CX Contact Help

Import Specification Files

12/16/2025

Contents

Import Specification Files

> ### Important
> The content of this document has been moved and is no longer being updated in this location.
> For the latest content and most recent updates, see the CX Contact Help on the Genesys Multicloud site.

If you don't want to apply a data mapping schema to a contact list being imported into CX Contact, you can use a specification file.

An input specification file (*.spc) is a text file used to identify the location of the fields in your data that are needed to populate your contact list. For example, if the First Name field starts at the seventh position in your data format, that information is included in the input specification file. Using the specification file, you can selectively extract some of the fields from your data while ignoring the ones you don't need. Here's an example:

```
# 5th Column (51st to 60th spot)
homePhone:s/^.{50}(\d{10})/$1/

# Columns 1, 2 and 3 to ClientID
ClientID:s/^(.{26})/$1/

# Work Phone
workPhone:s/^.{120}(\d{10})/$1/

# First Name to FirstName
firstName:s/^(.{60})/$1/

# Last Name to LastName
lastName:s/^.{60}(.{60})/$1/
```

The topics covered on this page are as follows:

- Import a specification file
- Specification file keywords
- Specification files and contact list database tables

Import a Specification File

Once you have created the specification file, use the following steps to import the file:

1. On the Specification Files page, click **New**.
2. In the Type field, leave the **Input** option.
3. Click **Select File** to locate the file on your PC or network. Input specification files must be text files with an spc extension.
4. Specify a name (required) and description (optional) for the specification file.
5. Click **Import Specification**.

> ### Important
>
> If you attempt to upload a specification file that is not formatted correctly, the system will return an error and the file will fail to upload.

All specification files appear in a table on the main Specification Files page. The table contains the following information for each specification file:

Name	The name of the specification file.
Description	A description of the specification file (if you provided one.
Type	The type of specification file. The default is Input.
Last Modified Date	The date the specification file was last modified.
Created Date	The date the specification file was initially imported.

Once you import a specification file, you can use the tools in the Actions menu to do any of the following:

Edit	Click the pencil icon to edit any of the specification file's properties.
Download	Click the download icon to download a specification file.
Delete	Click the trash can icon to delete a specification file.

Specification File Keywords

The following table lists specification file keywords and the corresponding contact list fields they map to.

Keyword	Contact List Field
• fname • first name • firstname	c_first_name
• lname • last name • lastname • name	c_last_name
• company	c_company
• other1-otherN	c_other1 (applies to other1-otherN)
• clientid • client id	c_client_id
• tz • time_zone • timezone	c_tz_uid
• zip • zip_code • zip code • postal_code • postal code	c_postal_code
• country • country_code • country code	c_country_code_iso
• state • region	c_state_code

Specification Files and Contact List Database Tables

If you use a specification file before uploading contact data into CX Contact and you use user-defined fields in the specification file, CX Contact will store the user-defined data for fields Other1-Other20 in the main calling list table, and it will store user-defined fields from Other21 and onward in the secondary table.

Here is an example specification file:

```
FirstName:s/<CSV>/$1/
LastName:s/<CSV>/$2/
ClientID:s/<CSV>/$3/
HomePhone:s/<CSV>/$4/
WorkPhone:s/<CSV>/$5/
CellPhone:s/<CSV>/$6/
Other1:s/<CSV>/$7/
Other2:s/<CSV>/$8/
Other3:s/<CSV>/$9/
Other4:s/<CSV>/$(10)/
Other25:s/<CSV>/$(11)/
Other28:s/<CSV>/$(12)/
```

That yields the following input file:

```
"John","Snow","540-18-4455","+16504661200","+19252381988","+14159873456","O1value","O2value","O3value","O4value
```

Now here are the contact list tables that are created from this file:

```
Resulting calling list tables:

outbound_22_30=> select record_id, c_last_name, c_first_name, c_client_id, contact_info, c_other1, c_other2, c_other3, c_other4, chain_id, chain_n from cc_list_153 order by 1;
 record_id | c_last_name | c_first_name | c_client_id | contact_info  | c_other1 | c_other2 | c_other3 | c_other4 | chain_id | chain_n
-----------+-------------+--------------+-------------+---------------+----------+----------+----------+----------+----------+---------
         1 | Snow        | John         | 540-18-4455 | +16504661200  | O1value  | O2value  | O3value  | O4value  |        1 |       0
         2 | Snow        | John         | 540-18-4455 | +19252381988  | O1value  | O2value  | O3value  | O4value  |        1 |       1
         3 | Snow        | John         | 540-18-4455 | +14159873456  | O1value  | O2value  | O3value  | O4value  |        1 |       2
(3 rows)


outbound_22_30=> select * from cc_list_153_ud;
 ud_id | ud_chain_id | ud_key  | ud_value
-------+-------------+---------+----------
     1 |           1 | Other25 | O25value
     2 |           1 | Other28 | O28value
(2 rows)
```

Related Topics

- Create a Data Mapping Schema