salesforce

Contact Builder Implementation Guide

Salesforce, Summer '19

'19

CONTENTS

CONTACT BUILDER IMPLEMENTATION GUIDE

The Contact Builder app provides access to contact data in your account and to the relationships designed to help you use that data. Use Contact Builder to manage, consolidate, organize, and link data from all Marketing Cloud apps and external sources.

Who This Guide Is For

This guide is for Marketing Cloud admins, partners, and developers to implement Contact Builder and configure specific features.

Before You Start

Contact Builder is available in all editions of Marketing Cloud. However, Email Studio does not leverage the contact model.

Set Up Contact Builder

Review this information and follow these steps to set up Contact Builder in your Marketing Cloud account.

Contact Builder

The Contact Builder app provides access to contact data contained within your account and the relationships designed to help you use that data. Use Contact Builder to manage, consolidate, organize, and link data from all Marketing Cloud apps and external sources.

- Enterprise resource planning (ERP) systems
- Customer relationship management (CRM) systems
- Point-Of-Sale (POS) systems

Contact Builder provides a single view of a customer that displays all interactions with your brand. You can leverage all collected information to personalize customer conversations at any size in any channel. That single view of the customer can contain the mobile number necessary to send SMS messages via MobileConnect, the email address to send email messages via Email Studio, and the mobile device identification used for sending push messages via MobilePush. All of the associated address, subscription, and tracking information associated with activities and journeys relate back to that single contact record.

Contact Builder provides several tools used to manage contact data for use in building 1:1 relationships:

- Contacts Configuration - Determine how Contact Builder processes imported contact information
- Data Designer - Define information about your contacts and relate that data directly to the contact record by linking data extensions
- Data Extensions - Create and manage the data extensions that hold contact information
- Imports - Create the processes that move contact information into your data extensions
- Data Sources - Visualize where your contact data originates and assign attributes to those sources

Use Contact Builder with the Journey Builder, MobileConnect, and MobilePush apps at this time. This functionality will expand to other Marketing Cloud apps in the coming months.

Journey and Contact Data

Journey data preserves the state of a contact's data at the moment an entry event fires, facilitating the use of that data throughout a journey. Contact data captures the data values in the event source data extension at the time when Journey Builder evaluates it. Both are uniquely useful in journey creation.

Journey Data	Contact Data
Initial data value about a customer.	Current data value about a customer.
Provides attributes in the state they were in when the contact entered the journey.	Provides attributes in the state they are in when evaluation occurs after the entry event has fired.
Comprised of event data and activity data.	
Use when a contact is likely to exist in a journey more than once simultaneously.	
Use for comparison when a data value is expected to change.	

For some customer journeys, it's useful to filter against the initial data about a customer. That data value often qualifies the customer to enter the journey, even if it later changes. For example, the journey data about a customer shows that the first purchase was decaffeinated coffee. Even if the customer does not purchase it again, it is sometimes useful to filter against that initial piece of data in a journey to market a new decaffeinated flavor.

For other journeys, it is necessary to filter against the most recent customer data. For example, the contact data about a customer shows a purchase of a new regular coffee flavor. This information is useful to filter against in a journey to market a sale on that flavor.

Create a Population

Follow these steps to create a population in Marketing Cloud Contact Builder.

1. Navigate to Data Designer within Contact Builder.

2. Click the arrow next to Create Attribute Group and choose **Create Population**. You can also click **Create Population** in the Population section of the Data Designer constellation view.

3. Follow the instructions and select a data extension within Contact Builder as your new population.

   Use a data extension with a primary key value and link to the contact record on the Contact ID or Contact Key attribute. A population requires a 1:1 relationship. You cannot link any data extension set as a population to an attribute set.

Create an Attribute

Follow these steps to create an attribute in Contact Builder.

Create an attribute either directly on the contact record or within an attribute group.

1. To access data extensions you create, follow the directions in this step:

   a. Click the Data Extensions tab.

   b. To edit and click the data extension name, navigate to the data extension.

2. To access data extensions or tables created by the system, such as MobileConnect Demographics or MobilePush Demographics, follow the directions in this step:

    a. On the Data Designer page, click the attribute group containing the data extension or table to edit.

    b. Click **View & Edit**.

    c. Hover over the name of the data extension or table.

    d.
    Click the ✎ icon.

3. Click **Create Attribute**.

4. Enter a descriptive name for your attribute in the Attribute Name field.

5. Select the data type for the attribute in the Data Type drop-down menu.

6. To encrypt the data in the attribute, choose the Encrypt Data checkbox. Once you add records to this attribute, you cannot change the encryption setting.

7. For text attributes, you can select different options to mask the values presented in the text. Choose the checkbox under the Type dropdown menu and select your masking options:

- Mask all characters
- Mask all except for first word
- Mask all except for last character
- Mask all except for last 4 characters
- Mask all except for last word

8. To assign a default value to an attribute, enter that value in the Default Value field. The default value must match the data type for the attribute.

9. Select whether the attribute must contain a value for the contact using the **Required** and **Not Required** buttons.

10. Click **Complete**.

Data Extension Linking

Link data extensions in Contact Builder to relate data in existing data extensions to either the contact record or another data extension.

Start your process by linking at least one data extension to a contact record.

- Contact Builder prompts you to select a data extension to link to the contact record.
- Your available data extensions appear in the modal. Select the data extension you wish to link.
- With this step, you create your data model in an attribute group. To most effectively organize your data, select and link all data extensions that contain information about the specified topic.

Select Cardinality

When linking data extensions, you need to select the cardinality of the relationship. You can specify one-to-one, many-to-one, or many-to-many relationships. These relationships define how attributes within the data extensions relate to each other.

Link Attributes

Link attributes by selecting an attribute from the contact record and the data extension to link. You must select attributes with the same data type, but you do not have to select attributes with the same name.

Once you link at least one data extension to the contact record, link all data extensions related to the attribute group. You can link subsequent data extensions to the contact record or to other data extensions to build complex relationships.

- Hover over the link icon to view and edit relationships between data extensions.
- Hover over the slide-out menu to link and edit your data extensions. You can also remove the data extension from the data model without deleting the data extension.

You can also use events to bring behavioral information into Contact Builder. This information can help you segment and better target marketing activities toward your contacts.

Data Source Creation

Data Sources establish where the data contained within Contact Builder originates. You can assign attributes to these data sources and view precisely how data enters your account and how you can best use that data.

Contact Builder includes synchronized and custom data sources. Synchronized data sources enable access to data from other Salesforce clouds. Custom data sources help you identify other external sources of data.

Use the Data Extension screen to access and manage data extensions within Contact Builder.

Data Import

Use the Imports screen to save and reuse import definitions that bring information into Contact Builder.

You can opt to mask or encrypt data within your account to ensure compliance with corporate privacy policies and governmental regulations. Masking data prevents data from displaying in view of Marketing Cloud application users. Encrypted data functionality encrypts any data stored in your Marketing Cloud account. Contact your Marketing Cloud account representative for more information about enabling these features in your account.

Contact Builder Access

Marketing Cloud admins can restrict which users within an account can access Contact Builder.

Modify the user permissions for each user to determine their level of access:

- Access - User can access the Contact Builder app
- View All Contacts - User can view contact information within Contact Builder
- Delete Contacts - User can delete contact records from Contact Builder
- Administer Contact Data Model - User can change data relationships for attributes groups and other entities within Contact Builder

Contact Deletion in Contact Builder

Delete contacts to remove contact information from your Marketing Cloud Contact Builder account. Delete contact information to to meet data compliance regulations. This feature applies only to contacts in Email, MobileConnect, MobilePush, and Contact Builder.

Enablement

Enable Contact Delete for your account in the Contact Configuration section of Contact Builder. For Enterprise 2.0 accounts, enablement occurs at the parent account level for the entire organization.

When enabled, this feature operates differently for different types of tenants.

- Enterprise 2.0 accounts require deletions to originate from the parent account. However, child business units can include contact records created by triggered sends, data operations such as queries or filters, or synchronized data sources. Delete those records using the child business unit MID in your request.
- Lock-and-Publish accounts isolate deletions to that account, no matter which accounts exist in the larger structure.
- On-Your-Behalf accounts require deletions to originate in the parent account and cascade to child accounts.
- Agency and Agency Client accounts isolate deletions to that account, no matter which accounts exist in the larger structure.

All contact deletion processes are final, and Marketing Cloud cannot restore deleted contact information.

Process

This functionality includes a 14-day default period where Contact Builder suppresses contact information from showing in channel apps such as Email or MobileConnect. You can change the length of this time period in your account. The information remains in your account, but you cannot view contact records or access it for send activities. Once the period elapses, we delete contact information for deleted contacts from lists and sendable data extensions. This process does not apply to any non-sendable data extensions not included in a population via Data Designer, and you are responsible for deleting applicable information from those sources on your own.

1. Deletion process initiated
2. Suppression period - contact information remains in your account, but you cannot view or access it
3. Final deletion - Marketing Cloud removes deleted contact information from your account

MobileConnect and MobilePush Contact Deletion

By deleting this contact, you are also deleting all opt-out preferences associated with that contact. As a result, if you reintroduce this contact into Marketing Cloud, we do not automatically honor any previous opt-out preferences stored for this contact. If you re-import

the contact, you must also re-import the contact's opt-out status or that information is lost. Ensure that you receive express permission to send messages to an individual before you import or re-import any contact into Marketing Cloud. Honor the individual's prior opt-outs, and confirm that your messaging complies with all applicable laws.

Synchronized Data Sources Contact Deletion

For Synchronized Data Sources, delete the information from the original data source in Sales, Service, or another cloud. Allow that change to sync to Marketing Cloud.

Contact Builder Use Case

This use case demonstrates an effective process to set up Contact Builder in your Marketing Cloud account.

Use Contact Data in a Journey

To use contact data in a Marketing Cloud journey, create a population and link an orders data extension.

This example uses a master subscriber data extension and an orders data extension that is common in retail settings. The master subscriber data extension contains contact information, and the orders data extension contains information related to previous and current orders made by those contacts. Both data extensions contain a CustomerID field used to identify contacts.

1. In Marketing Cloud, click **Audience Builder** and select **Contact Builder**.

2. Click **Data Designer**.

3. Click **Populations**.

4. Click **Create Population**.

5. Select the subscriber master data extension.

6. Relate the SubscriberKey field to your ContactKey field, and click **Create**.

7. Click **Data Designer**.

8. Click **Create Attribute Group**.

9. Choose a name and icon for your attribute group.

10. Click **Create**.

11. Click **Link Data Extensions**.

12. Select the population you created for the first entity. Select the orders data extension for the second entity.

13. Link the CustomerID field in the data extensions.

14. Click **Save**.

Use this data model to build out journeys and execute decision splits.

Contact Builder Best Practices

Contact Builder helps you maintain the contact information you use to identify individual contacts across multiple channels and personalize messaging and marketing activities. To ensure that your marketing activities function, make sure that all contact information relates to a single contact record.

Use a Single Contact Key

Contact Builder uses a single contact key value to identify a contact within a Marketing Cloud account. This value is identical to subscriber key in Email Studio. Use the contact key value to assign information from several communication channels, such as mobile numbers and email addresses, to a single contact record. Follow these best practices to implement a unique contact key for each contact:

- Use a unique value that does not relate to a specific channel. If you import a value from an external data source, use a string-based field, even if the external data source uses a number as the unique identifier. If you use your Marketing Cloud account to gather data, consider using a GUID or other generated ID in a string field as the contact key value.

- Use a contact key value that does not change based on customer behavior. For example, a physical address makes a poor contact key value, as it can potentially apply to several people or change frequently.

- To maintain consistent contact information and implementation, use the subscriber key value in Email Studio as your unique contact key value in Contact Builder.

- When taking your information from an external information source, use the unique identifier from that system as your contact key to ensure consistent data imports. If your unique customer identifier is a numerical value, use string-based field for the contact key when you create the data extension with information from the external data source.

- Contact key values cannot contain these characters.

    - ~
    - !
    - @
    - #
    - $
    - %
    - ^
    - &
    - *
    - (
    -)
    - -
    - +
    - }
    - {
    - /
    - \

Find Consistent Information to Tie Potential Duplicate Contact Records Together

You cannot merge or de-duplicate contact information in Marketing Cloud. Therefore, you can see a contact that is represented twice within the system. For example, a contact includes an email address in Email Studio and a separate device ID in MobilePush. Use another demographic value to help de-duplicate this contact via an external service or partner. For example, you would use another demographic value if there are two different records with the same first name, last name, and physical address values likely represent the same contact. Elimination of duplicate records keeps your contact count lower and reduces billing costs and maintains the integrity of a unique contact record for marketing activities.

Understand the Difference Between All Subscribers and All Contacts Lists

The All Contacts list includes all contacts in a Marketing Cloud instance. The All Subscribers list specifically refers to subscriptions created within Email Studio. Email Studio references the All Subscribers list to prevent sends to unsubscribed email addresses. However, this action applies only to Email Studio information and activities. A contact can unsubscribe from a subscription in Email Studio and still receive messages from other apps, such as MobileConnect or MobilePush.

By default, Contact Builder uses a different demographic source of information for each channel. The different demographic attribute groups displayed in Contact Builder contain this information. However, the unique contact key value used to identify a specific contact remains consistent for all channel apps. To use a single set of demographic information for all channels, create an attribute group in Contact Builder. Use that group as the source of information for your marketing activities. Make that attribute group available to all necessary users and copy to other business units within your account.

Understand the Contact Record Deletion Process

When possible, do not delete contacts to ensure your marketing activities honor all applicable unsubscriptions. While you can delete contacts related to Email Studio from your Marketing Cloud account, this deletion does not totally remove their information from Marketing Cloud. We retain this information to ensure that the appropriate email addresses do not receive messages from which they unsubscribed.

Change ID Number Data Type Before Linking to Contact Key

If you use a numeric value to identify individual contacts, change that data type to a string data type before linking that value to Contact Key. You cannot directly link numeric values to Contact Key. For best performance, change the data type for your numeric identifiers before importing contact information into Contact Builder. Do not link numeric identifiers and the CustomerID value, as this activity can cause performance issues.

Link Attribute Groups and Populations Using Contact Key When Possible

Link attribute groups and populations using the contact key value. Don't link using an email address field when the Contact Key or Subscriber Key value is available. If you must create a link using the email address, create a text attribute containing the email address and link using that value.

Understand the Differences Between Contact ID and Subscriber ID versus Contact Key and Subscriber Key

These system fields do not serve the same purpose in a Marketing Cloud account.

Most customers use a primary identifier for their customers and link their identifier to Marketing Cloud using the Marketing Cloud Contact Key or Subscriber Key value.

The Contact ID and Subscriber ID fields act as surrogate keys used by internal Marketing Cloud systems. You can see these fields in tracking extracts and system data views in Query Activities. To avoid performance issues, don't use these values for Marketing Cloud activities.

Limit Populations to Three Entities

Use populations to create large and distinct subgroups of your contacts, then segment contact records from there. For example, a doctor's office creates populations for staff, for patients, and for vendors. If you create populations in your account, limit those populations to three or fewer to improve performance.

Contact Delete Best Practices

Review this information and apply these best practices to your Contact Delete implementation. All contact deletion processes are final, and Marketing Cloud cannot restore deleted contact information.

- Contact Delete removes contact information across the entire enterprise for Enterprise 2.0 accounts.
  - This deletion affects all business units that share the same ClientID value as the MID of the account that initiated the delete.
  - If you use multiple Enterprise 2.0 accounts, perform the Contact Delete process for each account. The deletion process requires more time to remove contact information across a large number of business units.
  - Child business units can include contact records created by triggered sends, data operations such as queries or filters, or synchronized data sources. Delete those records using the child business unit MID in your request.

- Use ContactKey or ContactID values to delete contacts.

- If you linked multiple SubscriberKey or ContactKey values to a contact record, perform the Contact Delete process for all values.

- To request the status for a single contact, use an API request. For large batches of deletions, the suppression status shows for contacts when the entire batch processes.

- Data extension data-retention policies do not ensure regulatory compliance. Use both Contact Delete and data retention policies to help ensure full removal of data.

- We recommend setting data retention policies for a data extension when you first create it. You can apply data retention policies to new and existing data extensions. However, data extensions with more than 100 million records cannot accept retroactive data retention policies.

- If you set the suppression period length to any value except 0, that contact can't reintroduce their information by opting back in until the suppression period expires. Set the value to 0 to delete immediately and allow transactional senders to opt back in as necessary.

- To more quickly remove contact information from your account, set the suppression period length to 0.

- The deletion process yields to other account activities, such as sending, imports, automations, and query activities. These processes take precedence over deletions. As more activities run in the account, the deletions require more time.

- The deletion process scans all sendable data extensions in an account to remove the specified contact records. To improve the speed of this process, delete any unnecessary sendable data extensions in your account.

- When you delete a contact from a list or data extension, export any list or data extension to a file beforehand so that you can restart the contact delete process in the event of an error.

- When deleting a large number of contacts, the delete process can contain no more than one million records.

Best Practices for Data Extensions and Query Activities

Review this information to implement best practices for your data extensions and query activities. Choosing the proper field length and data type ensures efficient storage and retrieval of data in your data extensions.

Set a Data Extension Naming Convention

A standard naming protocol for your data extensions helps you quickly find and remove data you no longer need.

Share Data Extensions When Possible

If your tenant includes multiple business units that use the same data, share data extensions from the top-level business unit to avoid duplicate data. This process also prevents business units from using out-of-date information.

Store Frequently Changed Data in a Separate Data Extension

To prevent large imports to large data extensions and improve performance, store frequently changing data in a separate data extension. Then link that data extension to other data extensions containing more static data. Limit updates to 1 million records or fewer, and use batching when available. Schedule updates and automations to avoid running at the same time or when you make requests for that data.

Choose the Correct Data Field Types

There are several different data types within data extensions. Choose the correct data field types for your data extension to ensure that your data is efficiently stored and can be quickly retrieved.

For example, use the Phone data type when storing phone numbers as this data type does not allocate space for double byte character sets (DBCS) and thus utilizes less storage.

Avoid the use of nvarcharmax fields. More precise fields allow the application to better retrieve and process data. Only use nvarcharmax fields when absolutely no other choice is available. Use of this data type significantly decreases email send performance and is not recommended as a best practice.

Set Appropriate Field Lengths

Optimize your data extensions by precisely stating the number of characters that can be used in that field. Doing so reduces the amount of work necessary to store and retrieve data from your data extensions. Precisely constructed data extensions help to avoid inefficient data storage which can degrade overall performance. Only use unbounded columns, those fields whose length is not set, when no other option is available. For example, you can store large HTML blocks that require unbounded columns. However, make every effort to set appropriate field lengths. Limit the total number of field lengths in a data extension to less than 4000 total chacters.

For example, set the Text field length of the data extension equal to 100 characters if you import data into a Text field where the length of the source data never exceeds 100 characters. This step ensures that data extension is optimized by not unnecessarily allocating unused space that could later lead to poor performance.

Eliminate Deleted Data Columns

When a data extension column is deleted, the system retains it as a soft delete. Having a number of deleted columns within a data extension can cause performance to degrade over a period of time. If you notice performance issues in a data extension in which several columns have been deleted, rebuild the data extension by creating a new data extension and transferring the data from the old data extension to the new one with a query activity. When you rebuild the data extension, be sure to assign the correct field lengths and datatypes to the fields you create in the data extension.

Use data retention to specify the amount of time your account retains data and data extensions. This automatic process helps eliminate unneeded data in your account without the need to manually address the deletion.

Avoid Data Type Mismatches for Query Activities

The amount of work the application must perform increases greatly when it must join or transfer data between two fields with different data types. If you plan on frequently joining or moving data from one data extension to another, make sure the data types are the same to prevent performance issues. If the data extensions already exist, you can create a new data extension with the correct data types and

move the data from the old data extension to the new data extension to eliminate the issue. Move the data by exporting it from the old data extension and importing it into the new data extension.

Build Precise and Searchable Query Activity Arguments

Searchable arguments can make use of available data extension indexes to search more quickly and efficiently for your query results. Write your queries to ask for a specific piece of information instead of writing it to exclude large sections of data that must be searched. Ensure that you write your queries to limit the number of times you must interact with large groups of data and execute repetitive queries on smaller batches of data.

For example, you may get the same results you want with the first query in this section, but that query uses a function and causes the process to run slower.

Inefficient Query with Intrinsic Function:

```
WHERE SUBSTRING (lastname,1,1) = 'h'
```

If you are searching for somebody's last name with the first letter of H, write this query.

Efficient Query:

```
WHERE lastname LIKE 'h%'
```

Avoid Non-Searchable Operators for the WHERE Clause

Avoid using the following operators in the WHERE clause:

- OR operators
- NOT
- <
- >
- NOT EXISTS
- NOT IN
- NOT LIKE
- Intrinsic functions

These operators cannot take advantage of indexes and cause the query activity to run more slowly.

More efficient operators improve your queries, such as = or AND.

The sample query could be rewritten to avoid the non-searchable (non-SARGable) operator.

Inefficient Query with Non-Searchable Operators:

```
WHERE NOT age > 18 AND age <18
```

Instead, write this query:

Efficient Query:

```
WHERE age = 18
```

Use SELECT DISTINCT queries only when duplicate data exists in multiple data extensions. The need to use this query lessens when all data extensions involved use precise, unique data.

> 📝 **Note:** To ensure queries are executed without blocking from other SQL updates, query activities use SET TRANSACTION ISOLATION LEVEL READ UNCOMMITTED which is equivalent to WITH (NOLOCK). Using the NOLOCK query hint is unnecessary.

When using, SELECT TOP queries, specify the ORDER BY command to avoid unpredictable results. Due to caching algorithms, the data the query activity pulls back may not be the expected data. By specifying a precise order, the query activity returns the data as expected.

Explicitly specify the column names you need in your select clause opposed to using Select *. Using select * actually causes the Query Activity parser to look at the field names and snapshot the fields at the time the query is saved. In the future, if you add a new field to your data extension, it does not get automatically included

Inefficient Query

```
select * from DE
```

Efficient Query

```
select field1, field2, field3 from DE opposed to select * from DE
```

Optimize Your Data Views

For queries involving data views, implement with a 6-month cap. Often the performance is dramatically better limited to no more than 30 days.

Think about the impact on the duration of that transaction, amount of transaction log growth, auto-kill, rollback, and extended database recovery times. A typical pattern works great in the beginning and as the amount of data grows so does the times to execute until AutoKill gets involved. Break the work into pieces, if possible, or use an AltText Query Activity and implement with small, committed batches.

Limit your update sets to one million records and batch through them. The system creates a transaction out of an add and update.

You can require about assistance or a service engagement when dealing with the large amounts of data present in a typical Marketing Cloud account. Contact your Marketing Cloud account representative if you have questions or concerns about dealing with large amounts of data or optimizing your data extensions and queries.