Genesys Multicloud CX Web-based API Reference

Statistics Overview

11/27/2025

Contents

You can use the Statistics API to create a subscription for multiple statistics and then receive notifications when the values of those statistics change.

API Overview

You can work with the API in two modes:

1. Create a subscription and receive notifications. This is the preferred approach if you need up-to-date information over a period of time. When you create a subscription, you can start receiving calculated statistics. These statistics are either pre-configured for your account/contact center and or come from the definitions you provided in the request.
2. Get the current values of statistics without a subscription. This is useful if your client can't maintain a CometD connection or if your application requires this information only once. This API works with statistics that are pre-configured for your contact center.

The Statistics API is divided into the following categories:

- Requests (`/statistics`) — Subscribe to statistics, unsubscribe from statistics, and get the current value of statistics without a subscription.
- Notifications (`/notifications`) — Implements CometD channels for statistics notifications.

> **Important**
>
> The examples on this page illustrate only positive cases. For a full list of possible response codes, refer to the API reference information for the requests.

API Details

Find the API requests, responses, and details here:

- Statistics API
- Statistics Notifications API

Client libraries

Genesys also offers client libraries for the Statistics API in both Node.js and Java. These libraries

simplify how you interact with the API and they take care of a lot of the supporting code needed to make HTTP requests and enable CometD. Genesys recommends using the client libraries if possible. For help deciding if you should use them, check out the Getting Started page.

There are also some Statistics tutorials that demonstrate how to use the client libraries. See Tutorials for details.

Authentication

In order to initialize and use the Statistics API, you first need to authenticate with the Authentication Client Library or the Authentication API.

Get the values of statistics without a subscription

You can get the current value of predefined statistics from Stat Server without a subscription by making a **POST** request to /statistics/v3/operations/get-statistic-ex.

This operation has just one mandatory body parameter called **statistics**.

Sample request

POST to `/statistics/v3/operations/get-statistic-ex`

Here's the JSON body included with the above request:

```
{
    "data":{
       "statistics":[
          {
             "name":"CurrentAgentState",
             "objectType":"Agent",
             "objectId":"agent2"
          },
          {
             "name":"TimeInCurrentState",
             "objectType":"Agent",
             "objectId":"agent2"
          }
       ]
    }
 }
```

Sample JSON response body

```
{
    "status":{
       "code":0
    },
    "data":{
       "statistics":[
```

```
         {
            "name":"CurrentAgentState",
            "objectType":"Agent",
            "objectId":"agent2",
            "value":{
               ...
            },
            "timestamp":1490830620000
         },
         {
            "name":"TimeInCurrentState",
            "objectType":"Agent",
            "objectId":"agent2",
            "value":{
               ...
            },
            "timestamp":1490830620000
         }
      ]
   }
}
```

You can also use an API call to get the current value of **one** valid statistic from Stat Server by making a **GET** request to /statistics/v3/statistic-values/{statisticName}.

This request accepts two mandatory query parameters, **objectType** and **objectId**.

Sample request

GET `/statistics/v3/statistic-values/{statisticName}?objectType=XXX&objectId=YYY`

Sample JSON response body

```
{
   "status":{
      "code":0
   },
   "data":{
      "statistic":{
         "value":{
            ...
         },
         "timestamp":1490830620000
      }
   }
}
```

Create a subscription

To create a subscription, **POST** to the following path: /statistics/v3/subscriptions

This POST allows an optional query parameter `verbose=` and the body is JSON with the **operationId** and **data** parameters.

Sample request

POST to `/statistics/v3/subscriptions`

Here's the JSON body included with the above request:

```
{
    "operationId":"UUID1",
    "data":{
       "statistics":[
          {
             "statisticId":"UUID2",
             "objectId":"agent",
             "objectType":"Agent",
             "name":"AverageHandlingTime"
          },
          {
             "statisticId":"UUID3",
             "objectId":"agent",
             "objectType":"Agent",
             "definition":{
                "notificationMode":"Periodical",
                "notificationFrequency":10,
                "insensitivity":1,
                "category":"TotalAdjustedTime",
                "subject":"DNStatus",
                "intervalType":"GrowingWindow",
                "mainMask":"WaitForNextCall",
                "dynamicTimeProfile":"00:00",
                "dynamicFilter":"Media=voice",
                "maskType":"DN"
             }
          }
       ]
    }
 }
```

In the above sample, AverageHandlingTime references the corresponding statistic (pre-configured) for the given contact center.

Sample JSON response body

```
 {
    "status":{
       "code":0
    },
    "data":{
       "subscriptionId":"UUID1",
       "operationId":"UUID1",
       "statistics":[
          {
             "statisticId":"UUID2",
             "objectId":"agent",
             "objectType":"Agent",
             "name":"AverageHandlingTime",
             "value":{
                ...
             },
             "timestamp":1490830620000
          },
```

```
         {
            "statisticId":"UUID3",
            "objectId":"agent",
            "objectType":"Agent",
            "value":{
               ...
            },
            "timestamp":1490830620000
         }
      ]
   }
}
```

In the above sample:

- The status code is 0 because this is synchronous operation.
- The subscriptionId is filled from the operationId.

Get the values of statistics in a subscription

To retrieve the values of a set of statistics that was opened within a subscription, you need to make a **GET** request to the following path: /subscriptions/{id}/statistic-values

The GET allows two optional query parameters,`verbose=` and `statisticIds=`.

Sample request

GET `/statistics/v3/subscriptions/UUID1/statistic-values`

Sample JSON response body

```
{
    "status":{
       "code":0
    },
    "data":{
       "subscriptionId":"UUID1",
       "statistics":[
          {
             "statisticId":"UUID2",
             "objectId":"agent",
             "objectType":"Agent",
             "name":"AverageHandlingTime",
             "value":{
                ...
             },
             "timestamp":1490830620000
          },
          {
             "statisticId":"UUID3",
             "objectId":"agent",
             "objectType":"Agent",
             "value":{
                ...
```

```
            },
            "timestamp":1490830620000
         }
      ]
   }
}
```

In the above sample:

- The status code is 0 because this is synchronous operation.

Delete a subscription

You can delete a subscription by sending a **DELETE** request to the following path: /statistics/v3/subscriptions/{id}. This closes all statistics associated with the specified subscription. It's a fire-and-forget operation that always returns a success response.

CometD notifications

You can subscribe to CometD notifications about service state and statistic value updates for statistics in your subscription.

Statistic value updates

Notification topic: `/statistics/v3/updates`

Notification sample:

```
{
    "name":"StatisticUpdate",
    "topic":"/statistics/v3/updates",
    "data":{
       "statistics":[
          {
             "subscriptionId":"UUID1",
             "statisticId":"UUID2",
             "value":{
                ...
             },
             "timestamp":1490830620000
          }
       ]
    }
}
```

Where:

- **subscriptionId** - The ID of the subscription the statistic is associated with.
- **statisticId** - The ID of the statistic.
- **value** - The value of the statistic.

- **timestamp** - The time the statistic was generated (provided by Stat Server).

Service state

If the Statistics service is unable to connect to internal server(s) it needs to communicate with, it notifies subscribers with a ServiceStateChange message where the serviceState is **UNAVAILABLE**. When this happens, the Statistics service drops all subscriptions, so you'll need to re-create your subscriptions when the service is available again.

Notification topic: `/statistics/v3/service`

Notification sample:

Service is unavailable

```
{
    "data":{
       "serviceState":"UNAVAILABLE"
    },
    "name":"ServiceStateChange",
    "topic":"/statistics/v3/service"
}
```

Service is available again

```
{
    "data":{
       "serviceState":"AVAILABLE"
    },
    "name":"ServiceStateChange",
    "topic":"statistics/v3/service"
}
```