarchies synchronously. The default value is 500 and it is recommended to not increase the value beyond 500. The maximum trigger hierarchy limit is also the threshold to display the complete hierarchy in the hierarchy tree. If the number of items to process in a hierarchy edit exceeds the maximum trigger hierarchy limit plus 1, CC360 processes the hierarchy edits asynchronously with a batch job. Also, if the number of nodes in the hierarchy is equal to or below the maximum trigger hierarchy limit, CC360 displays the complete hierarchy in the hierarchy tree. However, the display of a stair case hierarchy is not related to the maximum trigger hierarchy limit.

Ignore ISO Code

Configures CC360 to ignore standardizing data that users enter, based on the two digit country ISO code. Default is disabled.

Lead Duplicate Warning if Account Exists

Enables CC360 to generate a warning for duplicates if a lead matches an existing account. Default is disabled.

Override Account

Enables CC360 to overwrite details in the account with data from a master bean. Default is disabled.

Maintenance Window Active

Specifies that the maintenance window is active and CC360 will not run. Used to avoid errors when you upgrade Salesforce sandboxes and production Orgs with new versions. Default is disabled.

Maintenance Window End Time

Specifies the time when the maintenance window will end. CC360 is updated within the specified maintenance window.

Maintenance Window Start Time

Specifies the time when the maintenance window will start. CC360 is updated within the specified maintenance window.

Match Using Record Type Rules

Enables or disables the use of record type matching settings for the duplicate check, duplicate contact check, matching, and data conversion jobs.

When enabled, the jobs use the matching settings defined for the record types and segregates data by record types, allowing you to keep duplicates in separate record types.

Default is disabled which means that the jobs use the match settings defined for the object.

Maximum Job Logs

Specify the maximum number of errors or successes that can be logged when a batch job is run.

Running Mode

Save Job Statistics

Enables CC360 to save job statistics. Default is disabled.

Search Result Limit

Maximum number of records that can appear in a CC360 search result. Default is 200.

Segment Key Index Length

Length of the optional indexed segment key generated in LDV Org.

Segment Key Index Name

API name of a new external ID field that you create on the master bean and bean objects.

Setup Completed

Activates the CC360 setup. Default is disabled.

Standardize Country

Standardizes the billing country to the billing country value that is defined in the international settings. For example, if the billing country is specified as United States, it is standardized to U.S. Default is enabled.

Use Default Settings

Enables CC360 to use the default settings. Default is disabled.

Validate Country

Validates the value specified in the Billing Country field. Default is enabled.

Data Sources

You can define the available data sources and set the priority of the fields for the data source in beans.

Use the setup wizard to configure the following data sources settings:

Name

Name of the data source or record type.

If the record type is enabled, use the following name format to define data source for one or more record types:

```
<Data Source>:<Record Type Name1>,<Record Type Name2>,...<Record Type NameN>
```

For example, SAP:Sales,Marketing indicates that data source definition is applicable to the Sales and Marketing record types.

Check for Duplicates on Create

Enables CC360 to check for duplicates when a new record is created. Default is disabled.

Default User ID

The user identified by this ID is set as the default owner of new account, lead, and contact beans.

Description

A description for the data source set.

Dynamic Priority

The API name of the numeric field where the dynamic priority value is stored.

Enable for Search

Displays the source as a search option in the Account, Lead, and Contact tabs. Default is enabled.

Object Class

Object class of the data source. For example, Account, Contact, or Product.

Priority Fields

Priority of the source on the corresponding field in beans in the consolidation process. You can set the priority of the fields ranging from 0 to 100. 100 is the highest priority.

If the priority is 0, CC360 does not update the corresponding field on the consolidated record and the master bean. If the priority is 100, CC360 updates the corresponding field on the master bean. If beans have an equal priority, the bean with the most current Last Updated Date is used to update the master bean. If the Last Updated is Null, the most current modification date is used to update the master bean.

Data Sources for a Record Type

If the record type is enabled in Salesforce, you can define data sources for each record type.

When you run a consolidation batch job, the batch job looks for the data source definition specific to each record type. If the batch job cannot find a data source definition for a record type, the batch job ignores the record.

Enrichment Settings

You can supplement records with information from external data sources. Configure the fields of the objects that Cloud MDM - Customer 360 for Salesforce must override when the fields in the associated master bean are updated.

You can enrich the following objects:

- Accounts, person accounts, cases, and opportunities.
- Custom objects.

Enrichment settings must have the same name as the object to facilitate mapping of fields from the account to the master bean. You must also configure the circumstances under which the field mapping can update.

You can configure the enrichment settings in the custom settings page or in the setup wizard. The enrichment settings of an object consist of a series of text fields, one for each field on the master bean that you can map to an account field.

To define a mapping in the custom settings, enter the API name for the account field that you need to map to in the relevant enrichment setting field. Follow the API name with a colon (:) and then the enrichment operator that describes how you want the enrichment to behave.

Use the following enrichment operators:

- - (hyphen). Ignores the field.
- 0 (zero). Updates the field only if the value is null.
- + (plus). Updates the field if the present value is not null.

- ++ (double plus). Updates the field.

For example, if you enter `BillingCity:++` in the City field, CC360 updates the field. The default operator is 0.

To define a mapping in the setup wizard, select the account field and the enrichment option for that field.

You can configure the following enrichment settings:

Name

Name of the object type or the record type for which you want to define the enrichment settings.

You can use one of the following name formats to define enrichment settings:

- For an object type, `<Object Name>`
  For example, PersonAccount.
- For one or more record types, `<Object Name>:<Record Type Name1>,<Record Type Name2>,...<Record Type NameN>`
  For example, Account:Personal,Sales indicates that the enrichment settings are applicable to the Personal and Sales record types if the record type is enabled.

City

Mapping for the city field on the bean. Specify the API name of the corresponding field on the bean.

Country

Mapping for the country field on the bean. Specify the API name of the corresponding field on the bean.

Postal Code

Mapping for the postal code field on the bean. Specify the API name of the corresponding field on the bean.

State

Mapping for the state field on the bean. Specify the API name of the corresponding field on the bean.

Street

Mapping for the street field on the bean. Specify the API name of the corresponding field on the bean.

Birthdate

Mapping for the birthdate field on the bean. Specify the API name of the corresponding field on the bean.

Company

Mapping for the company field on the bean. Specify the API name of the corresponding field on the bean.

Custom Fields 1 through 20

Mapping for custom fields 1 through 20 on the bean. Specify the API name of the corresponding field on the bean.

Quick Cleansing

Accelerates the data cleansing process and reduces the number of script statements in the Salesforce.com triggers. Default is disabled.

Domain

Mapping for the domain field on the bean. Specify the API name of the corresponding field on the bean.

Email

Mapping for the email field on the bean. Specify the API name of the corresponding field on the bean.

First Name

Mapping for the first name field on the bean. Specify the API name of the corresponding field on the bean.

Keep Data after Object Deletion

Saves a copy of the data in the data pool after deletion of the object in Salesforce.com. Default is disabled.

Last Name

Mapping for the last name field on the bean. Specify the API name of the corresponding field on the bean.

Mobile Phone

Mapping for the mobile phone field on the bean. Specify the API name of the corresponding field on the bean.

Parent Account

Mapping for the parent account field on the bean. Specify the API name of the corresponding field on the bean.

Parent ID

Mapping for the parent ID field on the bean. Specify the API name of the corresponding field on the bean.

Phone

Mapping for the phone field on the bean. Specify the API name of the corresponding field on the bean.

Record Type ID

Mapping for the record type ID field on the bean. Specify the API name of the corresponding field on the bean.

Synchronize

Automatically synchronizes the corresponding Salesforce.com object such as Account, Lead, and Contact. Default is enabled.

Ultimate Parent Name

Mapping for the name of the ultimate parent in the hierarchy.

Ultimate Parent

Mapping for the ID of the ultimate parent in the hierarchy.

Website

Mapping for Website field on the bean. Specify the API name of the corresponding field on the bean.

Note: When you update a date or a time field, use the following format: For date. `<yyyy>-<mm>-<dd>`

For time. `<hh>:<mm>:<ss>`

For example, `1982-03-21 11:20:32`

Enrichment Settings for a Record Type

If the record type is enabled in Salesforce, you can define enrichment settings specific to each record type of an object.

When you run a consolidation batch job, the batch job looks for the enrichment settings in the following order:

1. Enrichment settings specific to each record type.
2. If the enrichment settings are not available for a record type, the batch job looks for the enrichment settings defined for the object to which the record type belongs.

   For example, if you do not define enrichment settings for an account record type named Account:Personal, the batch job looks for the enrichment settings defined for the Account object.
3. If the enrichment settings are not available for the object to which the record type belongs, the batch job uses the default enrichment settings of the object.

Internal Settings

Use internal settings to change the default behavior of CC360, such as logging batch statistics, setting the default record types of accounts, and skipping master bean validation.

You can configure the internal settings from the Setup Wizard or through the Salesforce Developer Console.

Configuring Internal Settings by Using the Setup Wizard

You can enable or disable internal settings in the Setup Wizard and customize your organization.

1. From the **Setup** section of the **CC360 Admin** tab, click **Run the Setup Wizard**.
2. Click **Internal Settings**.

   The **Internal Settings** page appears.
3. Select or clear the following the internal settings:

   Person Account vs Lead Comparison

   Indicates whether to enable duplicate check for person accounts, leads, and contacts. Default is disabled.

   Update Processed Count in Execute

   Indicates whether to update the processed record count in the Batch Queue object. Default is disabled.

   Show Comparison Types in Matching

   Indicates whether to display the type of matching used in the **Fuzzy Data Comparison** page. Default is enabled.

   Queue Monitor Automatic Refresh

   Indicates whether to automatically refresh the Running Jobs table and the Queue Entries table in the batch queue. Default is enabled.

Log Response JSON from Informatica MDE

Indicates whether to log the JSON response from Informatica MDM Multidomain Edition for debugging purpose. Default is disabled.

Log QueryLocator Statement

Indicates whether to log the return statements from the Salesforce QueryLocator method. Default is disabled.

Log Batch Statistics

Indicates whether to log the batch statistics. Default is disabled.

Enable Alternate Hierarchy Mode

Indicates whether to display the alternate hierarchy in the Hierarchy section of the account page. Default is disabled.

Synchronous Run Automatic

Indicates whether to synchronously run the matching and duplicate contact check jobs. Default is disabled.

Reschedule Failed Batch Jobs

Indicates whether to reschedule the failed batch jobs. Default is disabled.

Display API Name

Indicates whether to display the API names of the fields or objects instead of their labels in the **DaaS Mapping** and **Synchronization Setting** pages.

Disable Object Types in Scout Pages

Indicates whether to display the Salesforce object types in Scout pages.

Disable Record Types in Scout Pages

Indicates whether to display the record types in Scout pages.

4. Click **Save**.
5. To use the default settings, click **Revert to Defaults**.
6. Click **Done**.

Configuring Internal Settings by Using the Salesforce Developer Console

You can also use the Salesforce Developer Console to configure the internal settings.

You can configure internal settings for the following CC360 components:

- Batch jobs
- Beans
- DaaS
- Fields
- Hierarchies
- Leads
- Multi-Org
- Page layout

- Record type
- Picklists
- Visualforce Scout pages

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Salesforce Classic, in the upper-right corner of the page, select your name, and click **Developer Console**.
   - In Lightning Experience, click the quick access menu ( ), and then click **Developer Console**.
2. In the **Force.com Developer Console** window, under **Debug**, select **Open Execute Anonymous Window**.

   The **Enter Apex Code** window opens.
3. Enter the code for configuring the internal setting.
4. Click **Execute**.

Internal Settings for Batch Jobs

You can configure internal settings and change the default behavior of the batch jobs.

You can configure the following internal settings for the batch jobs:

Custom Filter:<Job name>

You can configure custom filters for the batch jobs to filter the records that you want to process. Custom filters allow you to filter the records that you want to process. For example, you can configure a filter to identify records that belong to a particular company.

Note: You can configure custom filters for all batch jobs except the Data Conversion batch job and the Data as a Service batch job.

To configure custom filters, use the `Custom Filter:<Job name>` internal setting and set a value to it.

Use the following format to set a value for the custom filter internal setting:

```
API_SettingsManager.updateInternalSetting('Custom Filter:<Job name>', '<Filter
1>' [<Operator> '<Filter 2>]');
```

The following table describes the parameters used in the format of the custom filter internal setting value:

Parameter	Description

Job name	Name of the batch job.
	You can set one of the following values based on the batch job for which you want to configure the custom filters:
	`- Consolidation`
	`- DuplicateBeanCheck`
	`- DuplicateCheck`
	`- ExternalCleansing`
	`- HierarchyMigration`
	`- Matching`
	`- MergeAccount`
	`- MergeContact`
	`- MergeLead`
	`- Migrate`
	`- MigrateAccount`
	`- MigrateContact`
	`- MigrateLead`

Filter 1	Filter criteria to identify the records that you want the batch job to process. For example, `DS_Last_Updated__c > 2014-12-314T05:42:14+0000`
Filter 2	Optional. Additional filter criteria to identify the records that you want the batch job to process. For example, `DS_company__c='INFA'`.
Operator	Required if you use more than one filter criteria. Logical operators `AND`. For example, `(DS_Last_Updated__c > 2014-12-314T05:42:14+0000 AND DS_company__c='INFA')`.

The following sample code filters records for the Matching job :

```
API_SettingsManager.updateInternalSetting('Custom
Filter:Matching','DS_company__c='INFA' AND 'DS_Record_Type_Name__c='Sales');
```

Note: You cannot configure custom filters for the previously scheduled jobs.

Current Set Number

If you have batch jobs that run in multiple concurrent processes, you can configure the `Current Set Number` internal setting to populate the Concurrent Set custom field of the Batch Queue object. The value of the Concurrent Set custom field identifies related batch queue records when batch jobs run in multiple concurrent processes.

To populate the Concurrent Set custom field of the Batch Queue object, set the value of the `Current Set Number` internal setting. Default is 0.

The following sample code populates the Concurrent Set custom field:

```
API_SettingsManager.updateInternalSetting('Current Set Number','10');
```

Note: The value is incremented whenever the batch jobs run in multiple concurrent processes.

enableDefaultJobPriorityForRunAutomatic

If a user does not define priorities or defines incorrect priorities for Run Automatic batch jobs in the performance settings, CC360 assigns default priorities for each batch jobs. You can switch to the default priorities instead of CC360 priorities.

CC360 assigns the following default priorities to the batch jobs:

Batch Job Name	Priority
Matching	1
Duplicate Contact Check	2
Duplicate Check	3
Consolidation	4
Conversion	5

To switch to default priorities, set the value of the `enableDefaultJobPriorityForRunAutomatic` internal setting to true. Default is false which indicates that the jobs run on user-defined priorities.

The following sample code switches to default priorities instead of user-defined priorities:

```
API_SettingsManager.enableDefaultJobPriorityForRunAutomatic(true);
```

enableQueueMonitorAutomaticRefresh

You can disable the automatic refresh of the Running Jobs table and the Queue Entries table in the batch queue.

To disable automatic refresh, set the value of the `enableQueueMonitorAutomaticRefresh` internal setting to false. Default is true.

The following sample code disables the automatic refresh of the Running Jobs table and the Queue Entries table:

```
API_SettingsManager.enableQueueMonitorAutomaticRefresh(false);
```

Log Batch Statistics

CC360 does not log batch statistics.

To log batch statistics, set the value of the `Log Batch Statistics` internal setting to true.

The following sample code logs the batch statistics:

```
API_SettingsManager.updateInternalSetting('Log Batch Statistics','true');
```

Synchronous Run Automatic

CC360 can synchronously run the matching and the duplicate contact check jobs for a record whenever a user adds a record.

To synchronously run the matching and the duplicate contact check jobs, set the value of the `Synchronous Run Automatic` internal setting to 0. Default is 1.

The following sample code synchronously runs the matching and the duplicate contact check jobs:

```
API_SettingsManager.updateInternalSetting('Synchronous Run Automatic',0);
```

Reschedule Failed Jobs

CC360 job scheduler reschedules only the successfully completed jobs.

To reschedule the failed jobs, set the value of the `Reschedule Failed Jobs` internal setting to true. Default is false.

The following sample code configures the job scheduler to reschedule the failed jobs:

```
API_SettingsManager.updateInternalSetting('Reschedule Failed Jobs','true');
```

Internal Settings for Beans

You can configure internal settings and change the default behavior of beans.

You can configure the following internal settings for beans:

Ignore Deleted Beans

The beans that have the Account Deleted, Lead Deleted, or Contact Deleted flag enabled in the bean details page are not part of the consolidation process.

If you want the deleted beans to be a part of the consolidation process, set the `Ignore Deleted Beans` internal setting value to false. Default is true.

The following sample code indicates to include the deleted beans in the consolidation process:

```
API_SettingsManager.updateInternalSetting('Ignore Deleted Beans','false');
```

Exclude Deleted Beans

When you search for records in the Lightning and Visualforce Scout pages, the beans that have the Account Deleted, Lead Deleted, or Contact Deleted flag enabled in the bean details page are not displayed in the search result.

If you want the deleted beans to be displayed in the search result, set the `Exclude Deleted Beans` internal setting value to false. Default is true.

The following sample code indicates to include the deleted beans in the search result:

```
API_SettingsManager.updateInternalSetting('Exclude Deleted Beans','false');
```

SkipMBCount[:<Master Bean Name>]

If you exceed the Salesforce governor limits in your organization, you cannot link the beans of other records to the master bean of an account record.

You can enable the linking of beans to the master bean if you skip validation of master beans.

To skip the validation of master bean, set the `SkipMBCount` internal setting value to true. Default is false.

The following sample code skips validation for all master beans:

```
API_SettingsManager.updateInternalSetting('SkipMBCount','true');
```

To skip the validation for a specified master bean, set the `SkipMBCount:<Master Bean Name>` internal setting value to true. Default is false.

The following sample code skips validation for a specified master bean:

```
API_SettingsManager.updateInternalSetting('SkipMBCount:MB-288306','true');
```

Internal Settings for DaaS

If you have DaaS enabled in your organization, you can configure internal settings and change the default behavior of DaaS and the DaaS jobs.

You can configure the following internal settings for DaaS and DaaS jobs:

DaaS Trigger Lookback in mins

When you create a record in the Scout pages and use DaaS to verify address, email address, and phone number, the verification details are stored as a transaction entry in the EventQueue table. Since the record is not yet saved in the database, the verification details are not associated with any Salesforce object.

CC360 uses the user ID and object name to associate Salesforce objects with the verification details in the EventQueue table created within the time period specified in the `DaaS Trigger Lookback in mins` internal setting.

If you save the record after a time frame that is greater than the `DaaS Trigger Lookback in mins` value, the record is saved with a not-verified status. The `DaaS Trigger Lookback in mins` internal setting ensures that the objects get associated with valid and updated verification details.

To specify the time period, set the `DaaS Trigger Lookback in mins` internal settings value in minutes. Default is 30 minutes.

The following sample code specifies the time frame which CC360 uses to associate the verification details with a Salesforce object:

```
API_SettingsManager.updateInternalSetting('DaaS Trigger Lookback in mins','5');
```

DisableDaasTrigger

You can disable DaaS for standard Salesforce objects and custom objects.

To disable DaaS, set the value of the `DisableDaasTrigger_<sObjectKeyPrefix>|<sObjectName>` internal setting to 1, where `sObjectName` indicates the name of the Salesforce object and `sObjectKeyPrefix` indicates the key prefix of the object. Default is 0 which indicates that the DaaS is enabled.

The following sample code disables DaaS for an account with key prefix 001:

```
API_SettingsManager.updateInternalSetting('DisableDaasTrigger_001',1);
```

The following sample code disables DaaS for accounts:

```
API_SettingsManager.updateInternalSetting('DisableDaasTrigger_Account',1);
```

LTNG Auto Verify Address Suggestions

When you verify addresses in Lightning Experience and select a suggested address, DaaS does not verify the selected address again. You can configure to verify a selected address again.

To verify a selected address, set the `LTNG Auto Verify Address Suggestions` internal setting value to `false`.

The following sample code verifies a selected address again:

```
API_SettingsManager.updateInternalSetting('LTNG Auto Verify Address
Suggestions','false');
```

MaxSchedulesData as a Service

You can create multiple templates for multiple instances of the DaaS job. You can run each job at different schedules on different objects. If you do not have any governor limits in your organization, you can also set the maximum number of schedules that you can configure for the DaaS job. Default is 30 schedules.

To change the default maximum number of schedules that you can configure, configure the `MaxSchedulesData as a Service` internal setting.

The following sample code changes the maximum number of schedules for the DaaS job:

```
API_SettingsManager.updateInternalSetting('MaxSchedulesData as a Service', 50);
```

Sfdc Webservice Timeout in Milli Secs

You can configure the Salesforce web service timeout interval for the address, email address, and phone number verification services. Default is 120 seconds.

To change the default time interval, configure the `Sfdc Webservice Timeout in Milli Secs` internal setting.

The following sample code configures the Salesforce web service timeout interval:

```
API_SettingsManager.updateInternalSetting('Sfdc Webservice Timeout in Milli
Secs','10000');
```

Internal Setting for Fields

Fuzzy search uses the first character of the company name in the record segment to search for matching or duplicate records. You can configure the length of the characters used in the record segment.

To configure the length of the characters in the Company field, set the `Fuzzy Search Company Field Size` internal setting value. Default is 1 which means that the first character of the company name is used in the fuzzy search of duplicate records.

The following sample code configures the length of the characters in Company field:

```
API_SettingsManager.updateInternalSetting('Fuzzy Search Company Field Size','3');
```

Internal Settings for Hierarchies

You configure internal settings and change the default behavior of hierarchies.

You can configure the following internal settings for hierarchies:

Enable Alternate Hierarchy Mode

You must enable alternate hierarchy to view alternate hierarchies in the Hierarchy section of the account page.

To enable alternate hierarchy, set the `Enable Alternate Hierarchy Mode` internal setting value to true. Default is false.

The following sample code enables alternate hierarchy:

```
API_SettingsManager.updateInternalSetting('Enable Alternate Hierarchy Mode','true');
```

Alternate Hierarchy Page Size

You can change the maximum number of nodes that an alternate hierarchy displays.

To change the maximum number of nodes for alternate hierarchies, set the `Alternate Hierarchy Page Size` internal setting value. Default is 200.

The following sample code changes the maximum number of nodes for alternate hierarchies:

```
API_SettingsManager.updateInternalSetting('Alternate Hierarchy Page Size','100');
```

enableHierarchyAccountEnrichment

If you disable account hierarchy enrichment, you can update the records in the hierarchies for which you do not have sufficient privileges.

To disable account hierarchy enrichment, set the `enableHierarchyAccountEnrichment` internal setting value to false. Default is true.

The following sample code disables account hierarchy enrichment:

```
API_SettingsManager.updateInternalSetting('enableHierarchyAccountEnrichment','False')
;
```

Lock Cloud MDM Hierarchy and Unlock Cloud MDM Hierarchy

You can lock single or multiple hierarchies to prevent users from editing the hierarchy-related fields of the nodes.

To lock a hierarchy, use the `Lock Cloud MDM Hierarchy(<Account ID>)` internal setting where `<Account ID>` is the parent account ID.

The following sample code locks hierarchy of the parent account named `accid`:

```
DSE.API_HierarchyHandler.LockCloudMDMHierarchy(accid);
```

You can also unlock locked hierarchies. To unlock a hierarchy, use the `Unlock Cloud MDM Hierarchy(<Account ID>)` internal setting.

The following sample code unlocks hierarchy of the parent account named `accid`:

```
DSE.API_HierarchyHandler.UnLockCloudMDMHierarchy(accid);
```

Hierarchy Details Page Size

You can configure the number of records that the Customer 360 page displays.

To change the number of nodes, set the `Hierarchy Details Page Size` internal setting value. Default is 25.

The following sample code changes the number of nodes displayed in the hierarchy details page to 50:

```
API_SettingsManager.updateInternalSetting('Hierarchy Details Page Size','50');
```

Use Inline Hierarchy Related List

If you want to see contacts, leads, or opportunities in the Account Details page, set the `Use Inline Hierarchy Related List` internal setting value to true. Default is false.

The following sample code displays contacts, leads, or opportunities in the Account Details page:

```
API_SettingsManager.updateInternalSetting('Use Inline Hierarchy Related
List','true');
```

Internal Settings for Leads

You can configure internal settings and change the default behavior of leads.

You can configure the following internal settings for leads:

Create Opportunity Task

If you convert a lead to an opportunity, CC360 creates tasks for that opportunity.

If you do not want to create tasks for the opportunities, set the `Create Opportunity Task` internal setting value to false. Default is true.

The following sample prevents creation of tasks for the opportunities:

```
API_SettingsManager.updateInternalSetting('Create Opportunity Task','false');
```

Disable Validation On Converted Lead

The leads that have the Require Validation for Converted Leads flag enabled in the Lead Settings page are part of the validation process after you convert them.

If you do not want to validate the converted leads, set the `Disable Validation On Converted Lead` internal setting value to true. Default is false.

The following sample code disables validation for the converted leads:

```
API_SettingsManager.updateInternalSetting('Disable Validation On Converted
Lead','true');
```

EnableLeadPersonAccountComparison

You can enable duplicate check for person accounts, leads, and contacts.

To enable duplicate check, set the `EnableLeadPersonAccountComparison` internal setting value to true. Default is false.

The following sample code enables duplicate check for person accounts, leads, and contacts:

```
API_SettingsManager.updateInternalSetting('EnableLeadPersonAccountComparison','True')
;
```

Enable LeadConversion Name Search

You can perform search operation for a lead even if the segment of the account linked to the lead is different from the segment of the lead.

To perform the search operation, set the `Enable LeadConversion Name Search` internal setting to true. Default is false.

The following sample code enables search operation for a lead even if the segment of the account linked to the lead is different from the segment of the lead:

```
API_SettingsManager.updateInternalSetting('Enable LeadCoversion Name Search','true');
```

Internal Setting for Multi-Org

You can limit the number of responses from the Master Org to Spoke Orgs.

To limit the responses, set the `MasterDataResponseSize` internal setting value.

The following sample code sets the value of the internal setting to 200:

```
API_SettingsManager.updateInternalSetting('MasterDataResponseSize','200');
```

Internal Setting for the Visualforce Scout Pages

When you use DaaS to verify address, email address, or phone number in a Visualforce Scout page, the page times out in 30 seconds. You can configure the timeout interval that can range from 30 through 120 seconds. Default is 30 seconds.

To configure the timeout interval, use the `Sfdc Visualforce Timeout in Milli Secs` internal setting.

The following sample code sets the timeout interval to 40 seconds:

```
API_SettingsManager.updateInternalSetting('Sfdc Visualforce Timeout in Milli Secs',
40000);
```

Internal Settings for Page Layout

You can configure internal settings and change the page layout of the Visualforce pages.

You can configure the following internal settings:

Roles to Hide Search on Scout Pages

You can hide the custom buttons in the CC360 account, contact, and lead Visualforce pages. The custom buttons are **Search**, **Search Option**, **Hide Required Fields**, and **Show Required Fields**.

To hide the custom buttons for a specific user profile, set the value of the `Roles to Hide Search on Scout Pages` internal setting to the user profile. By default, the custom buttons are displayed in the account, contact, and lead Visualforce pages.

The following sample code indicates to hide the custom buttons for the Marketing profile:

```
API_SettingsManager.updateInternalSetting('Roles to Hide Search on Scout
Pages','MarketingProfile');
```

Account Scout Redirect URL

You can override the account Visualforce page and redirect it to a custom account Visualforce page.

To override the account Visualforce page, set the value of the `Account Scout Redirect URL` internal setting to the URL of a custom account Visualforce page.

Use the following code format to override the account Visualforce page and redirect it to a custom Visualforce page.

```
API_SettingsManager.updateInternalSetting('Account Scout Redirect URL','/apex/<URL
of the custom account Visualforce page>id=@RECORDID@');
```

For example, the following sample code sets the value of the internal setting to the URL of a custom account Visualforce page:

```
API_SettingsManager.updateInternalSetting('Account Scout Redirect URL','/apex/
MyAccountRedirectpagename?id=@RECORDID@');
```

Contact Scout Redirect URL

You can override the contact Visualforce page and redirect it to a custom contact Visualforce page.

To override the contact Visualforce page, set the value of the `Contact Scout Redirect URL` internal setting to the URL of a custom contact Visualforce page.

Use the following code format to override the contact Visualforce page and redirect it to a custom contact Visualforce page:

```
API_SettingsManager.updateInternalSetting('Contact Scout Redirect URL','/apex/<URL
of the custom contact Visualforce page>id=@RECORDID@');
```

For example, the following sample code sets the value of the internal setting to the URL of a custom contact Visualforce page:

```
API_SettingsManager.updateInternalSetting('Contact Scout Redirect URL','/apex/
MyContactRedirectpagename?id=@RECORDID@');
```

Lead Scout Redirect URL

You can override the lead Visualforce page and redirect it to a custom lead Visualforce page.

To override the lead Visualforce page, set the value of the `Lead Scout Redirect URL` internal setting to the URL of a custom lead Visualforce page.

Use the following code format to override the lead Visualforce page and redirect it to a custom lead Visualforce page:

```
API_SettingsManager.updateInternalSetting('Lead Scout Redirect URL','/apex/<URL of
the custom lead Visualforce page>id=@RECORDID@');
```

For example, the following sample code sets the value of the internal setting to the URL of a custom lead Visualforce page:

```
API_SettingsManager.updateInternalSetting('Lead Scout Redirect URL','/apex/
MyLeadRedirectpagename?id=@RECORDID@');
```

Display API Name

The object list in the **Synchronization Settings** page displays the labels of the object.

To display the API names of the objects, set the value of the `Display API Name` internal setting to true. Default is false.

The following sample code configures the **Synchronization Settings** page to display the API names of the objects:

```
API_SettingsManager.updateInternalSetting('Display API Name','true');
```

Internal Settings for Record Type

You can configure internal settings and change the default behavior of record types.

You can configure the following internal settings for record type:

Default Record Type For Person Account

You can explicitly set the default record type for person accounts.

To set the default record type for person accounts, configure the `Default Record Type For Person Account` internal setting.

The following sample code sets the default record type for person accounts:

```
API_SettingsManager.updateInternalSetting('Default Record Type For Person
Account','012i000000084CgAAI');
```

`012i000000084CgAAI` is the record type ID of the record type.

Default Record Type For Account

You can explicitly set the default record type for business accounts.

To set the default record type for business accounts, configure the `Default Record Type For Account` internal setting.

The following sample code sets the default record type for business accounts:

```
API_SettingsManager.updateInternalSetting('Default Record Type For
Account','013i000000089CgAAI');
```

`013i000000089CgAAI` is the record type ID of the record type.

resetDefaultRecordTypesValues

You can reset the explicitly set default record types to reflect the default record types that you set in the **Edit Record Type Settings** page.

To reset the default record types, use the `resetDefaultRecordTypesValues` internal setting.

The following sample code resets the explicitly set default record types of person accounts:

```
API_SettingsManager.resetDefaultRecordTypesValues();
```

Internal Setting for Picklists

If you have state and country picklists enabled in your organization, you must configure internal setting.

In Salesforce, if state and country picklists are enabled, the state and country text fields are saved as Integration Value fields. An integration value is a customizable text value that is linked to a state or country code. When you update a code value on a record, that record's state or country text field is populated with the corresponding integration value. Similarly, when you update a state or country text field with a valid integration value, the corresponding state or country code field is updated.

In a state and country picklists-enabled organization, if the code or text value of a state or country is updated, the integration values on your records are not updated. Name values are not stored on records. So you must enable the `enableExtractCountryFromPicklist` to retrieve country names from Salesforce state or country code.

To dynamically retrieve the state and country names, set the value of the `enableExtractCountryFromPicklist` internal setting to true. Default is false.

The following sample code enables retrieval of state and country names from Salesforce state or country code:

```
DSE.API_SettingsManager.enableExtractCountryFromPicklist(true);
```

Fuzzy Matching

Fuzzy Matching Overview

Fuzzy matching can match strings that are not exactly the same but have similar characteristics and similar patterns. The duplicate check batch job, duplicate contact check batch job, and matching batch job rely on fuzzy matching to identify duplicates and matches.

Matching is a two step process that determines a match score between two records. First, CC360 performs an exact match on the matching segments to exclude records that are unlikely to have matches. Then, CC360 performs a fuzzy match on the remaining records to calculate a match score between pairs of records. If the match score of the two records achieves the match score threshold, CC360 considers the two records a match.

You can configure how the fuzzy matching process calculates the match score. Use the Fuzzy Data Comparison page to compare two records to test your matching settings.

Matching Segment

The segment field in the master bean record contains a matching segment. The matching segment is a string of characters that CC360 uses to filter records before it performs fuzzy matching.

To improve fuzzy match performance, CC360 performs an exact match on the matching segments to eliminate records that are unlikely to match. CC360 then performs fuzzy matching on the remaining records.

By default, CC360 creates the segment using the two characters of the ISO country code followed by the characters in the company norm field. You can configure CC360 to use any two fields to create a matching segment.

The following graphic shows the matching segment, the company norm, and the country ISO code on the **Master Bean Detail** page:

1. Matching Segment Field
2. Company Norm Field
3. Country ISO Code Field

Batch Acceleration Level

The batch acceleration level determines the inclusiveness of the exact match on the matching segment. You can increase the batch acceleration level to increase fuzzy matching performance.

The batch acceleration level is the number of characters of the match segment that Cloud MDM uses to match. For example, if you set the batch acceleration level to 10, then the first 10 characters of the match segment must match before the records can proceed to the fuzzy matching process. CC360 eliminates any record whose first 10 characters of the matching segment do not match another record from the fuzzy matching process.

Start with a low batch acceleration level to be more inclusive in your fuzzy match. You can then increase the batch acceleration level to exclude more records from the match process to improve fuzzy matching performance. If you increase the batch acceleration level too much, you might exclude matching records from the fuzzy matching process. CC360 does not consider records excluded from the fuzzy matching process as matches.

Configuring the Batch Acceleration Level

The batch acceleration level determines the inclusiveness of the exact match on the matching segment. You can increase the batch acceleration to increase fuzzy matching performance.

1. Perform one of the following tasks based on the Salesforce environment that you use:
   - In Salesforce Classic, perform the following tasks:
     1. Select your name and click **Setup**.
     2. Under **App Setup**, select **Develop** > **Custom Settings**.
   - In Lightning Experience, perform the following tasks:
     1. Click the quick access menu ( ) and then **Setup Home**.
        The **Setup Home** page appears.
     2. Under **PLATFORM TOOLS**, select **Custom Code** > **Custom Settings**.

   The **Custom Settings** page appears.
2. Select **Manage** for **Performance** settings.

   The **Performance Settings** page appears.
3. From the **Performance Setting** page, select **Edit** for the performance setting of a batch job.
4. In the **Batch Acceleration Level** field of the **Performance Settings Edit** page, enter the batch acceleration level.
5. Click **Save**.

Matching Settings

You can configure the matching settings to determine how CC360 calculates match scores. You can also configure the matching settings to determine how exclusive the initial prefiltering is. You can set the match score threshold that two records must achieve for CC360 to consider the records a match.

The matching settings affect the results of the matching and duplicate check batch jobs. CC360 compares records field by field and gives scores for the field matches. CC360 then adds the scores of the weighted field matches to determine a total match score. If the score is at or higher than the match score threshold, CC360 considers the records a match.

The matching batch job uses the high level matching threshold level in the matching settings. Edit the account matching settings to configure matching settings for the matching batch job.

The duplicate check and duplicate contact check batch jobs use the duplicate threshold level in the matching settings. Edit the account matching settings to configure matching settings for the duplicate check batch job. Edit the contact matching settings to configure matching settings for the duplicate contact check batch job.

You can also assign a weighting to each field score to determine the impact that the field score has on the total match score. Use the Fuzzy Data Comparison Page to test your match settings before you run the matching, duplicate check, and duplicate contact check batch jobs.

Match Scores

CC360 generates a match score for the records processed by the fuzzy matching process. You can configure how CC360 calculates the match score.

During fuzzy matching, CC360 calculates a field score between zero and 100 for the standard fields and the first four custom fields. You can configure the weighting that CC360 gives each field score. The total match score is the sum of the weighed match scores.

You can also configure how CC360 calculates the field score when one field is null.

Match Score Weighting

The match score weighting determines the impact a field score has on the total match score.

CC360 calculates a total match score based on the total weighted field scores. The weighted field score ensures fields that can help uniquely identify a company have more impact on the total match score than a field that is common among many records.

For example, the company name is likely to uniquely identify a record. Conversely, the billing city does not uniquely identify a record because many people and businesses can share the same city. You can give the company name a higher weighting than the city name field to ensure the company name score has a greater influence on the total match score.

The weighting is a percentage between 0 and 100. If you set the weighting to 50 and the field score is 75, CC360 calculates the weighted field score as 50% of 75. The weighted field score in this example is 37.5.

You can also set the weighting to 200 to ensure a field is an exact match. If you set the weighting for a field to 200 and the fields are not an exact match, CC360 sets the total match score to zero. You can use a weighting of 200 for the country ISO code field. A company with branches in two different countries might have many matching fields, but you might not want CC360 to consider the two branches a match because they are two separate business entities. Set the weighting to 200 on the country ISO code field to ensure CC360 does not consider these two separate business entities as the same entity.

Note: Assign a weighting of 200 with care. If a field with a weighting of 200 is not an exact match, CC360 does not consider the two records a match, regardless of how closely other fields match.

Configuring the Match Score Weightings

The match score weightings determines the impact a field score has on the total match score. CC360 calculates a total match score based on the total weighted field scores. The duplicate check batch jobs use the weighted field scores to identify duplicate records.

To configure the match score weightings, edit the matching settings.

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Salesforce Classic, perform the following tasks:
     1. In the upper-right corner of the page, select your name, and click **Setup**.
     2. Under **App Setup**, click **Develop** > **Custom Settings**.
   - In Lightning Experience, perform the following tasks:
     1. Click the quick access menu ( ), and then click **Setup**.
        The **Setup Home** page appears.
     2. Under **PLATFORM TOOLS**, click **Custom Code** > **Custom Settings**.

   The **Custom Settings** page appears.
2. Select **Manage** for **Matching Settings**.

   The **Custom Setting Matching Settings** page appears.
3. Click **Edit** for the object settings you want to edit or click **New** to create object settings.
4. On the **Matching Settings Edit** page, enter a weighting for each of the weighting fields.
5. Click **Save**.

Null Scores

The null score settings determines how CC360 calculates the field score when one field for one of the records is null.

You can set the score that CC360 gives a field when one of the fields is null. For example, two records might be a duplicates even if one of the records has a null email field. The email field is not critical to the match, so you can configure CC360 to assign a score when one field is null so the null field does not penalize the total match score.

Configuring the Null Scores

The null score determines how CC360 calculates the field score when one field for one of the records is null. You can configure CC360 to assign a score when one field is null so the null field does not penalize the total match score.

To configure the score, CC360 assigns a field if one of the records contains null values in the field, edit the matching settings.

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Salesforce Classic, perform the following tasks:
     1. In the upper-right corner of the page, select your name, and click **Setup**.
     2. Under **App Setup**, click **Develop** > **Custom Settings**.
   - In Lightning Experience, perform the following tasks:
     1. Click the quick access menu (), and then click **Setup**.
        The **Setup Home** page appears.
     2. Under **PLATFORM TOOLS**, click **Custom Code** > **Custom Settings**.

   The **Custom Settings** page appears.
2. Select **Manage** for **Matching Settings**.

   The **Custom Setting Matching Settings** page appears.
3. Click **Edit** for the object settings you want to edit or click **New** to create object settings.
4. On the **Matching Settings Edit** page, enter a null score value for each of the null score fields.
5. Click **Save**.

Matching and Duplicate Thresholds

You can configure the threshold that a total match score must achieve for CC360 to consider records a match or a duplicate.

When the matching batch job runs, the external data record and the master bean record must have a match score that meets or exceeds the high matching threshold. If the total match score achieves the high matching threshold, CC360 considers the external data record and the master bean record a match. CC360 then consolidates the two records when you run a consolidation batch job.

When the duplicate check batch job and duplicate contact check batch job run, records must have a match score that meets or exceeds the duplicate threshold. If the total match score achieves the duplicate threshold, CC360 considers the records as duplicates. CC360 then merges the two records when you run a merge batch job.

CC360 does not use the low matching threshold to determine matches or duplicates.

Configuring the Matching and Duplicate Thresholds

CC360 identifies duplicate records based on the matching and duplicate thresholds. You can configure the threshold that a total match score must achieve for CC360 to consider records a match or a duplicate.

To configure the matching and duplicate thresholds, edit the matching settings.

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Salesforce Classic, perform the following tasks:
     1. In the upper-right corner of the page, select your name, and click **Setup**.
     2. Under **App Setup**, click **Develop** > **Custom Settings**.
   - In Lightning Experience, perform the following tasks:
     1. Click the quick access menu (), and then click **Setup**.
        The **Setup Home** page appears.
     2. Under **PLATFORM TOOLS**, click **Custom Code** > **Custom Settings**.

   The **Custom Settings** page appears.
2. Select **Manage** for **Matching Settings**.

   The **Custom Setting Matching Settings** page appears.
3. Click **Edit** for the object settings you want to edit or click **New** to create object settings.
4. On the **Matching Settings Edit** page, enter a duplicate threshold in the duplicate threshold field.
5. Enter a matching threshold in the high matching threshold field.
6. Click **Save**.

Compare Records Page

CC360 uses a process of fuzzy matching to identify duplicate and matching records. You can select two records and compare their match score to the matching threshold and the duplicate threshold from the **Fuzzy Data Comparison** page under the **Fuzzy Matching** tab.

When you compare two records on the Compare Records page, you can verify whether CC360 considers two records a duplicate or match based on your matching settings. You can then adjust the weighting, null score, matching threshold, and duplicate threshold based on the fuzzy matching score results.

You can test CC360 match results before you run the data conversion match job. If you test your match results first, you can adjust your matching settings before you create records for external data.

The following areas appear on the Compare Records page:

Data Records

The two objects to compare. The objects can be accounts, leads, contacts, beans, or master beans.

Results of Fuzzy Matching: Summary

Displays the total score, the maximum score, the matching threshold you set, and the duplicate threshold you set. Adjust the thresholds, null scores, and the weightings to get the appropriate results.

Results of Fuzzy Matching: Details

Displays the score, weighting, and weighted score of each field.

International Settings

The international settings is a basic CC360 setting that is required to define the available billing countries and their ISO codes. CC360 uses the country names and ISO codes to standardize country information for all accounts, contacts, leads, and records from other systems.

International settings are default settings within CC360. The default settings are for the USA.

Note: If you need to modify country ISO codes, you must get a .CSV file from Informatica and use a data loading tool such as Informatica Cloud to import it into Salesforce.

You can configure the following international settings:

Name

Name for the row. The name must be unique and same as the ISO code. Default is US.

Country

The billing country name that will appear on the records. Default is USA.

Country ISO Code

The ISO code of the country. Default is US.

Currency ISO Code

ISO code of a currency. Default is US.

International Name

International name of the country. Default is United States of America.

Local Name

Local name of a country. Default is America.

Old Names

Possible other names for the country. Specify country names separated by a comma. The field has a maximum length of 255 characters. Default is USA, America, US.

Job Monitor

The job monitor logs batch statistics over a series of potentially concurrent, looped batch jobs.

You can configure CC360 to log the following job processing information:

Name

Name of the batch job. For example, Matching.

Concurrent Batch Jobs

Number of batch jobs that CC360 must process concurrently. Default is 1.

End Time

A field used by CC360 to populate the time when a job would end.

Items Currently Processing

Specifies the number of job items being processed at any time.

Job Log Id

The ID of the job log. Click on the Job Log tab to view the logs.

Job Running

Enables you to view jobs that are in progress in the Apex job monitor.

Job Status

Specifies the status of the job.

Start Time

A field used by CC360 to populate the time when a job would start.

Total Items Processed

Specifies the total number of job items that are processed.

Total Job Items

Specifies the total number of job items to be processed.

Job Schedules

Job schedule settings are used internally by CC360 to manage job schedules. You must not modify these settings.

Name

Name of the data set to use when Apex refers to the custom setting data.

Cron Expression

A string that represents the time for the scheduled job.

Cron Trigger ID

The ID of the job schedule trigger.

Legal Forms

The legal forms settings is a basic CC360 setting that is required to list all legal forms that CC360 recognizes and extracts from company names. The settings for legal forms are default and inherent within CC360.

Note: If you need to modify the list of legal forms, you must get a .CSV file from Informatica and use a data loading tool such as Informatica Cloud to import it into Salesforce.

You can configure the following legal forms settings:

Name

Name of the country for which the legal form name is defined. For example, US.

Country ISO Code

The ISO code for the country. For example, US.

Legal Form

The suffix, such as "Limited", that CC360 recognizes as the legal form in a company name. Specify a unique name. If you need to use different settings for the same legal form in different countries, use the country ISO code followed by the number sign and the legal form name. For example, use US#Limited, GB#Limited.

Search Term

Text that CC360 uses internally for a search.

Matching Settings

The matching settings define weighting for each field that the matching and duplicate check processes use.

If you do not change the matching settings, CC360 uses the default settings. If you want to update the matching settings, you can define matching settings for each object class or record type.

CC360 returns null scores when one record has a null value for a field.

You can configure the following matching settings:

Name

Name of the object class or the record type for which you want to define matching settings.

You can use one of the following name formats to define matching settings:

- For an object class, `<Object Class Name>`
  For example, Account.
- For one or more record types, `<Object Class Name>:<Record Type Name1>,<Record Type Name2>,...<Record Type NameN>`
  For example, Case:High Severity,Medium Severity indicates that the matching settings are applicable to the High Severity and Medium Severity record types.
- If you want to include the records of an object class that do not belong to any record type, `<Object Class>:,<Record Type Name1>,<Record Type Name2>,...<Record Type NameN>`
  The comma (,) after colon (:) in the name format represents the records of the object class that do not belong to any record type. For example, Account:,R1 indicates that the matching settings are applicable to the R1 record type and the account records that do not belong to any record type.

Duplicate Threshold

Threshold for being detected as a duplicate record. When the duplicate check batch job and duplicate contact check batch job run, records must have a match score that meets or exceeds the duplicate threshold. If the total match score achieves the duplicate threshold, CC360 considers the records as duplicates. Default is 90.

High Matching Threshold

The highest threshold for matching records. When the matching batch job runs, the external data record and the master bean record must have a match score that meets or exceeds the high matching threshold. If the total match score achieves the high matching threshold, CC360 considers the external data record and the master bean record a match. Default is 90.

Keep Duplicates when Re-Processing

Retains associated duplicate records even when the duplicate checked flag is reset. Default is disabled.

Low Matching Threshold

Not used.

Maximum Score

The maximum possible score. Do not populate this field. Default is no value.

Null Score Billing City

Null score for the Billing City field. Default is 100.

Null Score Billing Country

Null score for the Billing Country field. Default is 100.

Null Score Billing Postal Code

Null score for the Billing Postal Code field. Default is 100.

Null Score Billing State

Null score for the Billing State field. Default is 100.

Null Score Billing Street

Null score for the Billing Street field. Default is 100.

Null Score Birthdate

Null score for the Birthdate field. Default is 0.

Null Score Country ISO Code

Null score for the Country ISO Code field. Default is 100.

Null Score Custom Fields 1 through 6

Null score for custom fields 1 through 6.

Null Score Domain

Null score for the Domain field. Default is 0.

Null Score Email

Null score for the Email field. Default is 0.

Null Score First Name

Null score for the First Name field. Default is 100.

Null Score Last Name

Null score for the Last Name field. Default is 100.

Null Score Legal Form

Null score for the Legal Form field. Default is 100.

Null Score Mobile Phone

Null score for the Mobile Phone field. Default is 0.

Null Score Company Norm

Null score for the Company Norm field. Default is 100.

Null Score Phone

Null score for the Phone field. Default is 0.

Weighting Billing City

Weighting for the Billing City field. Default is 10.

Weighting Billing Country

Weighting for the Billing Country field. Default is 0.

Weighting Billing Postal Code

Weighting for the Billing Postal Code field. Default is 5.

Weighting Billing State

Weighting for the Billing State field. Default is 5.

Weighting Billing Street

Weighting for the Billing Street field. Default is 5.

Weighting Birthdate

Weighting for the Birthdate field. Default is 0.

Weighting Country ISO Code

Weighting of the Country ISO Code field. Weighting should be set to 200 to ensure that only accounts with the same country ISO code are detected as duplicates. Default is 200.

Weighting Custom Fields 1 through 6

Weighting for custom fields 1 through 6. The custom fields have the following behavior:

- If custom fields 1 and 2 have numeric values, CC360 performs an exact match.
- If custom fields1 and 2 have alphanumeric values, CC360 performs a fuzzy match.
- CC360 performs a fuzzy match for custom fields 3 and 4 if the weight is <=100. If the weight is >100, CC360 performs an exact match.
- CC360 performs an exact match for custom fields 5 and 6.

Weighting Domain

Weighting for the Domain field. Default is 10.

Weighting Email

Weighting for the Email field. Default is 0.

Weighting First Name

Weighting for the First Name field. Default is 0.

Weighting Last Name

Weighting for the Last Name field. Default is 0.

Weighting Legal Form

Weighting for the Legal Form field. Default is 0.

Weighting Mobile Phone

Weighting for the Mobile Phone field. Default is 0.

Weighting Company Norm

Weighting for the Company Norm field. Default is 75.

Weighting Phone

Weighting for the Phone field. Default is 10.

Enable Native Salesforce Matching

Enables CC360 to use the Salesforce Levenshtein matching algorithm.

Native Salesforce Match Threshold Level

Threshold for detecting a matching record when CC360 uses the Saleforce matching algorithm. The Saleforce algorithm might result in slightly different match scores than the CC360 matching algorithm.

Matching Settings for a Record Type

If the record type is enabled in Salesforce, you can define matching settings specific to each record type of an object class, such as account or contact .

When you run a duplicate check, duplicate contact check, or matching batch job, the batch job looks for matching settings based on the following CC360 settings:

- Use Default Settings
- Match Using Record Type Rules

The following table lists the matching settings that a duplicate check, duplicate contact check, data conversion, or matching batch job can use based on the CC360 Settings:

Match Using Record Type Rules	Use Default Settings	Batch Job Behavior
Enabled	Enabled	The batch job uses the default CC360 matching settings.
Enabled	Disabled	The batch job looks for matching settings in the following order: 1. Matching settings specific to each record type. 2. If the matching settings are not available for a record type, the batch job looks for the matching settings defined for the object class to which the record type belongs. For example, if you do not define matching settings for an account record type named Account:Personal, the batch job looks for the matching settings defined for the Account object class. 3. If the matching settings are not available for the object class to which the record type belongs, the batch job performs one of the following tasks: - If you do not have matching settings for any record type, the batch job uses the default CC360 matching settings. - If you have matching settings for at least one record type, the batch job ignores the records.
Disabled	Enabled	The batch job uses the default CC360 matching settings.
Disabled	Disabled	The batch job looks for matching settings in the following order: 1. Matching settings defined for the object class to which the record type belongs. For example, for an account record type named Account:Personal, the batch job looks for the matching settings defined for the Account object class. 2. If the matching settings are not available for the object class to which the record type belongs, the batch job uses the default CC360 matching settings.

Object Classes

The object classes settings lists all object classes with their object families, such as object class Contact with object family Company. You can use the setup wizard to configure these settings.

You can configure the following object classes settings:

Name

Name of the object class such as Account, Contact, and Lead.

Detect Duplicates from all Sources

Enables CC360 to detect duplicates during the synchronize process. Default is enabled.

Object Family

Family to which an object class belongs, such as Company.

Performance Settings

The performance settings optimize the performance of data processing for CC360. You can use the setup wizard to configure these settings.

You can configure the following performance settings:

Name

Name of the performance job. Do not change the default value. Default is DuplicateCheck.

Batch Acceleration Level

Determines the inclusiveness of the exact match on the matching segment. You can reduce the batch acceleration level to increase fuzzy matching performance. The acceleration level should be equal to the number of digits in the record count of the batch. For example, if the batch record count is 100,000, the acceleration level is 6. Default is 3.

Batch Size

Size of the batch that is processed. Default is 10.

Concurrent Jobs

Specifies the number of batch apex jobs that can be processed in parallel. CC360 can process up to 5 concurrent jobs.

Fuzzy Limit

The limit for fuzzy comparisons. Default is 200.

Job Priority

Determines the priority that a job is assigned when CC360 runs in automatic mode. Highest priority is 1. If set to 0, the job is skipped. Default is 1.

Limits Threshold

The threshold limit for the batch job. Default is 90.

Loop

Enables CC360 to repeat a job until all records are checked. Default is enabled.

Query Limit

The query limit for the batch job. Default is 1000.

Run Automatic

Schedules jobs to run whenever a user adds or modifies a record. Default is disabled.

Search Acceleration Level

Determines the acceleration level of a fuzzy search. The higher the value, the faster the fuzzy logic. Increase the acceleration to achieve faster search results. Default is 3.

Note: Higher acceleration levels may reduce the number of results returned.

Segment Limit

The segmentation limit for the batch job. Default is 20.

Segment Settings

The segment settings define custom segment structures for different object classes, such as account and contact.

Use the setup wizard to configure the following segment settings:

Name

Name of the object class or the record type for which you want to define the segment settings.

You can use one of the following name formats to define segment settings:

- For an object class, `<Object Class Name>`
  For example, Account.
- For one or more record types, `<Object Class Name>:<Record Type Name1>,<Record Type Name2>,...<Record Type NameN>`
  For example, Account:Personal,Sales indicates that the segment settings are applicable to the Personal and Sales record types.

Component Fields 1 through 5

API names of the bean fields to use to generate the segment.

Component Length 1 through 5

The number of characters to use from the API names of beans that you specify in the component fields.

You cannot define segment settings for person accounts. The person accounts use the segment settings that you define for Contact. If you use Salesforce record types, the person accounts use the segment settings of the record type that you define for Contact. If you do not define the record type-specific segment settings, the person accounts use the segment settings of the default record type that you set for Contact.

Segment Settings for a Record Type

If the record type is enabled in Salesforce, you can define segment settings specific to each record type of an object class, such as account and lead.

When you run a migration batch job, the batch job looks for the segment settings in the following order:

1. Segment settings specific to each record type.

2. If the segment settings are not available for a record type, the batch job looks for the segment settings defined for the object class to which the record type belongs.

   For example, if you do not define segment settings for an account record type named Account:Personal, the batch job looks for the segment settings defined for the Account object class.

3. If the segment settings are not available for the object class to which the record type belongs, the batch job uses the default segment settings of the object class.

Synchronization Settings

The synchronization settings define the mapping of fields from the following objects to the CC360 beans:

- Standard Salesforce objects, such as accounts, person accounts, leads, contacts, cases, and opportunities.
- Custom objects.

Use the setup wizard to configure the following synchronization settings:

Name

Name of the object or the record type for which you want to define the synchronization settings.

You can use one of the following name formats to define synchronization settings:

- For an object type, `<Object Name>`
  For example, PersonAccount.
- For one or more record types, `<Object Name>:<Record Type Name1>,<Record Type Name2>,...<Record Type NameN>`
  For example, Case:High Severity,Medium Severity indicates that the synchronization settings are applicable to the High Severity and Medium Severity record types.

City

Mapping for City field on the bean. Specify the API name of the corresponding field on the bean.

Country

Mapping for Country field on the bean. Specify the API name of the corresponding field on the bean.

Postal Code

Mapping for Postal Code field on the bean. Specify the API name of the corresponding field on the bean.

State

Mapping for State field on the bean. Specify the API name of the corresponding field on the bean.

Street

Mapping for Street field on the bean. Specify the API name of the corresponding field on the bean.

Birthdate

Mapping for Birthdate field on the bean. Specify the API name of the corresponding field on the bean.

Company

Mapping for Company field on the bean. Specify the API name of the corresponding field on the bean.

Custom Fields 1 through 20

Mapping for custom fields 1 through 20 on the bean. Specify the API name of the corresponding field on the bean.

Quick Cleansing

Accelerates the data cleansing process and reduces the number of script statements in the Salesforce triggers. Only disable during the initial data load and during migration, or if you experience problems with the order of execution of custom triggers. Default is enabled.

Domain

Mapping for Domain field on the bean. Specify the API name of the corresponding field on the bean.

Email

Mapping for Email field on the bean. Specify the API name of the corresponding field on the bean.

First Name

Mapping for First Name field on the bean. Specify the API name of the corresponding field on the bean.

Keep Data after Object Deletion

Saves a copy of the data in the data pool after deletion of the object in Salesforce. Default is disabled.

Last Name

Mapping for Last Name field on the bean. Specify the API name of the corresponding field on the bean.

Mobile Phone

Mapping for Mobile Phone field on the bean. Specify the API name of the corresponding field on the bean.

Parent ID

Mapping for Parent ID field on the bean. Specify the API name of the corresponding field on the bean.

Phone

Mapping for Phone field on the bean. Specify the API name of the corresponding field on the bean.

Record Type ID

Mapping for Record Type ID field on the bean. Specify the API name of the corresponding field on the bean.

Synchronize

Automatically synchronizes the corresponding Salesforce object such as Account, Lead, and Contact. Default is enabled.

Website

Mapping for Website field on the bean. Specify the API name of the corresponding field on the bean.

Synchronization Settings for a Record Type

If the record type is enabled in Salesforce, you can define synchronization settings specific to each record type of an object.

When you run a migration batch job, the batch job looks for the synchronization settings in the following order:

1. Synchronization settings specific to each record type.

2. If the synchronization settings are not available for a record type, the batch job looks for the synchronization settings defined for the object to which the record type belongs.

   For example, if you do not define synchronization settings for an account record type named Account:Personal, the batch job looks for the synchronization settings defined for the Account object.

3. If the synchronization settings are not available for the object to which the record type belongs, the batch job uses the default synchronization settings of the object.

Visualforce Page Fields

The Visualforce page field settings define custom fields from field sets on the CC360 Visualforce pages.

You can configure the following Visualforce page fields:

Name

Name of the data set to be used when the custom settings data is referenced by Apex.

Page Fields 1 through 5

Defines custom fields from field sets on the CC360 Visualforce pages. Specify the field names for each object separated by a comma.

Configuring Custom Settings Manually

You can configure custom settings for CC360 from the Custom Settings page without using the setup wizard.

You must log in to Salesforce before you can configure custom settings.

1. Based on the Salesforce environment that you use, perform one of the following tasks:
   - In Salesforce Classic, perform the following tasks:
     1. In the upper-right corner of the page, select your name, and click **Setup**.
     2. Under **App Setup**, click **Develop** > **Custom Settings**.
   - In Lightning Experience, perform the following tasks:
     1. Click the quick access menu ( ), and then click **Setup**.
        The **Setup Home** page appears.
     2. Under **PLATFORM TOOLS**, click **Custom Code** > **Custom Settings**.

   The **Custom Settings** page appears.
2. Click **Manage** next to the name of the CC360 setting that you want to change.

   For example, to specify custom data sources settings, click **Manage** next to Data Sources label.

   The page to manage the specific CC360 settings appears.
3. Click **Edit.**

   The settings edit page appears.
4. Specify the settings that you must configure for CC360.

5. Click **Save**.

   CC360 saves the specified custom settings.

Configuring Custom Settings Using the Setup Wizard

Use the setup wizard to simplify the implementation process and verify that the custom settings are valid. The setup wizard systematically navigates you through the CC360 configuration steps.

1. From the **Setup** section of the **CC360 Admin** tab, click **Run the Setup Wizard**.
2. To use the Setup Wizard to configure custom settings, click **Assistant** in the **Setup** section.

   The wizard navigates you through the configuration of the following custom settings:

   - Standardization settings
   - Synchronization settings
   - Data source settings
   - De-duplication settings
   - Enrichment settings
   - Profile and permission settings
   - Field set settings
   - External adapter settings
   - Customer 360 settings
   - Internal settings
3. To restore the default settings for the custom settings you can configure with the Setup Wizard, click **Use Defaults**.
4. To restore the default settings for specific custom settings, click **Set Defaults** from the appropriate Setup Wizard Assistant step.

Sharing the CC360 Settings

You can share the CC360 custom settings with Informatica when Informatica Global Customer Support troubleshoots the issues you encounter.

1. From the **Setup** section of the **CC360 Admin** tab, click **Run the Setup Wizard**.
2. Under **Phone Home**, click **Share**.

   The **List of Cloud Customer 360 Settings to Share** page appears.
3. Review the list of settings to share, and click **Agree**.

   The current CC360 settings are shared with Informatica.

APPENDIX B

Glossary

alternate hierarchy

Alternate hierarchy contains a list of nodes. These nodes link to the same master bean, nodes that have the same data source, and nodes that belong to the same hierarchy type.

automerge

Process that merges data.

bean

An object that holds data. CC360 creates a bean for every source record.

best version of the truth

The record maintained by the master bean that consists of the best data from the source records.

company norm

The company name without the legal form.

consolidation

The process of consolidating data from multiple source systems.

country name normalization

The process of standardizing the data in the country name field.

DaaS

Data as a Service (DaaS) contains address verification, phone number verification, and email address verification services. DaaS can be integrated with Cloud MDM - Customer 360 for Salesforce to verify addresses, email addresses, and phone numbers.

DaaS job

Verifies addresses, email addresses, and phone numbers of the Salesforce records.

data conversion

The process of converting beans from external data sources to Salesforce records.

de-duplication

Process of removing duplicate records.

duplicate check

A job that identifies duplicate records.

enrichment

The process of enhancing Salesforce records with consolidated data from the master beans after you run the consolidation batch job.

fuzzy matching

A probabilistic matching approach.

legal form

The business entity designation or frequently occurring element in a company name.

legal form normalization

The process of extracting the legal form from the company norm and populating the legal form field with normalized data.

master bean

Master beans link the related contact beans and lead beans to the account and also maintain the best version of the truth. CC360 creates a master bean for every account bean.

Master Org

Master Org is the central org on which you install Cloud MDM - Customer 360 for Salesforce.

multidimensional hierarchies

Contains records that are present in more than one hierarchy.

Multi-Org

Multi-Org capability automates the cleansing, standardization, de-duplication, and master data management of data from multiple Orgs in a distributed Salesforce environment.

object

A table similar to a database table that stores information.

organization

A deployment of Salesforce with a defined set of licensed users that is similar to the operational reference store of the MDM Multidomain Edition. An organization is the virtual space salesforce.com provides to an individual customer. Your organization includes all of your data and applications.

package

A group of Force.com components and apps that are made available through the AppExchange.

profile

Contains a set of permissions to perform tasks in the Cloud MDM - Customer 360 for Salesforce environment.

Spoke Org

Spoke Org connects to the Master Org to get updated accounts and person accounts.

standardization

Standardizes data that is used inconsistently, such as country names.

state and country picklists

State and country picklists are predefined and standardized states and countries. If you enable state and country picklist, you can select states and countries instead of entering state and country data into text fields.

synchronization

The process of migrating Salesforce data into Cloud MDM - Customer 360 for Salesforce.

ultimate parent

The root node of the hierarchy. The ultimate parent is the topmost node in the hierarchy.

INDEX

A

account beans
  creating 14
Account Scout tab
  about 15
accounts
  adding a hierarchy information section 91
Address Verification Settings
  Service Parameters 39
Asynchronous Cloning Process
  about 23
Asynchronous Hierarchy Process
  about 23
asynchronous hierarchy processing
  about 99
  monitoring 100
automerge
  about 64

B

batch acceleration level
  about 54, 154, 155
batch console
  about 23
batch jobs
  Asynchronous Cloning Process 23
  Asynchronous Hierarchy Process 23
  Consolidation 23
  custom filters 24
  Data Conversion 23
  Duplicate Check 23
  Duplicate Contact Check 23
  external batch job 31
  internal settings 143
  managing batch jobs 22
  Matching 23
  Merge Accounts 23
  Merge Contacts 23
  Merge Leads 23
  Migrate Accounts 23
  Migrate Contacts 23
  Migrate Leads 23
  pausing and resuming jobs 27
  priorities 25
  running a batch job 26
  scheduling a batch job 26
  Synchronous Run Automatic 26
batch queue 27
batch size
  configuring 102
batch statistics
  viewing 33
beans
  creating 14
  internal settings 146
Beans tab
  about 15

C

CC360 batch console
  about 23
Cloud MDM - Customer 360 for Salesforce
  architecture 12
  beans 14
  data cleansing settings 17
  master beans 14
  overview 11
  processes 13
Cloud MDM tab
  about 15
configuration
  custom settings 169, 170
  Customer 360 Settings 102
Consolidation batch job
  about 23
convert
  master bean to Salesforce object 78
  real time 78
custom settings
  CC360 profiles 132
  CC360 settings 135
  configuration using the setup wizard 170
  data sources 137
  enrichment settings 138
  international settings 159
  job monitor 159
  job schedules 160
  legal forms 160
  manual configuration of 169
  matching settings 161
  object classes 165
  performance settings 165
  restoring defaults 170
  segment settings 166
  synchronization settings 167
  validating settings 170
  Visualforce page fields 169

D

Data as a Service (DaaS)
  about 35
  internal settings 146
data cleansing options
  international settings 21

data cleansing options *(continued)*
legal forms 20
Data Conversion batch job
about 23
duplicate check
repeating the batch job 61
Duplicate Check batch job
about 23
duplicate checks
disabling checks between person accounts and business accounts 62
Duplicate Contact Check batch job
about 23
duplicate records
prohibiting creation of 53
Duplicates tab
about 15

E

Enrichment Parameters 39
external ID
about 14

F

fields
adding to hierarchy information section 91
internal settings 148
Fuzzy Data Comparison page
about 158
fuzzy matching
about 158
details of results 158
summary of results 158
Fuzzy Matching tab
about 15, 158
fuzzy segment
about 14

H

hierarchies
about 89
creating a large hierarchy after initial data load 95
creating a large hierarchy during initial data load 92
hierarchy type 89
internal settings 148
processing 100
troubleshooting 104, 131
ultimate parent 89
hierarchy batch size limit
configuring 102
hierarchy information section
adding fields 91
adding to accounts 91
hierarchy processing parameters
configuring 102
hierarchy scout node page size
configuring 102

I

Informatica Cloud Real Time setup
connection 109
processes 110
service connector 106
internal settings
batch jobs 143
beans 146
DaaS 146
fields 148
hierarchies 148
leads 149
Multi-Org 150
page layout 150
picklists 152
record type 151
Visualforce Scout pages 150
international settings
configuring 21

L

leads
internal settings 149
Lightning Experience
Fuzzy Matching 60, 87

M

Maintenance batch job
about 23
master bean
segment field 153
master beans
creating 14
Master Beans tab
about 15
match score weighting
about 156
match scores
about 155
matching
about 153
Matching batch job
about 23
matching segment
about 153
matching settings
about 155
configuring 59, 157
record type 164
matching threshold
about 157
maximum hierarchy limit
configuring 102
maximum trigger hierarchy limit
configuring 102
Merge Accounts batch job
about 23
merge batch job
about 64
for person accounts 64
Merge Contacts batch job
about 23

Merge Leads batch job
about 23
Migrate Accounts batch job
about 23
Migrate Contacts batch job
about 23
Migrate Leads batch job
about 23
multi-org
internal settings 150
Multidomain MDM integration
about 105
Informatica Cloud Real Time Integration setup 106
remote site settings 111
search connection 112

N

normalization
during account bean creation 14
null scores
about 157
configuring 59, 157

O

Other Duplicates tab
about 15

P

page layout
internal settings 150
person accounts
merge batch job 64
picklist
internal settings 152
settings 20

R

real time 78
record type
data sources 74, 138
record type *(continued)*
enrichment settings 77, 141
internal settings 151
matching settings 164
person accounts 134
segment settings 166
synchronization settings 168
record types 134
results of fuzzy matching
details 158
summary 158

S

segment
about 153
setup
Informatica Cloud Real Time 106
setup wizard
configuring custom settings 170
Synchronize Reporting Data batch job
about 23

T

troubleshooting
hierarchies 104, 131

V

Validation Parameters 39
Visualforce Scout pages
internal settings 150

W

weighting
about 156