



CommunityWFM Cloud Connect REST API for Contact Volume and Agent State Data

For CommunityWFM Software Version 25.3+
November 2025

3400 Waterview Parkway, Suite 325
Richardson, Texas 75080

phone 877-668-6870
web CommunityWFM.com

Table of Contents

About this document.....	4
Prior to using the REST API	4
Types of Data Used in WFM.....	4
Historical Contact Volume Data.....	4
Agent State Data.....	5
People Data	5
Collection Point Data	5
Resources	5
Retrieve Token.....	6
Initial Login	6
Refresh Token	7
People.....	8
People Data Layout.....	8
Get List of Agents	8
Insert Single Person.....	9
Bulk Insert People.....	9
Collection Point.....	10
CollectionPoint Data Layout.....	10
Get List of CollectionPoints.....	10
Insert Single Collection Point.....	11

Bulk Insert Collection Points.....	11
Agent Activity	12
Agent Activity Data Layout	12
Bulk Insert Agent Activity	13
Call Volume	14
Historical Contact Volume Data Layout.....	14
Bulk Insert Call Volume	16

About this document

This document is a description of the CommunityWFM REST API.

The REST API provides a mechanism for importing agents, collection points, agent activity, and call volume into CommunityWFM. This data allows agent real-time adherence and forecasting with CommunityWFM.

Prior to using the REST API

To use this REST API you will need to obtain:

From your CommunityWFM Project Manager—

- a **Base URL**

From your WFM Administrator—

- a user credential
- a valid **username** (in email format)
- a **password**

This credential created by the WFM administrator should **only be used for API access** and not be assigned to a person.

Any time **AccountID** is referenced: use "1" unless instructed otherwise.

Types of Data Used in WFM

Historical Contact Volume Data

Fifteen (15) - minute interval contact data is used for forecasting agent schedules. This data may be automatically populated or imported on demand through the application. A historical data collection service within CommunityWFM will be enabled to routinely capture new data that is present in the API data, whether it is uploaded once a day, once a week, or every fifteen minutes.

Agent State Data

Agent state data is used for adherence reporting. This data, posted using the Agent Activity API, will be automatically populated to the application database. A real-time data collection service within CommunityWFM will be enabled to routinely capture new data that is present in the API data.

Data is available in the application as real-time as it is presented to the Agent Activity API and is suitable for real-time adherence applications. Batch uploads of agent state data provide robust adherence data, but on a delayed basis.

People Data

Simple agent data, such as name and data source ID; used to create new agent profiles.

You may also define agent profiles manually within the CommunityWFM application.

Collection Point Data

Simple information about collection points, such as queue name. This is used to create a list of available queues for the CommunityWFM user.

Resources

This REST API exposes the following resources:

- **People** – This allows you to define a list of people whose real-time agent state data you will provide in the Agent Activity table, below. This data includes AgentId, First Name, and Last Name.
- **Collection Point** – This allows you to define a list of collection points, sometimes referred to as skills or queues, whose interval data you will provide in the Call Volume. This data includes CollectionPointId and Description.
- **Agent Activity** – This allows you to insert agent states to be used for the real-time adherence. This data includes AgentId, Event Time, Activity(state) and an optional Reason Code.
- **Call Volume** – This allows you to insert interval volume data to be used for forecasting. This data includes at a minimum: Interval, Total ContactsOffered for the interval, Contacts Abandoned for the interval, and Total HandleTime for the interval.

Retrieve Token

The Community Cloud Connect API requires a token to make API calls. Below are the calls needed to get the required token.

Initial Login

This will provide an initial token necessary to make subsequent calls.

POST **{Base URL}/api/ccs/auth/login**
Header: Content-Type application/json

Body: {
 "userName":"valid email address",
 "password":"valid password",
 "setCookie": false
 }

Sample Response:

```
{  
  "Success": true,  
  "ExceptionResult":null,  
  "Token": "eyJabcGiOiJIUzI1NiIsInR5cCI6IkpX...",  
  "RefreshToken":"eyJhbGhijkiJIUzI1NiIsInR5c...",  
  "Message":"Successful FORMS Authentication",  
  "Data":null  
}
```

Refresh Token

This will get a refreshed token, tokens expire in 1 hour.

POST **{Base URL}/api/ccc/auth/refreshtoken**
Header: **Content-Type** **application/json**

Body: {
 "refresh_token":" eyJhbHijkiJIUzI1NilsInR5c...",
 "grant_type":"string",
 }

Sample Response:

```
{  
  "Success": true,  
  "ExceptionResult": null,  
  "Token": "",  
  "RefreshToken": "",  
  "Message": "token refresh successful",  
  "Data": {  
    "Token": "eyJhbGcAcabafiOiJIUzafae...",  
    "RefreshToken": "ayJdsfGHiOiJIUzI1Nil...",  
  }  
}
```

People

The People API is a quick and easy way to get all agents into CommunityWFM.

People Data Layout

People API Schema		
Parameter	Data Type	Description
AccountId	PK, Int, not null	Unique identifier per data source provided by CommunityWFM technical team.
AgentId	PK, nvarchar(50), not null	Login ID from device that ties agent states to agent name
FirstName	nvarchar(20), not null	First name of agent
LastName	nvarchar(20), not null	Last name of agent

Get List of Agents

This retrieves a list of all agents currently in the People table. This is useful for determining if the insert methods below were successful.

GET **{Base URL}/api/ccc/people/list**

Authorization: Auth Type: Bearer "token received from login or refresh"

Parameter: accountId 1

Insert Single Person

This will allow you to insert an agent at a time.

POST **{Base URL}/api/ccc/people/InsertPerson**

Authorization: Auth Type: Bearer "token received from login or refresh"

Header: Content-Type application/json

Body: {
 "AccountId":"1",
 "AgentId":"2",
 "FirstName":"First2",
 "LastName":"Last2"
 }

Bulk Insert People

This will allow you to insert a list of people.

POST **{Base URL}/api/ccc/People/BulkInsertPeople**

Authorization: Auth Type: Bearer "token received from login or refresh"

Header: Content-Type application/json

Body: {
 "AccountId":"1",
 "AgentInserts" : [
 {
 "AgentId":"2",
 "FirstName":"Joe",
 "LastName":"Doe"
 },
 {
 "AgentId":"3",
 "FirstName":"Jane",
 "LastName":"Doe"
 }
]
 }

Collection Point

The Collection Point API is a quick and easy way to put all collection points (skills/queues) into CommunityWFM.

CollectionPoint Data Layout

CollectionPoint API Schema		
Parameter	Data Type	Description
Accountid	PK, Int, not null	Unique identifier per data source provided by CommunityWFM technical team.
CollectionPointid	PK, nvarchar(50), not null	System ID of collectionpoint (ie skill #, Application #)
Description	nvarchar(255), not null	Description of collection point

Get List of CollectionPoints

This will get a list of all collection points currently in the CollectionPoints table. This is useful for determining if the insert methods below were successful.

GET {Base URL}/api/cc/CollectionPoint/List

Authorization: Auth Type: Bearer "token received from login or refresh"

Parameter: accountid 1

Insert Single Collection Point

This will allow you to insert one collection point at a time.

POST **{Base URL}/api/ccc/CollectionPoint/InsertCollectionPoint**

Authorization: Auth Type: Bearer "token received from login or refresh"

Header: Content-Type application/json

Body: {
 "AccountId":"1",
 "CollectionPointId":"1",
 >Description":"Collection Point 1"
 }

Bulk Insert Collection Points

This will allow you to insert a list of collection points.

POST **{Base URL}/api/ccc/CollectionPoint/BulkInsertCollectionPoints**

Authorization: Auth Type: Bearer "token received from login or refresh"

Header: Content-Type application/json

Body: {
 "AccountId":"1",
 "CollectionPointInserts" : [
 {
 "CollectionPointId":"1",
 >Description":"Collection Point 1"
 },
 {
 "CollectionPointId":"2",
 >Description":"Collection Point 2"
 }
]
 }

Agent Activity

The Agent Activity API is a quick and easy way to get all agent state changes into CommunityWFM. We recommend that you queue up data for 30 seconds before sending it to the Agent Activity API.

Agent Activity Data Layout

Important: Event Time **must** be in UTC time.

Agent Activity API Schema		
Parameter	Data Type	Description
AccountId	PK, Int, not null	Unique identifier per data source provided by CommunityWFM technical team.
AgentId	PK, nvarchar(50), not null	Agent login ID from system, used to match transactions to agent name
EventTime	PK, datetime, not null	Primary Key, (Start Time of state; <u>MUST</u> be in UTC)
Activity	nvarchar(10), not null	State Code, the work state name (i.e., On a Call, wrap-up)
ReasonCode *	nvarchar(10), not null	Reason Code for secondary state description, (i.e., Aux with reason)

* Optional column that may or may not be required for a particular application.

Bulk Insert Agent Activity

This will allow you to insert a list of agent state changes.

POST **{Base URL}/api/ccc/AgentActivity/BulkInsertAgentActivity**

Authorization: Auth Type: Bearer "token received from login or refresh"

Header: Content-Type application/json

Body: {
 "AccountId":"1",
 "AgentUpdates":[
 {
 "AgentId":"4",
 "EventTime":"10/20/2020 2:13:12",
 "Activity":"notready",
 "ReasonCode":"1"
 },
 {
 "AgentId":"5",
 "EventTime":"10/20/2020 2:13:12",
 "Activity":"OnCall"
 }
]
 }

Call Volume

The Call Volume API is a quick and easy way to get call volume data into CommunityWFM. We recommend sending this data once every 15 minutes.

Historical Contact Volume Data Layout

Historical Collection By Interval API Schema		
Parameter	Data Type	Description
AccountId	PK, int, not null	Unique identifier per data source provided by Community technical team.
CollectionPoint	PK, nvarchar(50), not null	Contains the queue name, skill or collection point for a particular row of data.
Interval	PK, datetime, not null	Contains the date and time of the interval for the particular row of historical data. 24 hr. time format most useful. <i>Time zone can be local or UTC.</i> Intervals <u>must</u> be in a 15-minute increment on the 15 minutes. Examples: 2024-06-15 13:00 2024-06-15 13:15 2024-06-15 13:30 2024-06-15 13:45
ContactsOffered	bigint, not null	Number of contacts offered during this time interval for the given skill.
ContactsNotHandled	bigint, not null	Number of contacts not handled during this time interval for the given skill. Note that this value contains all contacts not handled, not just those with a disposition of 'abandoned.'
TalkTime	Float, not null	Total number of seconds of talk time and wrap-up time for the interval and the given skill.

HoldTime (required by API can be set to 0)	Float, not null	Total number of seconds of hold time for the interval and the given skill. This field is optional and may be used if hold time is not included in the TalkTime column.
ServiceLevelPct (required by API can be set to 0)	Float, not null	Service level achievement percentage for a given skill. Data from this column is not currently utilized.
CollectionTimeStamp*	DateTime, null	The timestamp when the row is inserted into the database. <i>Time zone can be local or UTC.</i>
OutboundContacts*	bigint, null	Number of <i>outbound</i> contacts during this time interval for the given skill.
OutboundTimeStamp*	Float, null	Total number of seconds of talk time and wrap-up time for the interval and the given <i>outbound</i> skill. Note: Please disregard the name of the field - it should not be filled with a DateTime time stamp.
User Data 1*	Any	Unique identifier provided by Customer
User Data 2*	Any	Unique identifier provided by Customer
User Data 3*	Any	Unique identifier provided by Customer
User Data 4*	Any	Unique identifier provided by Customer
User Data 5*	Any	Unique identifier provided by Customer
User Data 6*	Any	Unique identifier provided by Customer
User Data 7*	Any	Unique identifier provided by Customer
User Data 8*	Any	Unique identifier provided by Customer
User Data 9*	Any	Unique identifier provided by Customer
User Data 10*	Any	Unique identifier provided by Customer

* Optional column that may or may not be required for a particular application.

Bulk Insert Call Volume

This will allow you to insert a list of call volume changes.

POST **{Base URL}/api/ccc/CallVolume/BulkInsertCallVolume**

Authorization: Auth Type: Bearer "token received from login or refresh"

Header: Content-Type application/json

```
Body: {
  "AccountId": "1",
  "CallVolumeUpdates": [
    {
      "UpdateExisting": "false",
      "CollectionPoint": "1",
      "Interval": "10/30/2020 2:15:00",
      "ContactsOffered": "5",
      "ContactsNotHandled": "2",
      "TalkTime": "600",
      "HoldTime": "30",
      "ServiceLevelPct": "90",
      "CollectionTimeStamp": "10/30/2020",
      "OutboundContacts": "2",
      "OutboundTimeStamp": "3"
    },
    {
      "UpdateExisting": "false",
      "CollectionPoint": "2",
      "Interval": "10/30/2020 2:15:00",
      "ContactsOffered": "8",
      "ContactsNotHandled": "0",
      "TalkTime": "300",
      "HoldTime": "30",
      "ServiceLevelPct": "90",
      "CollectionTimeStamp": "10/30/2020",
      "OutboundContacts": "2",
      "OutboundTimeStamp": "3"
    }
  ]
}
```

```
    },  
    {  
      "UpdateExisting":"true",  
      "CollectionPoint":"2",  
      "Interval":"10/30/2020 1:15:00",  
      "ContactsOffered":"9",  
      "ContactsNotHandled":"3",  
      "TalkTime":"300",  
      "HoldTime":"30",  
      "ServiceLevelPct":"90",  
      "CollectionTimeStamp":"10/30/2020",  
      "OutboundContacts":"2",  
      "OutboundTimeStamp":"3",  
      "UserData1":"some extra data",  
      "UserData2":"some extra data",  
      "UserData3":"some extra data",  
      "UserData4":"some extra data",  
      "UserData5":"some extra data",  
      "UserData6":"some extra data",  
      "UserData7":"some extra data",  
      "UserData8":"some extra data",  
      "UserData9":"some extra data",  
      "UserData10":"some extra data",  
    }  
  ]  
}
```