Webex WFO Data Import/Export Reference Guide

For Deployments with New WFM

First Published: July 20, 2021
Last Updated: October 04, 2024

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0882

Contents

Introduction

You can import and integrate historical and real-time data from any automatic contact distributor (ACD) to the Webex WFO database using APIs. This document describes how to use these APIs to do the following:

- Import contacts in bulk
- Import post-call survey IVR data
- Import and sync file-based data

Import contacts in bulk

Third parties use the Bulk Contact Import APIs to merge and insert metadata and recordings in a multi-part HTTPS request.

There are two Bulk Import APIs:

- Bulk Contact Import API—Used by third parties; allows insertion of both metadata and recordings in a multi-part HTTPS request

  NOTE You can also use the Bulk Contact Import API to import contacts in bulk through the Data Server. See Using the Data Server for bulk contact import.

- Real-time Contact API—Used by Webex WFO Smart Desktop recording client; inserts metadata first and then uploads recordings separately later, based on the response from the metadata insertion. This allows the client to delay uploading recordings and allows contacts to be inserted while the call is still in progress. However, this forces the client to track IDs to upload the recordings for the correct contact.

Protocol and URI

Bulk contact import API

URI	`/api/upload/contacts`
Method	POST
Permissions	Administer Tenant
Content Type	multipart/form-data

Real-time bulk contact import API (metadata)

URI	`/api/rest/wfo/contact/import`

Method	POST
Permissions	Capture Contacts, Record Voice/Record Screen
Content Type	application/JSON

Supported formats

The following formats can be included in a multipart request.

Type	Description
CSV	A comma-separated file used to assign metadata.
JSON	The same metadata format as the Real-time API, but can be used for bulk import.
WAV	An audio recording format.
WEBM	A combined audio and video recording format.
WEBMA	An audio-only (WebM container) format.
WEBMV	A video-only (WebM container) format.
SPX	An audio format.
WMV	A combined audio and video recording format (or video only if paired with audio in the same contact).
OPUS	An audio format.

Request and response fields

The CSV and JSON files include fields defined in the following table. Not all fields are used in both types of files. The file the field applies to is indicated in the description.

Name	Req?	Description
AgentId	Y	Used in CSV and JSON. The Agent ID in one of three formats. Processing figures out

Name	Req?	Description
		which format is used based on parsing the contents. ■ Person ID. A unique identifier from WfoPerson.id. This number is also used in the User Export spreadsheet (Application Management > Global > Users > Import and Export > Export > User ID column). It is not the same as the ID in the Webex WFO user profile. ■ AD Login. A domain\username (requires "\"). ■ Email address. An email address (requires "@"). When using a CSV to upload contacts, the agent ID is required. If you are using JSON to make the request, the agent ID is optional. In the latter case, the agent ID is set to the ID of the authenticated user initiating the upload. Max characters = 254 Default = none
AssocCallId	N	Used in CSV and JSON. An ID that ties contacts together. For example, a transferred call from one agent to another each have the same ID. Max characters = 52 Default = NULL
Audio.Location	N	Used in CSV and JSON. In the Audio sub-object. The key (file) name of the recording in the multipart request. This can be any supported recording format (audio/screen/combined). Only a single audio file per contact is allowed. The key name must have a valid extension that matches the media type of the recording. The extension identifies the file as an audio or screen recording, or both. Max characters = 128 Default = None

Name	Req?	Description
Audio.StartTimeMs	N	Used in CSV and JSON. In the Audio sub-object. The start time in milliseconds GMT since 1970-01-01 (UNIX time) of the audio recording. This is used to determine the audio offset from when the contact starts. Max characters = long Default = ContactStartTimeMs
CalledAddress	N	Used in CSV. The called phone number. Max characters = 64 Default = Empty string
Called	N	Used in JSON. The called phone number. Max characters = 64 Default = NULL
CallId	N	Used in CSV and JSON. An ID that identifies a contact. Max characters = 128 Default = NULL
Calling	N	Used in JSON. The calling phone number. Max characters = 64 Default = NULL
CallingAddress	N	Used in CSV. The calling phone number. Max characters = 64

Name	Req?	Description
		Default = Empty string
ClientTimeZone	N	Used in CSV and JSON. The time zone in UTC format. Windows Time is also supported. The Desktop Recording client sends Windows Time, which is mapped to Olson time. Max characters = 255 Default = Customer's time zone as defined in Webex WFO **EXAMPLE** –06:00
ContactStartTimeMs	N	Used in CSV and JSON. The start time in milliseconds GMT since 1970-01-01 (UNIX time). A value in this field is required, so if the value is missing, the API uses the current upload time. Note that this likely results in a poor user experience, with many contacts that have the same timestamp. **IMPORTANT** If you are importing contacts with Excel, you must format the Start Time column to display milliseconds (consult the Excel user documentation for more information). Otherwise, Excel truncates milliseconds, resulting in a false time and preventing recordings from importing correctly. Max characters = long Default = current upload time **EXAMPLE** 1447100000000 - 11/09/2015 20:13:20 GMT
Direction	N	Used in CSV and JSON. The direction of the call, inbound or outbound. 1 = outbound 0 = inbound

Name	Req?	Description
		Max characters = 1 Default = NULL
Line	N	Used in CSV and JSON. The agent's line/extension. Max characters = 64 Default = NULL
metadata.<custom metadata field name>	N	Used in CSV. The custom metadata fields to populate. The field will be created if it does not exist. Any column beginning with "metadata" will be treated as a custom metadata field. **EXAMPLE** To set "accountNumber", create a column named "metadata.accountNumber". Max characters field name = 39 Max characters of custom metadata value = 2056
CustomMetadata	N	Used in JSON. The custom metadata fields to populate. The field will be created if it does not exist. The object contains data in the form of name/value pairs. **EXAMPLE** "accountNumber":"123456" Max characters field name = 39 Max characters of custom metadata value = 2056
Recording1	Y	Used in CSV. The key (file) name of the recording in the multipart request. This can be any supported recording format (audio/screen/combined). Only a single audio file per contact is allowed. The key name must have a valid extension that matches the media type of the recording. The extension

Name	Req?	Description
		identifies the file as an audio or screen recording, or both. Max characters = 128 Default = None
Recording2	N	Used in CSV. The key (file) name of the recording in the multipart request. This can be any supported recording format (audio/screen/combined). Only a single audio file per contact is allowed. The key name must have a valid extension that matches the media type of the recording. The extension identifies the file as an audio or screen recording, or both. Max characters = 128 Default = None
Recording3	N	Used in CSV. The key (file) name of the recording in the multipart request. This can be any supported recording format (audio/screen/combined). Only a single audio file per contact is allowed. The key name must have a valid extension that matches the media type of the recording. The extension identifies the file as an audio or screen recording, or both. **NOTE** There must be an audio file or the import fails. Max characters = 128 Default = None
Recording2Offset	N	Used in CSV. The offset of Recording2 from Recording1. **EXAMPLE** An audio file (WAV) that starts 5 seconds after the screen file (WEBM) has an offset of 5000 (5000 = 5 seconds). Max characters = Long

Name	Req?	Description
		Default = 0
Screen.Location	N	Used in CSV and JSON. In the Screen sub-object. The key (file) name of the recording in the multipart request. This can be any supported recording format (audio/screen/combined). Only a single audio file per contact is allowed. The key name must have a valid extension that matches the media type of the recording. The extension identifies the file as an audio or screen recording, or both. Max characters = 128 Default = None
Screen.StartTimeMs	N	Used in CSV and JSON. In the Screen sub-object. The start time in milliseconds GMT since 1970-01-01 (UNIX time) of the screen recording. This is used to figure out the screen offset from when the contact starts. Max characters = long Default = ContactStartTimeMs

CSV file examples

CSV can be uploaded as part of a multipart upload request. Some rules regarding the format are as follows.

- The number of columns is variable. For example, if you always want to use the customer's time zone, you do not have to include the TimeZone column in the CSV.
- The columns included in the CSV can be in any order.
- The number of columns in each row must match the number of header columns.
- If a value has a comma, it must be surrounded by quotes.
- If a value is not known for a specific contact, but the header exists, use a empty string for that column.

> **NOTE** A successful response is formatted in JSON, because the response adds some status for each contact and recording.

Full example

This example shows a file that uses every field possible for a CSV file.

```
AgentId,ContactStartTimeMs,TimeZone,AssocCallId,CallId,CalledAddress,Line,CallingAddress,Direction,Recording1,Recording2,Recording2Offset,metadata.accountNumber
abc/bunkowm,1447100000000,America/Chicago,103585664793210000,30611848,1801,1800,1800,1,call1.webmv,call1.wav,5000,1234567890
mark.bunkowske@abc.com,1447110000000,America/Chicago,103585664793220000,30611848,1801,1800,1800,1,call2.wav,,,987654321
```

Short example

This example shows only the fields required for a CSV file.

```
AgentId,ContactStartTimeMs,Recording1
2,1447100000000,call1.wav
2,1447110000000,call2.wav
```

JSON file examples

Contact information can be imported into Webex WFO in JSON format as an alternative to CSV format.

Full Example

The following is an example of a formatted JSON file.

```
{
   "AgentId":"john.smith@acme.com",
   "AssocCallId":"103585664793254280",
   "CallId":"30611848",
   "CalledAddress":"1801",
   "CallingAddress":"1800",
   "ClientTimeZone":"Central Standard Time",
   "ContactStartTimeMs":1447075073000,
   "Direction":1,
   "Audio":[
      {
         "Location":"25.wav",
         "StartTimeMs":1447075080000
      }
   ],
   "Screen":[
         "Location":"25.webm",
         "StartTimeMs":1447075075000
   ],
   "CustomMetadata":{
         "accountNumber":"123456",
```

```
            "department":"sales"
        }
    }
```

Short example

This example shows only the fields required for a JSON file.

```
{
   "AgentId":"acme\smithj",
   "ContactStartTimeMs":1447075073000,
   "Audio":[
      {
         "Location":"25.wav"
      }
   ]
}
```

ZIP format

The ZIP format is handled differently than CSV or JSON, in that it is a collection of files that are processed as if they were individual files within the multipart request.

- The name of the file is the key that needs to be referenced in the CSV/JSON.
- Any folder structure in the ZIP file is flattened and ignored.

For example, a multipart request looks like the following.

```
batch.zip
   batch.csv (contains 2 rows, for call1 and call2)
   call1.wav
   call2.wav
```

This multipart request is processed as if the files were all in the ZIP or all individually in the multipart request.

Notes

- The order of files does not matter.
- An upload for a contact that contains a recording file name but does not include that recording will fail to be inserted.

- An upload that contains a recording the is not referenced in a CSV or JSON will ignore that recording.

IMPORTANT You must have Tenant Administrator access, access to the data server to use for the Bulk Contact Import, and the Bulk Import permission checked for your role.

Using the Data Server for bulk contact import

You can use the Bulk Contact Import API to upload contacts in bulk through the Data Server.

Using the Bulk Contact Import API requires the following:

- Webex WFO Administrator role with the Bulk Import permission enabled.
- Read/write access to the Data Server.

To upload contacts in bulk through the Data Server:

PREREQUISITE For a bulk import to successfully upload files using a data server, the data server must be configured within Webex WFO. This can be a Data Server that is already being used for any other purpose or a new Data Server. If you are configuring a new Data Server for bulk import, see the topic, “Data Server Configuration” in the *Webex WFO User Guide*.

1. To use a data server that is already configured in Webex WFO navigate to the **Data Server Configuration** page (Application Management > System Configuration > Data Server Configuration) and select that data server.
2. In the **Regional Data Server ACD Sync Settings** section, ensure that **Enable Capture** is selected and that the **Generic (Default)** ACD is assigned.
3. Click **Save**.
4. Create a CSV (not JSON) file that contains all required fields, plus any optional ones that you want to add.
5. Prefix the file with the word CONTACT. The word is case-sensitive, and you must type it in upper-case.

   EXAMPLE
   Your CSV file is named ExampleContacts.csv. You must rename it with the CONTACT prefix as follows:

   CONTACT.ExampleContacts.csv

6. Place the CSV file and all associated media files in the GIS <tenant> folder on the Data Server. This folder is in the location defined by the Regional Data Server GIS File Location field on the Data Server Configuration page.

   EXAMPLE C:\Program Files\Common Files\Webex WFO\Data Server\gis\<tenant>

Import post-call survey IVR data

Data that is collected from post-call surveys via an IVR can be imported into Webex WFO using a generic IVR integration that uses CSV files saved to a specific folder on the Data Server. See “Add Post-Call Surveys to Contacts” in the *Webex WFO User Guide* for more information about configuration.

> **NOTE** This folder location is configured in Webex WFO on the Data Server Configuration page (Application Management > Global > System Configuration > Data Server Configuration) in the Regional Data Server GIS File Location field.

The File Observer service triggers the import of these data files into the database and attaches the data to contact recordings using the Contact ID, Associated Contact ID, or the ICM Call ID.

Two CSV files are required:

- The **Form** CSV file contains the survey questions and must be processed first.
- The actual survey results are imported through the **Results** CSV file.

Form CSV file

The Form file name must follow the format **Form_<Form ID>.csv**, where <Form ID> is a number.

The Form CSV file is formatted to contain the following information:

```
<form name>,<form status>,<form date>,<total score>
1,DIGITS,"Contact Identifier",0
<question number>,<question type>,<question text>,<question response and weight>
```

The first row in the Form CSV file contains the following information:

Field	Description
form name	The name of the survey form.
form status	The form’s status can be **editable** or **active**. Editable forms can be modified with another import. With editable forms existing result data is

Field	Description
	deleted before updating the form details. Active forms cannot be changed.
form date	The form's date, in yyyy-mm-dd format.
total score	The total score possible in the survey.

The second row in the Form CSV file is required to have question ID 1 and is always the first question in any VR survey form. It is a placeholder for the contact identifier that is supplied in IVR survey results files (see Results CSV file):

```
1,DIGITS,"Contact Identifier",0
```

The third and all subsequent rows in the file contain the survey questions:

Field	Description
question number	The number assigned to the survey question. **IMPORTANT** Question number cannot be 1.
question type	The type of question. **NOTE** Only **OPTION** type questions (an answer on a scale, for example from 1 to 5) can have results saved and a survey must have at least one **OPTION** type question to be associated with a contact.
question text	The survey question.
question responses and weights	The question response and weight is a comma-separated array in the format <option id> - <text for result> - <value/weight>. Where ■ <option id> is an ID for the option that is unique in the scope of the question. ■ <text for result> is the text used to identify this option in a survey result line. ■ <value/weight> is the value of the question option.

Results CSV file

Each Results file is an output snapshot from the generic IVR system. For example, the IVR can be configured to export one file every 30 minutes and include all surveys taken within the last 30-minute interval. If a form has two questions, then each survey response file will have three lines per survey:

- Line 1 identifies the contact ID to associate with the survey answers.
- Lines 2–3 contain the survey answers for each question.

The Results file name must follow the format **Results_<yyyyMMdd>_<HHMM>_<unique ID>.csv** where <unique ID> is a value that makes sure that the file name is unique. It can be based on timestamp, agent ID, or a generic sequential increment.

The following is the format of each row in the Results CSV file:

```
<unique identifier>,<form ID>,<survey total earned score>,<question
   number>,<answer text>,<answer score/weight>
```

The first row of each survey result has the following additional syntax requirements:

```
<unique identifier>,<form ID>,<survey total earned score>,1,<contact ID or
   associated contact ID>,0
```

- The **1** indicates the first row has question number 1 (as is required).
- The <**contact ID or associated contact ID**> is the identifier for the contact.
- The **0** is a placeholder for the score/weight of this question.

The following table describes what each column in the file contains for each line.

Field	Description
unique identifier	An identifier matching the unique identifier in the Results file name.
form ID	The form ID used in the Form file name.
survey total earned score	The total score for the survey.
contact ID or associated contact ID	The identifier of the contact this survey applies to. **NOTE** The identifier that is used in the Results file is determined by the survey identifier selected on the Post Call Survey page in Webex WFO (Application Management > QM > QM Contact

Field	Description
	Flows > Post Call Survey).
question number	The question number matching the question number from the Form file.
answer text	The answer text matching one of the <text for result> option values for the question in the Form file.
answer score/weight	The score earned by the answer matching the <value/weight> for one of the option values for the question in the Form file.

A question is only included in the imported results if all of the following are true:

- The form ID matches the form ID of an imported form.
- The question number matches the question number on that imported form.
- The answer text and the answer score/weight match the text for result and value/weight for an answer option on that question.

Example

The following are examples of Form and Results CSV files for a scenario where a post-call survey consists of five questions. A customer answers the survey after a contact identified with the contact ID **987654321**.

The customer enters the following answers to the survey:

Question 301	3
Question 302	2
Question 303	3
Question 304	4
Question 305	4

The Form and Results file for this survey are as follows.

Form File Name: Form_3.csv

```
1 Customer_Satisfaction_Survey,editable,2019-10-17,200
2 1,DIGITS,"Contact Identifier",0
```

```
3 | 301, OPTION,Were you happy with wait time,1 - strongly disagree - 00,2 - disagree -
  | 10,3 - neither - 20,4 - agree - 30,5 - strongly agree - 40
4 | 302,OPTION,How was service,1 - strongly disagree - 00,2 - disagree - 20,3 - neither
  | - 10,4 - agree - 30,5 - strongly agree - 40
5 | 303,OPTION,Did we resolve issue,1 - strongly disagree - 00,2 - disagree - 20,3 -
  | neither - 10,4 - agree - 30,5 - strongly agree - 40
6 | 304,OPTION, Was agent knowledgeable - strongly disagree - 00,2 - disagree - 10,3 -
  | neither - 20,4 - agree - 30,5 - strongly agree - 40
7 | 305,OPTION,How satisfied with general services,1 - strongly disagree - 00,2 -
  | disagree - 10,3 - neither - 20,4 - agree - 30,5 - strongly agree - 40
```

Results File Name: Results_20191017_1309_1571310547.csv

```
1 | 1571310547,3,110,1,987654321,0
2 | 1571310547,3,110,301,neither,20
3 | 1571310547,3,110,302,disagree,20
4 | 1571310547,3,110,303,neither,10
5 | 1571310547,3,110,304,agree,30
6 | 1571310547,3,110,305,agree,30
```

Import and sync file-based data

You can import and synchronize QM and Analytics user and team data using GIS functionality to add and update this data.

NOTE You cannot delete data using files. That function remains a manual process.

The files are placed in the location configured in Webex WFO Application Management on the Data Server Configuration page in the Regional Data Server GIS File Location section. The Data Server will import files from this location. Once the files are processed by the sync process, the files are archived both on the Data Server and in the cloud.

Good files are archived under the ~/gis/archives<date> folder on the Data Server. They are kept for 1 week. Bad files are not uploaded. They are moved to the ~/gis/penaltyBox/<date> folder on the Data Server, and no further attempts are made to upload them.

Users file

User information is contained in a file called Users.csv. When the file is imported:

- Users are created if they do not exist in the Data Server, and roles are assigned.
- If users already exist, the user names and teams are updated, roles are assigned if not already assigned.

  NOTE Do not reactivate users that have been deactivated. This is to allow you to manually deactivate a team without deactivating users in the ACD.

- Users are assigned to the default team if the team colum is missing or if no team is specified for the user.
- Users must have at least one valid role assigned to them. If a role specified in the file does not exist in Webex WFO, then it is skipped without error.

The details of this file are as follows. Fields in the CSV file can be in any order from left to right.

Field	Required?	Type	Description
acdId	Yes	String	The user's identifier in the ACD.
acdServerId	Yes	Number	Identifier of the ACD. This is the number of the ACD shown on the ACD Configuration page.
displayTimeZone	No	String	The time zone the user's schedules are to be displayed in, in Olson Timezone format. If none is provided, the tenant's timezone is used.
employeeId	No	String	The user's employee ID.
enableScheduling	Yes	Boolean	True or False. Enables the user to be scheduled.
firstName	Yes	String	The user's first name.
lastName	Yes	String	The user's last name.
roles	Yes	String	The roles assigned to the user. Multiple roles are delimited by semicolons. The roles listed must exist.
teamAcdId	No	String	The ACD ID of the team associated with the user.
username	No	String	The user's Webex WFO user name.
windowsLogin	No	String	The user's Windows login, if Active Directory is used.

File example

```
1 acdServerId,acdId,employeeId,firstName,lastName,roles,teamAcdId,username,windowsLo
  gin,displayTimeZone,enableScheduling
2 1,1001,123,Larry,Jones,Agent;Supervisor,9001,larry.jones@t.com,larry.jones,America
  /Chicago,true
3 1,1002,456,Bob,Henderson,Agent,9001,bob.henderson@t.com,bob.henderson,America/Chic
  ago,true
4 1,1003,789,Sara,Williams,Agent,9002,sara.williams@t.com,sara.williams,America/Chic
  ago,true
```

Teams file

Team information is contained in a file called Teams.csv. When the file is imported:

- Teams are created if they do not exist in the Data Server.
- If teams already exist, the team names are updated.
- Teams that were synchronized before but do not exist in the current upload file are deactivated.
- Do not reactivate a team that has been deactivated. This is to allow you to manually deactivate the team without deactivating it in the ACD.

The details of this file are as follows. Fields in the CSV file can be in any order from left to right.

Field	Required?	Type	Description
acdId	Yes	String	The team's identifier in the ACD.
acdServerId	Yes	Number	Identifier of the ACD. This is the number of the ACD shown on the ACD Configuration page.
name	Yes	String	The team's name.

File example

```
1 acdServerId,acdId,name
2 1,9001,Sales
3 1,9002,Support
4 1,9003,Customer Relations
```

Field	Required?	Type	Description
acdId	Yes	String	The service queue's identifier in the ACD.
acdServerId	Yes	Number	Identifier of the ACD. This is the number of the ACD shown on the ACD Configuration page.
name	Yes	String	The team's name.

Import generic CDR data

Generic contact detail records (CDRs) from ACDs that are not natively integrated with Webex WFO can be imported into Webex WFO. This process reconciles these dialer contacts (associates the root calls with agents and screen recordings) so that the Webex WFO search and play functionality works correctly.

NOTE The reconciliation window is the last 8 hours. This process does not reach any further back than that to reconcile calls.

Upload via the Gathering Server

To use this option, you must first create a Generic ACD and associate it with a Data Server via the Webex WFO Data Server Configuration page (Application Management > Global > System Configuration > Data Server Configuration).

Uploading generic CDR files follows the same process used for uploading other CDR files.

In order for the Data Server to process these requests, the following key must be set to True in the pluginsConfigured.properties file:

```
com.calabrio.wfoserver.datagathering.recon.GenericCdrDataCollector=tr
   ue
```

The properties file is located here:

```
C:\Program Files\Common Files\Calabrio ONE\Data
   Server\config\pluginsConfigured.properties
```

CSV files are dropped in the base reconciliation cache directory in the GENERIC_CDR_source folder. In a default installation, this folder is located here:

```
C:\reconciliationCache\<tenant ID>\GENERIC_CDR_source
```

After the file is processed, the file contents are added to the date-based file in the GENERIC_CDR_backup directory. If there is a failure with upload, the content is requeued in the GENERIC_CDR_retry folder.

CSV data definition

The customer provides CSV files that conform to the data definition detailed here.

IMPORTANT The CSV file must not include a header.

```
1 | CallId,CallIdType,AgentId,AgentExtension,CallingNumber,CalledNumber,StartTime,End
```

```
Time,IsInbound
```

Field	Description	Format	Comments
CallId	Required. Call identifier from the source provider.	string, 128 char max	For reconciliation to work, this must match the universal ID for the associated root recording.
CallIdType	Required. Call identifier type.	static string: “GENERIC_ID” 64 char max	
AgentId	Required. Agent identifier.	string, 250 char max	Accepted identifiers: ■ AD login with domain: example mydomain\user.name ■ Email : example user.name@example.com ■ Calabrio database ID: example 42
AgentExtension	Not required. Agent extension.	string, 64 char max	
CallingNumber	Not required. Calling number.	string, 64 char max	
CalledNumber	Not required. Called number.	string, 64 char max	
StartTime	Required.	yyyy-MM-	

Field	Description	Format	Comments
	Call start time in UTC.	ddTHH:mm:ssZ	
EndTime	Required. Call end time in UTC.	yyyy-MM-ddTHH:mm:ssZ or OPEN	The “OPEN” value is for transfer scenarios where the CDR cannot provide a valid end time for the transferred leg. When the transferred leg is reconciled, the end time for the leg will match the end time of the root recording. The agent for the transferred leg will be the first agent to receive the transfer, regardless of how many additional transfers occur.
IsInbound	Not required. Call is inbound.	Boolean string: true/false/not provided will be null	

Example CDR with defined end time

```
123a-456b-789c,GENERIC_ID,ad\userA.nameA,1234,18005551234,5554443333,2021-05-07T19:57:3200Z,2021-05-07T20:07:00Z,true
```

Example CDR with open end time

```
321a-456b-789c,GENERIC_ID,ad\userB.nameB,1235,18005551235,5554443333,2021-05-07T20:07:00Z,OPEN,true
```