



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

API Reference

Context Services 8.5.2

Table of Contents

Context Services 8.5 API Reference	5
List of Operations	7
HTTP Response Codes and Errors	11
Unauthorized Strings	15
Service API	18
Service Operations	20
Associate Service	22
Complete Service	26
Complete State	30
Complete Task	35
Composite Start and Bulk Update	39
Delete Service By ID	50
Export Services to JSON Stream by Date	52
Export Services to File by Date	56
Merge Extension Data	61
Perform State Transition	64
Purge Services	69
Query Business Attribute Schema	72
Query Service by ID	75
Query Services	82
Query State by ID	88
Query States	93
Query Task by ID	96
Query Tasks	100
Start Service	104
Start State	109
Start Task	113
Update Expiration Time	117
Update Service Extension	119
Update State Extension	121
Update Task Extension	123
Service Resources	125
End Event List	126
Extension	133
State Transition Event	135

Service	139
State	142
Start Event List	145
Task	153
Customer Profile API	156
Interaction Operations	158
Query Interactions	159
Interaction Resources	162
Callback Interaction	163
Chat Interaction	170
E-mail Interactions	174
Phone Call Interaction	182
Profile Operations	186
Bulk Profile Import	187
Create Customer Profile	193
Delete Customer Profile	196
Delete Record From Profile Extension	198
Identify Customer	200
Insert Extension Records	204
Merge Customer Profile	206
Query Customer Profile	209
Update Customer Profile	212
Update Record In Profile Extension	215
Profile Resources	217
Customer Profile	218
Profile Attribute Schema	220
Profile Extension	224
Schema Operations	226
Create Identification Key	227
Create Profile Extension Schema	230
Change Metadata Cache	233
Delete Metadata Identification Keys	235
Delete Metadata Profile Extensions	237
Get Identification Keys	239
Get Metadata	241
Query Metadata Roles	244
Query Profile Extension Schema	247

Query Profile Schema	250
Schema Resources	253
Attribute Schema	254
Identification Key	257
Profile Extension Schema	260
Server Operations	263
Set Server Mode	264
Query Server Status	266

Context Services 8.5 API Reference

This guide provides detailed description of REST resources and queries available in Context Services 8.5. Each page is presented with out-of-the-box query examples to meet your business needs.

- For further details about API usage, refer to the [Context Services Developer's Guide](#).
- For further details about deployment with GMS and UCS, refer to the [Context Services User's Guide](#).

Features

The list of [operations](#) and [resources](#) should help you to create an application on top of Context Services using the REST technology:

- Service management only require to install Context Services in GMS:
 - Start and stop services, as well as nested state and tasks.
 - Retrieve events.
 - Access nested data available as flexible JSON extensions.
- Customer profile and interaction management requires a UCS deployment:
 - Create and fill in profiles,
 - Store interaction data.

Adding ContactCenterId and GroupId HTTP headers

In order to support multi-tenancy and business units, your Context Services queries need to include additional HTTP headers:

- ContactCenterId contains the Tenant DBID or the Tenant Name that you set in your configuration..
- GroupId contains an optional business unit name.

Important

You must make sure that you provide these IDs in HTTP headers, not as part of the JSON Body content.

If your request does not include these headers, the Context Services handles the request in single-

tenant mode and uses the provisioned tenant ID with no business unit name of your Context Service application.

Read also [Configuring Tenancy](#).

ContactCenterId Errors

If you provide an incorrect ContactCenterId string, your application receives the following error message:

```
{
  "message": "Invalid tenant specified : specified tenant in HTTP header
'ContactCenterId'='wrong value' not configured. Verify configuration.",
  "exception":
"com.genesyslab.gsg.services.contextservices.exceptions.ContextServicesExceptionResource"
}
```

If you do **not** provide the ContactCenterId string in a multi-tenant configuration, the error message is:

```
{
  "message": "Invalid tenant specified : no tenant specified in multi-tenant environment.
Verify configuration and 'ContactCenterId' HTTP header.",
  "exception":
"com.genesyslab.gsg.services.contextservices.exceptions.ContextServicesExceptionResource"
}
```

List of Operations



Purpose: Entry point of the API Reference.

Contents of the API Reference

As explained in the [Migration page](#), all the service resources are now part of GMS and all the information related to contact information stays in UCS. As a result, the CS API is divided between UCS and GMS.

Operations

Services

Services API Reference

Service Operations

- Maintenance Operations
 - [GET /genesys/1/cs/maintenance/services/stream](#)
 - [POST /genesys/1/cs/maintenance/services/files](#)
 - [POST /genesys/1/cs/maintenance/purge](#)
- Service Operations
 - [POST /genesys/1/cs/services/start](#)
 - [POST /genesys/1/cs/genesys/1/cs/customers/{customer id}/services/{service id}](#)
 - [POST /genesys/1/cs/services/{service id}/end](#)
 - [POST /genesys/1/cs/services_composite/start](#)
 - [GET /genesys/1/cs/services/{service id}](#)
 - [DELETE /genesys/1/cs/services/{service id}](#)
 - [GET /genesys/1/cs/services/anonymous/{contact key}/active](#)

- GET /genesys/1/cs/services/anonymous/{contact key}/completed
- GET /genesys/1/cs/services/anonymous/{contact key}
- GET /genesys/1/cs/customers/{customer id}/services/active
- GET /genesys/1/cs/customers/{customer id}/services/completed
- GET /genesys/1/cs/customers/{customer id}/services
- PUT /genesys/1/cs/services/{service_id} (TTL, Expiration Time)
- PUT /genesys/1/cs/services/{service id}/extensions/{ext name}
- State Operations
 - POST /genesys/1/cs/services/{service id}/states/{state id}/end
 - POST /genesys/1/cs/services/{service id}/states/transition
 - GET /genesys/1/cs/services/{service id}/states
 - GET /genesys/1/cs/services/{service id}/states/{state id}
 - POST /genesys/1/cs/services/{service id}/states/start
 - PUT /genesys/1/cs/services/{service id}/states/{state id}/extensions/{ext name}
- Task Operations
 - POST /genesys/1/cs/services/{service id}/tasks/{task id}/end
 - GET /genesys/1/cs/services/{service id}/tasks
 - GET /genesys/1/cs/services/{service id}/tasks/{task id}
 - POST /genesys/1/cs/services/{service id}/tasks/start
 - PUT /genesys/1/cs/services/{service id}/tasks/{task id}/extensions/{extension name}

Service Resources

- **End Event List:** List of the end events that your application can receive if a service, state, or task is terminated.
- **Extension:** JSON extension for the service API.
- **State Transition Event:** Your application can receive this event if a service performs a transition between two states.
- **Service:** The service resource, which is a container for nested states and tasks.
- **State:** The state resource.
- **Start Event List:** List of the start events that your application can receive if a service, state, or task is started.
- **Task:** The task resource.

Important

Because the extensions are now handled as JSON data, all the queries related to service, state, and task extensions are now deprecated.

Customer Profile

Customer Profile API Reference

Important

Prerequisites: You need to **enable** profiles in UCS.

- Profile Operations
 - `POST /profiles`
 - `GET /profiles`
 - `POST /profiles/${customer id}/extensions`
 - `GET /profiles/${customer id}`
 - `PUT /profiles/${customer id}`
 - `PUT /profiles/${customer id}/extensions/${ext name}/by/unique`
 - `PUT /profiles/${customer id}/extensions/${ext name}/delete/by/unique`
 - `PUT profiles/${customer_id}/merge/${src_id}/`
- Schema and Identification Key Operations
 - `GET /metadata/profiles/`
 - `POST /metadata/profiles/extensions`
 - `GET /metadata/profiles/extensions`
 - `GET /metadata/identification-keys`
 - `POST /metadata/identification-keys`
 - `GET ${content-type} /metadata`
 - `DELETE /metadata/profiles/extensions/${extension-name}`
 - `DELETE /metadata/identification-keys/${id_key-name}`

- Interaction Operations
 - GET /customers/\${customer id}/interactions
 - GET /services/\${service id}/interactions
 - GET /interactions/\${interaction id}

HTTP Response Codes and Errors



Purpose: Describes the HTTP responses that your application can receive from UCS.

Introduction

The Universal Contact Server returns HTTP status codes and messages for every operation, in the requested format. Status codes match **standard HTTP codes**, but messages can differ and provide additional details included in the header of the response.

Important

Additional result and error codes may be returned due to external web servers and layers involved in your operations.

Read Log Lines

Log lines are compliant with the following syntax:

```
14:33:39.485[1] Trc[2] 29998 [let-3851106] 2010-04-02[3] 14:33:39[3] 127.0.0.1[4]
8080[5] GET[6] /cv/server/status[7] - 200[8] - 0 2[9]
http://localhost:8080[10] Apache-HttpClient/4.0-beta2 (java 1.5)[11] -
```

- [1] = timestamp
- [2] = loglevel
- [3] = date and time when the request was received
- [4] = ip address
- [5] = server port
- [6] = http method
- [7] = requested url
- [8] = http response code
- [9] = execution time in ms
- [10] = called host/port
- [11] = user agent making the call

Successful Result

A successful response to a request is marked by HTTP Status Code 200 (OK). In that case, your application may get additional information in the header and the body of the response. Refer to the Response section of your operation's page to get the detailed list of returned information. Questions about the returned content can be submitted as comments in this wiki. The following table lists the **standard HTTP codes** used by Context Services for a successful response.

Successful results

Code	Title	Description
200	OK	Success!
201	Created	The request has been fulfilled and resulted in a new resource being created.
204	No Content	The operation was successful and returned with no content.

Errors

For responses with HTTP status code 4xx or 5xx, the response body contains an application-specific description of the error instead of a representation of the requested resource. Context Services errors consist of an application-specific error code and description, and are formatted in JSON as follows:

```
error = {
  "http_method" : "POST",
  "title" : "bad parameter",
  "description" : "bad parameter 'server mode' reason : server mode has not a correct value 'something'",
  "code" : 4020,
  "uri" : "http://localhost:8080/server/mode"
}
```

The following table lists the specific errors that operations can encounter. This list is not restrictive; additional error codes could be returned due to external web servers and layers involved:

Errors

Code	Title	Description
303	See Other	General error which can occur if your application refers to a URI which moved, due to API changes.
400	Bad Request	General error which can be one of the following reasons: <ul style="list-style-type: none"> Missing required parameter. Parameter value of unexpected type.

Code	Title	Description
		<ul style="list-style-type: none"> Invalid object syntax if you used JavaScript objects in your operation's input. Missing required attribute if you used JavaScript objects in your operation's input.
401	Not Authorized	Credentials are missing or incorrect, or the given user is not allowed to execute a given service (such as an administrative service method that changes the profile schema).
403 Not available for now.	Forbidden	<p>The operation is forbidden and the reason is specified in the error message. This error is returned in the following cases:</p> <ul style="list-style-type: none"> Attempt to query customer profile or service information when the server is in Maintenance Mode. Attempt to alter the schema while the server is in Production Mode. Attempt to add customer profiles prior to the definition of a profile schema. Attempt to change the server mode times out, due to maintenance operations or schema modifications to complete.
404	Not Found	The specified URI is invalid, or the requested resource (such as a customer, service, state, task, extension, or identification key) does not exist.
405	Method Not Allowed	Returned when an unsupported operation is requested. For instance, if a resource supports only PUT and GET operations, a POST request on this resource returns this error.
415	Unsupported Media Type	In the header of your request, Content-Type is not set to a valid value. Most operations of Context Services support only "application/json". Check the

Code	Title	Description
		operation description in the API .
500	Internal Server Error	An unexpected error occurred in UCS (for instance, a runtime exception). The error message suggests to forward logs to Genesys Customer Support.
502	Bad Gateway	Returned when one or more of the backend systems required to fulfill the response (the CV database or UCS, for example) are either unavailable or returned an error. Context Services is not responsible for this message. This error may occur due to external proxies or cache involved in your operation.
503	Service Unavailable	<p>UCS is unable to process the given request. Example situations include:</p> <ul style="list-style-type: none"> • Requests on a whole, or for a specific client/tenant/role, are being throttled. • The server is switching from production to maintenance mode, or vice versa, and does not accept requests for the moment.

Unauthorized Strings



Specifies the list of strings that your application cannot use to create resources and schemas.

Returned Error

Context Services operations enable your application to create resources, schemas, and extensions. However, the keywords identified on this page should not be used to create or fill in attribute values. If your application does use one of the reserved keywords in PUT or POST operations, then a HTTP 400 Bad Request is returned with a "bad parameter" message in the following format:

```
bad parameter '${parameter-name}' reason: '${unauthorized-keyword}' keyword is not authorized.
```

For instance, if a PUT operation used the 'Select' keyword as a field value for the parameter 'name', your application receives the following message:

```
bad parameter 'name' reason : 'Select' keyword is not authorized.
```

Keywords Reserved to Context Services

- extension
- none
- profile
- service
- state
- task

Keywords Reserved to Database

- alter
- by
- comment
- constraint
- create

Unauthorized Strings

- cursor
- database
- delete
- from
- having
- identity
- index
- inner
- insert
- is
- join
- left
- null
- order
- outer
- prepare
- primary
- procedure
- return
- right
- select
- set
- size
- table
- truncate
- union
- update
- when
- where

Type Keywords

- bigint
- binary

- bit
- blob
- boolean
- char
- clob
- currency
- date
- datetime
- decimal
- double
- float
- int
- integer
- long
- longvarbinary
- money
- nchar
- number
- numeric
- real
- smalldatetime
- smallint
- smallmoney
- string
- time
- timestamp
- tinyint
- varbinary
- varchar
- varchar2

Service API

All the queries related to data managed on the GMS side. These queries cover service management.

Service Resources

- **End Event List:** List of the end events that your application can receive if a service, state, or task is terminated.
- **Extension:** JSON extension for the service API.
- **State Transition Event:** Your application can receive this event if a service performs a transition between two states.
- **Service:** The service resource, which is a container for nested states and tasks.
- **State:** The state resource.
- **Start Event List:** List of the start events that your application can receive if a service, state, or task is started.
- **Task:** The task resource.

Service Operations

- Maintenance Operations
 - **GET** `/genesys/1/cs/maintenance/services/stream`
 - **POST** `/genesys/1/cs/maintenance/services/files`
 - **POST** `/genesys/1/cs/maintenance/purge`
- Service Operations
 - **POST** `/genesys/1/cs/services/start`
 - **POST** `/genesys/1/cs/genesys/1/cs/customers/{customer id}/services/{service id}`
 - **POST** `/genesys/1/cs/services/{service id}/end`
 - **POST** `/genesys/1/cs/services_composite/start`
 - **GET** `/genesys/1/cs/services/{service id}`
 - **DELETE** `/genesys/1/cs/services/{service id}`
 - **GET** `/genesys/1/cs/services/anonymous/{contact key}/active`
 - **GET** `/genesys/1/cs/services/anonymous/{contact key}/completed`

- GET /genesys/1/cs/services/anonymous/\${contact key}
- GET /genesys/1/cs/customers/\${customer id}/services/active
- GET /genesys/1/cs/customers/\${customer id}/services/completed
- GET /genesys/1/cs/customers/\${customer id}/services
- PUT /genesys/1/cs/services/{service_id} (TTL, Expiration Time)
- PUT /genesys/1/cs/services/\${service id}/extensions/\${ext name}
- State Operations
 - POST /genesys/1/cs/services/\${service id}/states/\${state id}/end
 - POST /genesys/1/cs/services/\${service id}/states/transition
 - GET /genesys/1/cs/services/\${service id}/states
 - GET /genesys/1/cs/services/\${service id}/states/\${state id}
 - POST /genesys/1/cs/services/\${service id}/states/start
 - PUT /genesys/1/cs/services/\${service id}/states/\${state id}/extensions/\${ext name}
- Task Operations
 - POST /genesys/1/cs/services/\${service id}/tasks/\${task id}/end
 - GET /genesys/1/cs/services/\${service id}/tasks
 - GET /genesys/1/cs/services/\${service id}/tasks/\${task id}
 - POST /genesys/1/cs/services/\${service id}/tasks/start
 - PUT /genesys/1/cs/services/\${service id}/tasks/\${task id}/extensions/\${extension name}

Important

Because extensions are replaced with flexible JSON data on the GMS side, all resources and requests specific to extensions are deprecated for services, states, and tasks.

Service Operations

Purpose: Groups the operations that your application can create for both services, states, tasks, and interaction resources.

- Maintenance Operations
 - GET /genesys/1/cs/maintenance/services/stream
 - POST /genesys/1/cs/maintenance/services/files
 - POST /genesys/1/cs/maintenance/purge
- Service Operations
 - POST /genesys/1/cs/services/start
 - POST /genesys/1/cs/genesys/1/cs/customers/{customer id}/services/{service id}
 - POST /genesys/1/cs/services/{service id}/end
 - POST /genesys/1/cs/services_composite/start
 - GET /genesys/1/cs/services/{service id}
 - DELETE /genesys/1/cs/services/{service id}
 - GET /genesys/1/cs/services/anonymous/{contact key}/active
 - GET /genesys/1/cs/services/anonymous/{contact key}/completed
 - GET /genesys/1/cs/services/anonymous/{contact key}
 - GET /genesys/1/cs/customers/{customer id}/services/active
 - GET /genesys/1/cs/customers/{customer id}/services/completed
 - GET /genesys/1/cs/customers/{customer id}/services
 - PUT /genesys/1/cs/services/{service_id} (TTL, Expiration Time)
 - PUT /genesys/1/cs/services/{service id}/extensions/{ext name}
- State Operations
 - POST /genesys/1/cs/services/{service id}/states/{state id}/end
 - POST /genesys/1/cs/services/{service id}/states/transition
 - GET /genesys/1/cs/services/{service id}/states
 - GET /genesys/1/cs/services/{service id}/states/{state id}
 - POST /genesys/1/cs/services/{service id}/states/start
 - PUT /genesys/1/cs/services/{service id}/states/{state id}/extensions/{ext name}
- Task Operations
 - POST /genesys/1/cs/services/{service id}/tasks/{task id}/end
 - GET /genesys/1/cs/services/{service id}/tasks

- GET /genesys/1/cs/services/{service id}/tasks/{task id}
- POST /genesys/1/cs/services/{service id}/tasks/start
- PUT /genesys/1/cs/services/{service id}/tasks/{task id}/extensions/{extension name}

Important

Because extensions are replaced with flexible JSON data on the GMS side, all resources and requests specific to extensions are deprecated for services, states, and tasks.

Associate Service

POST /genesys/1/cs/customers/\${customer_id}/services/\${service_id}

Description

Your application can use this operation after the service was created with the [Start Service](#) operation to associate an [Anonymous Service](#) with a customer, or to associate the service with a different customer. This operation overrides prior associations. For example:

1. Associate 'service 1' with 'customer 1'
2. Associate 'service 1' with 'customer 2'

Result: 'service 1' no longer appears in the active or completed service history of 'customer 1'. See [Query Services](#) for further information. This operation can also update some fields of the service resource in the body of the request. In 8.0.2, your application can specify values for extensions (single-valued and multi-valued). If your application already recorded a value or a set of values for a given extension, the previous record is replaced with the new one.

Operation

Prerequisites The service is started, see [Start Service](#).

Associate a service with a customer

ID	CV.WS.SRV.2		
Method	POST		
URL	/genesys/1/cs/customers/\${customer_id}/services/\${service_id}		
Name	Type	Mandatory	Description
URI Parameters			
\${customer_id}	string	yes	The ID of the customer.
\${service_id}	integer	yes	The ID of the service.
Optional Body: Service Start Event. This operation's body can include some fields of the Service Start Event for update (for instance, the interaction ID or the resource ID).			
service_type	long or string	no	The unique ID associated with the type of service, typically the DB ID of a value in the Service Type Business Attribute.

ID	CV.WS.SRV.2		
			Refer to Configuration Options for more details on <i>Business Attribute</i> mapping.
session_id	string	no	The ID of the related session, for instance, the orchestration session, or any other business session.
interaction_id	string	no	The ID of the related Genesys interaction. This ID can be used by other Genesys reporting products such as Stat Server, URS, Composer, and GVP.
application_type	long or string	no	The unique ID associated with the type or class of application issuing the service event. Refer to Configuration Options for more details on Business Attribute mapping. May be used to group related applications, potentially across resource types.
application_id	integer	no	The unique ID (such as Genesys DB ID) for the application issuing the service event, such as a GVP VoiceXML application or an Orchestration SCXML application.
resource_type	long or string	no	The unique ID associated with the type or class of resource which provides the service (for example GVP, Agent Desktop, Orchestration). Refer to Configuration Options for more details on Business Attribute mapping.
resource_id	integer	no	The unique DB ID for the specific resource which provides the service. For instance:

ID	CV.WS.SRV.2		
			<ul style="list-style-type: none"> the Genesys DB ID of a specific GVP or orchestration platform. the DB ID of a given agent, according to the context.
media_type	long or string	no	The media type applicable to the given service/state/task, for instance, email, voice, chat, etc. Refer to Configuration Options for more details on Business Attribute mapping.
est_duration	integer	no	The estimated service/state/task duration in seconds.
timestamp	date/time	no	The UTC time at which the event was raised, with a precision of milliseconds, using the ISO 8601 representation: [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z. If the application does not specify this timestamp, the server does it when the event is processed.
<extension key>	Any JSON type	no	Service attached data as key-value pairs. You can add as many key-value pairs as needed.

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response

HTTP code	200
HTTP Title	OK

Example

Operation

The following code sample associates the customer with a service, and specifies attached data as json arrays for "Feedback" and "Satisfaction".

```
POST http://localhost:8080/genesys/1/cs/customers/0004Va58A92T0018/services/1500
{
  "timestamp": "2009-05-12T12:05:12.145Z",
  "interaction_id": "123ABCAADFJ1259ACF",
  "media_type": 1,
  "resource_id": 5005,
  "resource_type": 2,
  "disposition": 10,
  "Feedback":
  {
    "FeedbackType": "survey",
    "rating": 7,
    "notes": "warm welcome at frontdesk, thanks for the nice trip"
  },
  "Satisfaction": [
    {
      "rating": 2,
      "pertinence": 8,
      "useful": true,
      "place": "Terranova mexico resort"
    },
    {
      "rating": 8,
      "pertinence": 4,
      "useful": false,
      "place": "Fancy resort Paris"
    }
  ]
}
```

Result

200 OK

Complete Service

POST /genesys/1/cs/services/\${service_id}/end

Description

Terminates the given service. Your application can update service information in the body of this operation. In 8.5.0, your application can attach data to the service by adding key-value pairs in JSON format. If your application already recorded attach data, the previous attached data are replaced with the new ones.

Operation

Complete a Service

ID	CV.WS.SRV.3		
Method	POST		
URL	/genesys/1/cs/services/\${service_id}/end		
Field Name	Type	Mandatory	Description
URI Parameter			
\${service_id}	integer	yes	In the request URI, the unique ID of the service.
Body: Service End Event. The body of this operation corresponds to the Service End Event resource.			
session_id	string	no	The ID of the related session (for instance, the orchestration session or any other business session). Limited to 32 characters.
interaction_id	string	no	The ID of the interaction at the service event's creation. Limited to 50 characters.
application_type	long or string	no	The unique ID associated with the type or class of application which issued the service event. May be used to group related applications, potentially across resource types. Refer to Configuration

ID	CV.WS.SRV.3		
			Options for more details on <i>Business Attribute</i> mapping.
application_id	integer	no	The Genesys DB ID for the application which issues the service event, such as a GVP VoiceXML application, Orchestration SCXML application, etc.
resource_type	long or string	no	The unique ID associated with the type or class of resource providing the service (such as GVP, Agent Desktop, Orchestration). Refer to Configuration Options for more details on <i>Business Attribute</i> mapping.
resource_id	integer	no	The unique DB ID for the specific resource which provides the service. For instance: <ul style="list-style-type: none"> The Genesys DB ID of a specific GVP or orchestration platform The DB ID of a given agent, according to the context.
media_type	long or string	no	The media type applicable to the given service, such as email, voice, chat, etc. Refer to Configuration Options for more details on <i>Business Attribute</i> mapping.
disposition	long or string	no	The unique ID for the business disposition assigned to the given service. For instance, this can be the DB ID of a Business Attribute used to enumerate a given organization's business dispositions.
disposition_desc	string	no	The text description which provides

ID	CV.WS.SRV.3		
			additional context on the business disposition.
timestamp	date/time	no	<p>The UTC time at which the service event was raised, with a precision of milliseconds, using the ISO 8601 representation : [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z.</p> <div style="border: 1px solid orange; padding: 5px; margin-top: 10px;"> <p>Important</p> <p>If the application does not specify this timestamp, the server adds it when the service event is processed.</p> </div>
<extension key>	Any JSON type	no	Service attached data as key-value pairs. You can add as many key-value pairs as needed.

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response	
HTTP code	204
HTTP Title	No Content

Example

Operation

POST http://localhost:8080/genesys/1/cs/services/36981723/end

```
{
  "timestamp": "2009-05-12T12:05:12.145Z",
  "interaction_id": "123ABCAADFJ1259ACF",
  "media_type": 1,
  "resource_id": 5005,
  "resource_type": 2,
  "disposition": 10,
  "disposition_desc": "SUCCESS"
  "Feedback":
  {
    "FeedbackType": "survey",
```

```
    "rating":7,
    "notes":"warm welcome at frontdesk, thanks for the nice trip"
  },
  "Satisfaction": [
    {
      "rating":2,
      "pertinence":8,
      "usefull":true,
      "place":"Terranova mexico resort"
    },
    {
      "rating":8,
      "pertinence":4,
      "usefull":false,
      "place":"Fancy resort Paris"
    }
  ]
}
```

Result

204 No Content

Complete State

POST /genesys/1/cs/services/\${service_id}/states/\${state_id}/end

Description

This operation terminates the state and updates its information with the body content.

Operation

Completes the state

ID	CV.WS.SRV.6		
Method	POST		
URL	/genesys/1/cs/services/\${service_id}/states/\${state_id}/end		
Field Name	Type	Mandatory	Description
URI Parameters			
\${service_id}	integer	yes	The unique 64-bit ID of the related service.
\${state_id}	integer	yes	The 32-bit ID of the state to complete.
Body: State End Event. This body can contain fields from the State End Event resource.			
session_id	string	no	The ID of the related session (for instance, the orchestration session or any other business session). Limited to 32 characters.
interaction_id	string	no	The ID of the interaction at the service event's creation. Limited to 50 characters.
application_type	long or string	no	The unique ID associated with the type or class of application which issued the state event. May be used to group related applications, potentially across resource types. Refer to Configuration Options for more details

ID	CV.WS.SRV.6		
			on <i>Business Attribute</i> mapping.
application_id	integer	no	The Genesys DB ID for the application which issues the state event, such as a GVP VoiceXML application, Orchestration SCXML application, etc.
resource_type	long or string	no	The unique ID associated with the type or class of resource providing the related service (such as GVP, Agent Desktop, Orchestration). Refer to Configuration Options for more details on <i>Business Attribute</i> mapping.
resource_id	integer	no	The unique DB ID for the specific resource which provides the related service. For instance: <ul style="list-style-type: none"> The Genesys DB ID of a specific GVP or orchestration platform The DB ID of a given agent, according to the context.
media_type	long or string	no	The media type applicable to the given state, such as email, voice, chat, etc. Refer to Configuration Options for more details on <i>Business Attribute</i> mapping.
disposition	long or string	no	The unique ID for the business disposition assigned to the given state. For instance, this can be the DB ID of a Business Attribute used to enumerate a given organization's business dispositions. Refer to Configuration Options for more details on

ID	CV.WS.SRV.6		
			<i>Business Attribute mapping.</i>
disposition_desc	string	no	The text description which provides additional context on the business disposition.
timestamp	date/time	no	The UTC time at which the service event was raised, with a precision of milliseconds, using the ISO 8601 representation: [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z. Important If the application does not specify this timestamp, the server adds it when the service event is processed.
<extension key>	Any JSON type	no	Service attached data as key-value pairs. You can add as many key-value pairs as needed.

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response	
HTTP code	200
HTTP message	OK

Example

Operation

The following sample completes a state and updates the associated single-valued extension "Feedback". It assumes that you are using integers for values.

```
POST http://localhost:8080/genesys/1/cs/services/6739/states/5362/end
{
  "interaction_id": "00001a57JGQ00BVS",
  "disposition": 10,
  "disposition_desc": "SUCCESS",
```



```

    "application_type":400,
    "application_id":40,
    "resource_type":200,
    "resource_id":20,
    "media_type":2,
    "Feedback":
    {
      "FeedbackType":"survey",
      "rating":7,
      "notes":"warm welcome at frontdesk, thanks for the nice trip"
    }
  }
}

```

Result

200 OK

Operation

The following sample completes a state and updates the associated single-valued extension "Feedback". It assumes that you configured business values, which allow string values.

```

POST /services/6739/states/5362/end
{
  "interaction_id":"00001a57JGQ00BVS",
  "disposition": 10,
  "disposition_desc": "SUCCESS",
  "application_type":"customer_online_survey",
  "application_id":40,
  "resource_type":"html",
  "resource_id":20,
  "media_type":"webform",
  "Feedback":
  {
    "FeedbackType":"survey",
    "rating":7,
    "notes":"warm welcome at frontdesk, thanks for the nice trip"
  }
}

```

Result

200 OK

Operation

The following sample completes a state and updates the associated single-valued extension "Feedback" and multi-valued extension "Satisfaction".

```

POST http://localhost:8080/genesys/1/cs/services/6739/states/5362/end
{
  "interaction_id":"00001a57JGQ00BVS",
  "disposition": 10,
  "disposition_desc": "SUCCESS",
  "application_type":"customer_online_survey",
  "application_id":40,
  "resource_type":"html",
  "resource_id":20,
  "media_type":"webform",
  "Feedback":
  {
    "FeedbackType":"survey",
    "rating":7,

```

```
    "notes":"warm welcome at frontdesk, thanks for the nice trip"
  },
  "Satisfaction": [
    {
      "rating":2,
      "pertinence":8,
      "usefull":true,
      "place":"Terranova mexico resort"
    },
    {
      "rating":8,
      "pertinence":4,
      "usefull":false,
      "place":"Fancy resort Paris"
    }
  ]
}
```

Result

200 OK

Complete Task

POST /genesys/1/cs/services/{service_id}/tasks/{task_id}/end

Description

This operation completes the task of a given type for the given service. The type is one of the enumerated values recorded for the corresponding business attribute. See [Business Attributes in Context Services](#) for further details.

- UCS assigns auto-incrementing identifiers to tasks, similar to the case of services and tasks.
- States and tasks are assigned 32-bit identifiers.

Operation

ID	CV.WS.SRV.9		
Method	POST		
URL	/genesys/1/cs/services/{service_id}/tasks/{task_id}/end		
Field Name	Type	Mandatory	Description
URI Parameters			
{service_id}	integer	yes	The unique 64-bit ID of the service.
{state_id}	integer	no	The unique 32-bit ID of the state.
{task_id}	integer	yes	The unique 32-bit ID of the task.
Body: Task End Event. This body contains fields from the Task End Event resource.			
session_id	string	no	The ID of the related session, for instance, the orchestration session, or any other business session.
interaction_id	string	no	The ID of the related Genesys interaction. This ID can be used by other Genesys reporting products such as Stat Server, URS, Composer, and GVP.
application_type	long or string	no	The unique ID

ID	CV.WS.SRV.9		
			associated with the type or class of application issuing the service event. May be used to group related applications, potentially across resource types. Refer to Configuration Options for more details on Business Attribute mapping.
application_id	integer	no	The unique ID (such as Genesys DB ID) for the application issuing the event, such as a GVP VoiceXML application or an Orchestration SCXML application.
resource_type	long or string	no	The unique ID associated with the type or class of resource which provides the service (such as GVP, Agent Desktop, Orchestration). Refer to Configuration Options for more details on Business Attribute mapping.
resource_id	integer	no	The unique DB ID for the specific resource which provides the service. For instance: <ul style="list-style-type: none"> the Genesys DB ID of a specific GVP or orchestration platform. the DB ID of a given agent, according to the context.
media_type	long or string	no	The media type applicable to the given task, for instance, email, voice, chat, etc. Refer to Configuration Options for more details on Business Attribute mapping.
disposition	long or string	no	The unique ID for the business disposition assigned to the given

ID	CV.WS.SRV.9		
			service/state/task. For instance, this can be the DB ID of a Business Attribute used to enumerate a given organization's business dispositions. Refer to Configuration Options for more details on Business Attribute mapping.
disposition_desc	string	no	The text description which provides additional context on the business disposition.
timestamp	date/time	no	The UTC time at which the service event was raised, with a precision of milliseconds, using the ISO 8601 representation : [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z. Important If the application does not specify this timestamp, the server adds it when the service event is processed.
<extension key>	Any JSON type	no	Task attached data as key-value pairs. You can add as many key-value pairs as needed.

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response

HTTP code	204
HTTP message	No Content

Example

Operation

```
POST http://localhost:8080/genesys/1/cs/services/95/tasks/3/end
{
  "interaction_id":0000CSS0FXZUP6UR,
  "state_id":24,
  "disposition": 10,
  "disposition_desc": "SUCCESS",
  "Survey":
  {
    "url":"http://ourServer/storage/userAnswers",
    "question1":7,
    "question2":true,
    "question3":"will be better with cable tv and on-demand video"
  },
  "Proposal": [
    {
      "car type":"cabriolet",
      "price":25 000,
      "seats":2,
      "comments":"200 cv, hardtop"
    },
    {
      "car type":"S.U.V.",
      "price":70 000,
      "seats":8,
      "comments":"4wd, leather seats"
    }
  ]
}
```

Result

204 No Content

Composite Start and Bulk Update

Introduced in: 8.5.110

Updated in: 8.5.205, 8.5.206

POST /genesys/1/cs/services_composite/start

Description

Provides a single API call to update or create all Service, State, and Task objects in a single REST call. You can use this request to submit the JSON structures for three objects: Service, State, and Task events. If the Service object's `customer_id`, `service_type`, and `contact_key` do not match an active Service, the system will create one using this data, including State and Task data. Else, within the found service, it will search for the active State and Task respectively matching the `state_type` and the `task_type` specified in the request and it will create them if they are not found.

As a result, Context Services will reply with the IDs of the found and/or created Service, State, and Task resources, and a flag indicating whether they were created or not.

Introduced in: 8.5.206

The `create_new_services` parameter enables you to force the creation of a service record, discarding the previous record if any. See also the [Create New Services](#) example below.

Bulk Update

Introduced in: 8.5.205

To perform a bulk update, use the start query and specify `update_extensions=true` in the start event. Instead of creating a new object, the operation will update the provided extensions of the given objects. This means that only the specified extensions will be updated. For example, if your service's extensions include the Satisfaction extension, and if you do not provide this extension in the query, the extension will neither be updated nor removed from the service.

Auto-complete and Auto-closed Features

Introduced in: 8.5.206

To complete conversation during the update operation, add `auto_closed=true` to the service-level keys in the JSON body. If you wish to complete conversation automatically after some time, you can

use `auto_complete_after=<period of time in seconds>` instead of `auto_closed` in the service-level keys of the JSON body.

Important

You can also auto-complete conversation after some time according to your service configuration. See the `auto-complete-enabled` and `auto-complete-after` options for further details.

See also the [Auto-Complete and Auto-Closed](#) example below.

Operation

POST /genesys/1/cs/services_composite/start			
Field Name	Type	Mandatory	Description
Body Parameters			
services	JSON array	yes	JSON array of Composite Start Events.
create_new_services Introduced in: 8.5.206	boolean	false	If true, force the service creation even if a previous record exists. If false (default), Context Services first checks if an active Service record exists and, if so, updates the service instead of creating a new one.

Composite Start Event

Field	Type	Mandatory	Description
service	JSON structure	yes	Service Start Event.
state Modified in: 8.5.206	JSON structure	no if <code>create_new_services=true</code>	State Start Event. If <code>create_new_services=true</code> , this start event is not required.
task	JSON structure	no	Task Start Event.
associate_state_task	boolean	no	true to associate the Task with the given State; false by default. If you set this parameter to true, it is equivalent to providing the State ID in a

Field	Type	Mandatory	Description
			request to start a Task.
update_extensions Introduced in 8.5.205	boolean	no	true to update the extensions of any provided service, state, or task object.

Response

The Context Management Service API answers every request with HTTP codes. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response	
200 OK	
Header	Location: <code>\${base_uri}/services/\${service_id}/tasks/\${task_id}</code> or <code>\${base_uri}/services/\${service_id}/states/\${state_id}</code> (if the task field was not provided in the request) where: <ul style="list-style-type: none"> • <code>\${base_uri}</code> is the URI of the created Task. • <code>\${task_id}</code> is the Task ID. • <code>\${state_id}</code> is the State ID.
Body	<p>Modified in: 8.5.205</p> <pre>{ service: { "service_id": \${service_id}, "created": "true/false", "updated": "true/false" } state: { "state_id": \${state_id}, "created": "true/false", "updated": "true/false" } task: { "task_id": \${task_id}, "created": "true/false", "updated": "true/false" } }</pre>

Examples

Composite Start Creation

POST /genesys/1/cs/services_composite/start

```

{
  "services" : [
    {
      "service": {
        "interaction_id": "123ABCAADFJ1259ACF",
        "application_type": "App_type_1",
        "application_id": 40,
        "est_duration": 60,
        "customer_id": "${customer_id}",
        "service_type": "MyService1",
        "media_type": "voice",
        "resource_id": 1183,
        "resource_type": "ResType1",
        "RelatedOffersService": [
          {
            "offer_name": "VIP credit card black ed.",
            "type": 9,
            "comments": "proposed to all client"
          },
          {
            "offer_name": "3 times payment GOLD",
            "type": 4,
            "comments": "limited offer"
          },
          {
            "offer_name": "life insurance",
            "type": 3,
            "comments": "healt check to be done before approval"
          }
        ]
      },
      "state": {
        "interaction_id": "123ABCAADFJ1259ACF",
        "application_type": "App_type_1",
        "application_id": 40,
        "resource_type": "ResType1",
        "resource_id": 1183,
        "media_type": "voice",
        "est_duration": 60,
        "state_type": "MyState1",
        "FeedbackState": {
          "FeedbackType": "survey",
          "rating": 7,
          "notes": "warm welcome at frontdesk, thanks for the nice trip"
        },
        "SatisfactionState": [
          {
            "rating": 2,
            "pertinence": 8,
            "usefull": true,
            "place": "Terranova mexico resort"
          },
          {
            "rating": 8,
            "pertinence": 4,
            "usefull": false,
            "place": "Fancy resort Paris"
          }
        ]
      },
      "task": {
        "interaction_id": "123ABCAADFJ1259ACF",

```

```

    "application_type": "App_type_1",
    "application_id": 40,
    "resource_type": "ResType1",
    "resource_id": 1183,
    "media_type": "voice",
    "est_duration": 60,
    "task_type": "MyTask1",
    "FeedbackTask": {
      "FeedbackType": "survey",
      "rating": 7,
      "notes": "warm welcome at frontdesk, thanks for the nice trip"
    },
    "SatisfactionTask": [
      {
        "rating": 2,
        "pertinence": 8,
        "usefull": true,
        "place": "Terranova mexico resort"
      },
      {
        "rating": 8,
        "pertinence": 4,
        "usefull": false,
        "place": "Fancy resort Paris"
      }
    ]
  },
  "associate_state_task": "true"
}
]
}

```

Result

```

HTTP 200 OK
location: /genesys/1/cs/services/111222/tasks/555666
{
  services: [
    {
      service: {
        "service_id": 111222,
        "created": "false",
        "updated": "false",
      },
      state: {
        "state_id": 333444,
        "created": "true",
        "updated": "false"
      },
      task: {
        "task_id": 555666,
        "created": "true",
        "updated": "false"
      }
    }
  ]
}

```

Bulk Update

Introduced in: 8.5.205

POST /genesys/1/cs/services_composite/start

```
{
  "services": [
    {
      "service": {
        "timestamp": "2009-05-12T12:05:12.145Z",
        "interaction_id": "123ABCAADFJ1259ACF",
        "application_type": 400,
        "application_id": 40,
        "est_duration": 60,
        "contact_key": "42",
        "service_type": 100,
        "media_type": 1,
        "resource_id": 5005,
        "resource_type": 2,
        "disposition": 10,
        "coupon": {
          "coupon_name": "DISCOUNTCODE15"
        },
        "satisfaction": {
          "score": 85,
          "agentID": 2025
        }
      },
      "state": {
        "interaction_id": "51",
        "application_type": 400,
        "application_id": 40,
        "resource_type": 200,
        "resource_id": 20,
        "media_type": 2,
        "est_duration": 60,
        "state_type": 100,
        "Feedback": {
          "FeedbackType": "survey",
          "rating": 7,
          "notes": "warm welcome at frontdesk, thanks for the nice trip"
        }
      },
      "task": {
        "interaction_id": 42,
        "est_duration": 460,
        "task_type": 462,
        "Survey": {
          "url": "http://ourServer/storage/userAnswers",
          "question1": 7,
          "question2": true,
          "question3": "will be better with cable tv and on-demand video"
        }
      },
      "associate_state_task": "true",
      "update_extensions": "true"
    }
  ]
}
```

Result

```
HTTP 200 OK
location: /genesys/1/cs/services/111222/tasks/555666
{
  services: [
    {
      service: {"service_id": 111222, "created": "false", "updated": "true" },
      state: {"state_id": 333444, "created": "true", "updated": "false" },
      task: {"task_id": 555666, "created": "true", "updated": "false" }
    }
  ]
}
```

Auto-closed and Auto-complete

Operation

```
POST /genesys/1/cs/services_composite/start
{
  "services": [
    {
      "service": {
        "interaction_id": "123ABCAADFJ1259ACF",
        "application_type":400,
        "application_id":40,
        "est_duration":60,
        "contact_key":"42",
        "service_type":100,
        "media_type": 1,
        "resource_id": 5005,
        "resource_type": 2,
        "auto_closed": true,
        "disposition": 10,
        "coupon": {
          "coupon_name": "DISCOUNTCODE15"
        },
        "satisfaction": {
          "score": 85,
          "agentID": 2025
        }
      },
      "state": {
        "interaction_id":"51",
        "application_type":400,
        "application_id":40,
        "resource_type":200,
        "resource_id":20,
        "media_type":2,
        "est_duration":60,
        "state_type":100,
        "Feedback": {
          "FeedbackType":"survey",
          "rating":7,
          "notes":"warm welcome at frontdesk, thanks for the nice trip"
        }
      },
      "task": {
        "interaction_id":42,
        "est_duration":460,
        "task_type":462,
        "Survey": {

```

```

        "url":"http://ourServer/storage/userAnswers",
        "question1":7,
        "question2":true,
        "question3":"will be better with cable tv and on-demand video"
    }
},
"associate_state_task": "true",
"update_extensions": "true"
},
{
"service": {
    "interaction_id": "123ABCAADFJ1259ACF",
    "application_type":400,
    "application_id":40,
    "est_duration":60,
    "contact_key":"43",
    "service_type":100,
    "media_type": 1,
    "resource_id": 5005,
    "resource_type": 2,
    "auto_complete_after": 3600,
    "disposition": 10,
    "coupon": {
        "coupon_name": "DISCOUNTCODE15"
    },
    "satisfaction": {
        "score": 85,
        "agentID": 2025
    }
},
"state": {
    "interaction_id":"51",
    "application_type":400,
    "application_id":40,
    "resource_type":200,
    "resource_id":20,
    "media_type":2,
    "est_duration":60,
    "state_type":100,
    "Feedback": {
        "FeedbackType":"survey",
        "rating":7,
        "notes":"warm welcome at frontdesk, thanks for the nice trip"
    }
},
"task": {
    "interaction_id":42,
    "est_duration":460,
    "task_type":462,
    "Survey": {
        "url":"http://ourServer/storage/userAnswers",
        "question1":7,
        "question2":true,
        "question3":"will be better with cable tv and on-demand video"
    }
},
"associate_state_task": "true",
"update_extensions": "true"
}
]
}

```

Result

```

HTTP 200 OK
location: /genesys/1/cs/services/111222/tasks/555666
{
  services: [
    {
      service: {"service_id": 111222, "created": "false", "updated": "true" },
      state: {"state_id": 333444, "created": "true", "updated": "false" },
      task: {"task_id": 555666, "created": "true", "updated": "false" }
    }
  ]
}

```

Create New Services

Operation

```

{
  "create_new_services":true,
  "services":[
    {
      "service":{
        "interaction_id":"123ABCAADFJ1259ACF",
        "application_type":"App_type_1",
        "application_id":40,
        "est_duration":60,
        "customer_id":"${customer_id}",
        "service_type":"MyService1",
        "media_type":"voice",
        "resource_id":1183,
        "resource_type":"ResType1",
        "RelatedOffersService":[
          {
            "offer_name":"VIP credit card black ed.",
            "type":9,
            "comments":"proposed to all client"
          },
          {
            "offer_name":"3 times payment GOLD",
            "type":4,
            "comments":"limited offer"
          },
          {
            "offer_name":"life insurance",
            "type":3,
            "comments":"healt check to be done before approval"
          }
        ]
      }
    },
    "state":{
      "interaction_id":"123ABCAADFJ1259ACF",
      "application_type":"App_type_1",
      "application_id":40,
      "resource_type":"ResType1",
      "resource_id":1183,
      "media_type":"voice",
      "est_duration":60,
      "state_type":"MyState1",
      "FeedbackState":{
        "FeedbackType":"survey",
        "rating":7,

```

```

    "notes":"warm welcome at frontdesk, thanks for the nice trip"
  },
  "SatisfactionState":[
    {
      "rating":2,
      "pertinence":8,
      "usefull":true,
      "place":"Terranova mexico resort"
    },
    {
      "rating":8,
      "pertinence":4,
      "usefull":false,
      "place":"Fancy resort Paris"
    }
  ]
},
"task":{
  "interaction_id":"123ABCAADFJ1259ACF",
  "application_type":"App_type_1",
  "application_id":40,
  "resource_type":"ResType1",
  "resource_id":1183,
  "media_type":"voice",
  "est_duration":60,
  "task_type":"MyTask1",
  "FeedbackTask":{
    "FeedbackType":"survey",
    "rating":7,
    "notes":"warm welcome at frontdesk, thanks for the nice trip"
  },
  "SatisfactionTask":[
    {
      "rating":2,
      "pertinence":8,
      "usefull":true,
      "place":"Terranova mexico resort"
    },
    {
      "rating":8,
      "pertinence":4,
      "usefull":false,
      "place":"Fancy resort Paris"
    }
  ]
},
"associate_state_task":"true"
}
]
}

```

Result

```

HTTP 200 OK
location: /genesys/1/cs/services/111222/tasks/555666
{
  services:[
    {
      service:{
        "service_id":111222,
        "created":"false",
        "updated":"false",

```



```
    },
    state:{
      "state_id":333444,
      "created":"true",
      "updated":"false"
    },
    task:{
      "task_id":555666,
      "created":"true",
      "updated":"false"
    }
  ]
}
```

Delete Service By ID

DELETE /services/\${service_id}

Description

Enables your application to delete a service if its ID is known.

Important

Your application needs a WRITE role to perform this request.

Operation

Method	DELETE		
URL	/genesys/1/cs/services/\${service_id}		
Field Name	Type	Mandatory	Description
URI Parameters			
\${service_id}	integer	yes	The service ID.

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response	
HTTP code	200
HTTP message	OK

<references />

Example

Operation The following operation deletes the given service with its active states and tasks.

```
DELETE http://localhost:8080/genesys/1/cs/services/361-a3793cef-d559-4f9f-a95d-64555a4ba371
```

Result

```
200 OK
```

In case of error, the result would have been:

```
400 Bad Request
{
  "message": "There are no service found for : 361-a3793cef-d559-4f9f-a95d-64555a4ba371",
  "exception": "com.genesyslab.gsg.services.contextservices.exceptions.NotFoundException"
}
```

Export Services to JSON Stream by Date

GET /maintenance/services/stream

Description

Your application can use this operation to export service data to a JSON stream for a given time range. You can filter the list of returned services according to their date, inner state, or even events. The service information is returned in the response's body and can include service information in addition to service IDs.

See also [the export features page](#) for further details about filters and usage of this query.

Important

According to the amount of data to return, in particular, if you include service information, this query can take long to provide your application with a response. In this case, you should rather export a list of service IDs, then use the standard [Query Service by ID](#) to retrieve information.

Operation

Export services to a JSON stream

Method	GET		
URL	/genesys/1/cs/maintenance/services/stream		
Name	Type	Mandatory	Description
URI Parameters			
time_from	date/time[1]	yes	Filters the service IDs which match the event and/or state filters starting from the given date/time. Example : 2014-09-02T08:00:00.000Z
time_to	date/time[1]	no	Filters the service IDs which match the event and/or state filters ending to the given

Method	GET		
			<p>date/time.</p> <p>By default, current time is taken as end range. Example : 2014-09-02T08:00:00.000Z</p>
filter_events	any, started, completed	no	<p>Filters on either types of service event occurrence.</p> <ul style="list-style-type: none"> any (default): The response includes the services for which an event occurred in the given time range. started: The response includes services for which a start event occurred during the given time range. completed: The response includes services for which a complete event occurred during the given time range.
export_content	boolean	no	<ul style="list-style-type: none"> "true" to include all service data in the JSON response. "false" (default) to include service IDs only.
filter_state	any, active, inactive	no	<div data-bbox="1153 1381 1429 1535" style="border: 1px solid orange; padding: 5px; margin-bottom: 10px;"> <p>Important</p> <p>You can use this parameter only if the export_content parameter is set to true.</p> </div> <ul style="list-style-type: none"> any (default): The response includes the service information for all of the selection. active: The response includes the conversation (or

Method	GET		
			<p>service information) if some event (specified by the filter_events parameter) occurred during the specified time range AND if the service is still active at the time of the request.</p> <ul style="list-style-type: none"> inactive: The response includes the conversation (or service information) if some event (specified by the filter_events parameter) occurred during the specified time range AND if the conversation is completed at the time of the request.

<references />

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response	
HTTP code	200
HTTP Title	OK
Body	String[]

Example

Operation

The following code sample associates the customer with a service, and specifies attached data as json arrays for "Feedback" and "Satisfaction".

```
GET http://localhost:8080/genesys/1/cs/maintenance/services/stream?time_from=2014-07-22T12:00:00.000Z&time_to=2014-07-22T18:00:00.000Z&export_content=false
```

Result

200 OK

```
[
  "361-93ae6d6b-8ea6-476a-bc2f-235ddc029994",
  "361-b552fb93-4b10-4761-9d50-850890bcab02",
  "361-0aaa7023-2db6-4fd3-8cf6-8e32619a0df8",
  "361-0b6b586f-1815-47b8-8a84-16d66162fff1c",
  "361-96d081b2-1918-46a5-b497-c235f882b33f",
  "361-9131cb94-2829-4fb0-9ae2-f9fd53639d1c",
  "361-5a4db9b7-8ee4-4848-97e7-7144cabe6790",
  "361-acf20476-d938-4583-b3cc-8b1568309247",
  "361-c1f24f0d-3e74-4194-90bb-5d330a9614c8",
  "361-cdd9bf74-4fb4-403a-aeb3-0d8875f963bb",
  "361-1ef7641d-6003-4210-b4d0-cc1cafa8411d",
  "361-1ee58ee2-4b54-4d22-8cbc-d000363b2282",
  "361-3ab5e8bc-adc1-494a-a4c6-b925da319dcc",
  "361-6fee8b55-607b-4731-aba5-6681a992d184",
  "361-ddd8e44a-3fdc-4da8-91a7-c709ba654877",
  "361-b021209c-f86c-4ea5-b991-51d6b848d920",
  "361-66808d00-6328-4332-a4c7-936094babdb6",
  "361-fe54ea55-f7e9-44d1-aafd-0a213a9b1bd1",
  "361-604b6ad4-308d-4d60-8a2a-ba59cc3d84cd",
  "361-c4925764-cef0-4ce0-b4d7-4e96023d1c62",
  "361-7d8caf1f-0136-483e-bfef-bf2257e4a25c",
  "361-a222846a-67f5-47e5-b5c9-d5c88bd2775b",
  "361-779d75da-7d64-4ffd-808b-9ccd6cf39a66"
]
```

Export Services to File by Date

POST /maintenance/services/files

Description

Your application can use this operation to export service data to the JSON or CSV file format for a given time range. You can filter the list of returned services according to their date, inner state, or events. The query creates a file per type of data exported (service, states, tasks), for instance:

- "C:/export20140813_160508_183_services.csv"
- "C:/export20140813_160508_183_states.csv"
- "C:/export20140813_160508_183_tasks.csv"

Important

This operation includes the service data for each resources returned.

See also [the export features page](#) for further details about the state and event filters.

Operation

Export services to files

Method	POST		
URL	/genesys/1/cs/maintenance/services/files		
Name	Type	Mandatory	Description
URI Parameters			
time_from	date/time[1]	yes	Filters the service IDs which match the event and/or state filters starting from the given date/time. Example : 2014-09-02T08:00:00.000Z
time_to	date/time[1]	no	Filters the service IDs which match the event and/or state filters

Method	POST		
			ending to the given date/time. By default, current time is taken as end range. Example : 2014-09-02T08:00:00.000Z
filter_events	any, started, completed	no	Filters on either types of service event occurrence. <ul style="list-style-type: none"> • any (default): The response includes the services for which an event occurred in the given time range. • started: The response includes services for which a start event occurred during the given time range. • completed: The response includes services for which an end event occurred during the given time range.
filter_state	any, active, inactive	no	<ul style="list-style-type: none"> • any (default): The response includes the conversation (or service information) if some event (specified by the filter_events parameter) occurred during the specified time range. • active: The response includes the conversation (or service information) if some event (specified by the filter_events parameter) occurred during the specified time range AND if the service is still active.

Method	POST		
			<ul style="list-style-type: none"> inactive: The response includes the conversation (or service information) if some event (specified by the filter_events parameter) occurred during the specified time range AND if the service is completed.
file_prefix	String	yes	<p>Specifies the file destination path and prefix. For example, if you set file_prefix to 'C:/export', the server generates the following one or three files:</p> <ul style="list-style-type: none"> "C:/export20140813_160508_183_s... "C:/export20140813_160508_183_st... "C:/export20140813_160508_183_ta...
compression	String	no	<p>Values:</p> <ul style="list-style-type: none"> uncompressed (default) zip gzip <p>If the compression is enabled, the server generates the export file with the accurate file extension.</p>
separator	String	no	<p>Specifies the separator used in the exported CSV file to separate the column data in a CSV file line.</p> <ul style="list-style-type: none"> Default: ","
format	json or csv	no	<p>Specifies the format of the exported file. Default value is: csv.</p>

<references />

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response	
HTTP code	200
HTTP Title	OK
Body	String[]

Example

Operation

The following code sample associates the customer with a service, and specifies attached data as json arrays for "Feedback" and "Satisfaction".

```
POST http://localhost:8080/genesys/1/cs/maintenance/services/
file?time_from=2014-07-22T12:00:00.000Z&time_to=2014-07-22T18:00:00.000Z&file_prefix=C:/export&format=csv
```

Response

202 Accepted

C:/export_20140902_125024_170_services.CSV file

```
377-fffa514f-75b4-45b8-a417-550337719328,9,111861477473,,true,692,3,7PH0C3M0MF,4,5VFB3U7F9F,3,689,5,0R7H63Q24D9
mexico resort,4,false,8,Fancy resort Paris,2015-03-17
377-aa4abd74-db0e-4026-9adc-01fea18978b6,3,111639426401,,true,928,6,0F80DRNFV0,7,PUCPBS2MCP,2,481,9,UY05AV5G2KF
mexico resort,4,false,8,Fancy resort Paris,2015-03-17
377-ee9920b1-32a6-4900-b8be-
b3f7ed328a6d,1,112956879726,,true,713,1,KER4FX089X,10,GNAYQTYRJA,2,256,7,Y61ED4QHTDK6AQ80,1,2014-09-02T12:29:58
mexico resort,4,false,8,Fancy resort Paris,2015-03-17
service_id,service_type,contact_key,customer_id,started_anonymous,est_duration,disposition,disposition_desc,sta
```

C:/export_20140902_125024_170_states.CSV file

```
377-ee9920b1-32a6-4900-b8be-
b3f7ed328a6d,377-c47423ca-4aed-458b-8fde-49b8435e7953,6,,25,5,K0GQKRSYGG,9,1845768615,1,256,9,112956879726,3,20
is the weather?,http://survey.genesys.com
377-ee9920b1-32a6-4900-b8be-b3f7ed328a6d,377-3b58e66c-
f499-4765-a1ff-860e8b2d12e2,2,,6,9,7YHBF2VX0G,9,1845768615,5,256,4,112956879726,3,2014-09-02T12:29:58.551Z,1845
is the weather?,http://survey.genesys.com
377-ee9920b1-32a6-4900-b8be-b3f7ed328a6d,377-c7ec4edf-
f4d6-4b76-8d99-aae94b9e92fd,7,,54,10,FPW0YDY8HW,6,1845768615,2,256,3,112956879726,3,2014-09-02T12:29:58.863Z,18
is the weather?,http://survey.genesys.com
377-ee9920b1-32a6-4900-b8be-
b3f7ed328a6d,377-fd30f203-3b62-465a-90e6-d2f6c7332c78,1,,55,3,XPJH03AV37,7,1845768615,5,256,5,112956879726,4,20
is the weather?,http://survey.genesys.com
377-ee9920b1-32a6-4900-b8be-b3f7ed328a6d,377-c8503d05-784d-4037-aadc-
e70874a5c053,10,,35,5,C6K8NTU2UC,9,1845768615,3,256,8,112956879726,4,2014-09-02T12:29:59.178Z,1845768615,2014-0
is the weather?,http://survey.genesys.com
service_id,state_id,state_type,previous_state_id,est_duration,disposition,disposition_desc,start_resource_type,
```

C:/export_20140902_125024_170_tasks.CSV file

```
377-ee9920b1-32a6-4900-b8be-b3f7ed328a6d, ,377-e438be2b-c3a8-4c2d-aa2b-018cc860df88,3,65,9,KC8TNWEQK8,5,1845768615,3,256,10,112956879726,5,2014-09-02T12:29:59.850Z,1845768615,20
377-ee9920b1-32a6-4900-b8be-b3f7ed328a6d, ,377-931a6f65-2f76-4675-b892-492c24680a7d,5,16,10,Y15M9U1SVB,8,1845768615,3,256,8,112956879726,10,
377-ee9920b1-32a6-4900-b8be-b3f7ed328a6d, ,377-ebff6032-1306-461b-9234-99315d836eed,9,57,7,VXF9BW8YJ9,2,1845768615,4,256,9,112956879726,7,20
377-ee9920b1-32a6-4900-b8be-b3f7ed328a6d, ,377-bfbabd29-3384-4926-b275-43e0f538c454,2,97,8,SPYF4CB6PT,9,1845768615,5,256,7,112956879726,10,2
377-ee9920b1-32a6-4900-b8be-b3f7ed328a6d, ,377-378ae588-d76b-4567-8557-e976032e2d0c,2,46,10,058GAT2KSY,6,1845768615,4,256,6,112956879726,1,2
377-ee9920b1-32a6-4900-b8be-b3f7ed328a6d, ,377-f63d0dc7-4ac2-4135-ad4c-d0b23aae7395,3,47,2,9C1QDWKQDF,7,1845768615,4,256,5,112956879726,5,2014-09-02T12:29:59.947Z,1845768615,2014-09-
377-ee9920b1-32a6-4900-b8be-b3f7ed328a6d, ,377-9f83dbf4-e9cc-4a47-9531-f6d9dcba9595,7,48,3,B7N0C0Q8US,5,1845768615,5,256,7,112956879726,10,2
377-ee9920b1-32a6-4900-b8be-b3f7ed328a6d, ,377-67ce4f33-16db-47c1-adb9-3eb61b25e795,7,55,1,MNX8V1U2KT,3,1845768615,2,256,9,112956879726,4,20
service_id,state_id,task_id,task_type,est_duration,disposition,disposition_desc,start_resource_type,start_sessi
```

Merge Extension Data

- **PUT** `/genesys/1/cs/services/${service_id}/extensions/${ext_name}/merge`
- **PUT** `/genesys/1/cs/services/${service_id}/states/${state_id}/extensions/${ext_name}/merge`
- **PUT** `/genesys/1/cs/services/${service_id}/tasks/${task_id}/extensions/${ext_name}/merge`

Added in: 8.5.110

Description

Updates an extension with the provided key-value pair or creates a new one if the extension does not exist in the given Service, State, or Task. If the extension exists, the system will add the new key-value pair without overwriting the extension's content.

Tip

Use this API to update extensions without reading them first and then calling one of the regular update [Update Service Extension](#), [Update State Extension](#), and [Update Task Extension](#) APIs.

Operation

Merge or create Service, State, Task extension data

Field Name	Type	Mandatory	Description
<ul style="list-style-type: none"> • PUT <code>/genesys/1/cs/services/\${service_id}/extensions/\${ext_name}/merge</code> • PUT <code>/genesys/1/cs/services/\${service_id}/states/\${state_id}/extensions/\${ext_name}/merge</code> • PUT <code>/genesys/1/cs/services/\${service_id}/tasks/\${task_id}/extensions/\${ext_name}/merge</code> 			
URI Parameters			
<code>\${service_id}</code>	Integer	yes	The ID of the service.
<code>\${state_id}</code>	Integer	yes	The ID of the state.
<code>\${task_id}</code>	Integer	yes	The ID of the task.
<code>\${ext_name}</code>	String	yes	The extension name.

Field Name	Type	Mandatory	Description
Body Parameters			
<extension> or <extension>[]	JSON	yes	<p>The attached data to merge as key-value pairs or as an array of key-value pairs.</p> <p>You can merge maps, such as:</p> <pre>"\${ext_name}": { "offer_name": "VIP credit card black ed.", "type": 9, "comments": "proposed to all client" }</pre> <p>This query will not work with extension lists such as:</p> <pre>{ext_name}": [{ "offer_name": "VIP credit card black ed.", "type": 9, "comments": "proposed to all client" }, { "offer_name": "VIP credit card black ed.", "type": 9, "comments": "proposed to all client" }]</pre>

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response
200 OK

Example

Operation to update a service extension

In this scenario, the first PUT request creates an extension, then the POST request merges its extension data with new data, then the GET request queries the service.

```
PUT /genesys/1/cs/services/8389/extensions/agent{
```

```
"agentID": 2025,
"score": 85,
"extensions": [
  "2281",
  "2284"
],
"location": "Toronto"
}
```

```
PUT /genesys/1/cs/services/8389/extensions/agent/merge
{ "extensions": [ "228144", "228444" ], "location": "Markham"}
```

Result

```
GET /genesys/1/cs/services/8389?extensions=agent
HTTP 200 OK{
  "service_id": 3005,
  "service_type": 100,
  "est_duration": 300,
  "agent": {
    "agentID": 2025,
    "score": 85,
    "extensions": [
      "228144",
      "228444"
    ]
  }
  "location": "Markham"
}
```

Perform State Transition

POST /genesys/1/cs/services/\${service_id}/states/transition

Description

Performs a state transition for a given service. In many cases, your application is responsible for completing a given state and starting a new state, both in the context of a single interaction. In that context, your application can use this operation (instead of two separate **Start State** and **Complete State** operations) to reduce both the data duplication and the overall UCS services workload.

Operation

ID	CV.WS.SRV.7		
Method	POST		
URL	/genesys/1/cs/services/\${service_id}/states/transition		
Field Name	Type	Mandatory	Description
URI Parameters			
service_id	integer	yes	The unique ID of the related service.
Body: State Transition Event			
This body contains fields from the State Transition Event .			
from	hash	yes	<p>The hash of the following fields describing the state from which the service is transitioning:</p> <ul style="list-style-type: none"> state_id (mandatory) : The 32-bit ID of the state to complete. disposition (optional): The unique ID for the business disposition assigned to the given state. For instance, this can be the DB ID of a

ID	CV.WS.SRV.7		
			<p>Business Attribute used to enumerate a given organization's business dispositions.</p> <p>Refer to Configuration Options for more details on Business Attribute mapping.</p> <ul style="list-style-type: none"> disposition_desc (optional): The text description which provides additional context on the business disposition. Limited to 64 characters.
to	hash	yes	<p>The hash of the following fields, which describe the state into which the service is transitioning:</p> <ul style="list-style-type: none"> state_type (optional): The unique ID associated with the type of service, typically the DB ID of a value in the Service Type Business Attribute. est_duration (optional): The estimated service duration in seconds.
<p>Important The following fields apply both to the state to complete and the state to start.</p>			
session_id	string	no	<p>The ID of the related session. For example, the orchestration session or any other business session.</p>
interaction_id	string	no	<p>The ID of the related Genesys interaction. This ID can be used by other Genesys reporting products such as Stat Server, URS, Composer, and GVP.</p>

ID	CV.WS.SRV.7		
application_type	long or string	no	The unique ID associated with the type or class<ref name="business"/> of application issuing the service event. May be used to group related applications, potentially across resource types.
application_id	integer	no	The unique ID (Genesys DB ID) for the application issuing the service event, such as a GVP VoiceXML application or an Orchestration SCXML application.
resource_type	long or string	no	The unique ID associated with the type or class<ref name="business"/> of resource which provides the service (such as GVP, Agent Desktop, or Orchestration).
resource_id	integer	no	<p>The unique DB ID for the specific resource which provides the service. For instance:</p> <ul style="list-style-type: none"> the Genesys DB ID of a specific GVP or orchestration platform. the DB ID of a given agent, according to the context.
media_type	long or string	no	The media type applicable to the given state. For example: email, voice, chat, etc.
timestamp	date/time	no	<p>The UTC time at which the event was raised, with a precision of milliseconds, using the ISO 8601 representation : [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z.</p> <p>Important</p>

ID	CV.WS.SRV.7		
			If the application does not specify this timestamp, the server does it when the event is processed.
<extension key>	Any JSON type	no	State attached data as key-value pairs. You can add as many key-value pairs as needed.

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response

HTTP code	201
HTTP message	Created
Header	Location: /service/{service_id}/states/{state_id} where: <ul style="list-style-type: none"> • {service_id} is the service ID. • {state_id} is the ID of the created state.
Body	{"state_id": {state_id}} where: {state_id} is the created state ID.

Example

Operation

```
POST http://localhost:8080/genesys/1/cs/services/735692/states/transition
{
  "timestamp": "2009-05-07T12:05:20.157",
  "session_id": "11000ABC-80236C1A-1010",
  "interaction_id": "123ABC908ABFFD8080",
  "from": {
    "state_id": 1001,
    "disposition": 1,
    "disposition_desc": "SUCCESS",
    "Feedback": {
      "FeedbackType": "survey", "rating": 7,
      "notes": "warm welcome at frontdesk, thanks for the nice trip"
    }
  }
}
```

```

    },
    "Satisfaction": [
    {
      "rating":2,
      "pertinence":8,
      "usefull":true,
      "pplace":"Terranova mexico resort"
    },
    {
      "rating":8,
      "pertinence":4,
      "useful":false,
      "pplace":"Fancy resort Paris"
    }
  ]
},
"to": {
  "state_type": 8,
  "est_duration": 500,
  "Sponsoring": { "Rank":"first","expire":7,
    "notes":"give customer free meal" }
}
}

```

POST http://localhost:8080/genesys/1/cs/services/735692/states/transition

```

{
  "timestamp": "2009-05-07T12:05:20.157",
  "session_id": "11000ABC-80236C1A-1010",
  "interaction_id": "123ABC908ABFFD8080",
  "from": {
    "state_id": 1001,
    "disposition": 1,
    "disposition_desc": "SUCCESS",
    "Feedback": { "FeedbackType":"survey","rating":7,
      "notes":"warm welcome at frontdesk,
thanks for the nice trip" },
    "Sponsoring": { "Rank":"first","expire":7,
      "notes":"give customer free meal" }
  },
  "to": {
    "state_type": 8,
    "est_duration": 500,
    "Satisfaction": [
      {
        "rating":7,
        "pertinence":3,
        "usefull":false,
        "pplace":"Australian beach resort complex"
      },
      {
        "rating":9,
        "pertinence":8,
        "usefull":true,
        "pplace":"Caribbean beach complex"
      }
    ]
  }
}
}

```

Result

```

201 Created
{"state_id":15158}

```

Purge Services

POST /maintenance/purge

Description

Purge the services which received a started and/or a completed event prior to a limit date. Use this query if your application needs to remove old services from the database.

Important

Your application needs a WRITE role to perform this request.

Operation

Method	POST		
URL	/genesys/1/cs/maintenance/purge		
Field Name	Type	Mandatory	Description
URI Parameters			
limit	String	yes	Specifies the time limit for the service deletion. Example : 2014-09-02T08:00:00.000Z
operation	String	yes	Specifies the type of operation to be performed when the purging process runs. <ul style="list-style-type: none"> purge.service.all to purge all the services which received a started event prior to the limit date. purge.service.started.anonymous to purge the anonymous services which received a

Method	POST
	<p>started event prior to the limit date and are not completed at the date of the purge.</p> <ul style="list-style-type: none"> • <code>purge.service.started</code> to purge the services which received a started event prior to the limit date and are not completed at the date of the purge. • <code>purge.service.completed</code> to purge the services which received a started event prior to the limit date and are completed at the date of the purge. • <code>purge.service.completed.anonymous</code> to purge the anonymous services which received a started event prior to the limit date and are completed at the date of the purge.

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response	
HTTP code	200
HTTP message	OK

<references />

Example

Operation The following operation deletes the given service with its active states and tasks.

```
POST http://localhost:8080/genesys/1/cs/maintenance/purge
{
  "limit": "2014-09-26T16:50:00.000Z",
  "operation": "purge.service.all"
}
```

Result

200 OK

Query Business Attribute Schema

Important

Prerequisites: You need to **enable** profiles in UCS.

GET /metadata/business-attributes

Description

Returns the schema for the list of available business attributes.

Operation

ID	CV.WS.CONFIG.BA		
Method	GET		
URL	<ol style="list-style-type: none"> All: /metadata/business-attributes /metadata/business-attributes/{business-attribute-name} /metadata/business-attributes?cv-attribute={attribute-full-name} 		
Parameter	Type	Mandatory	Description
{business-attribute-name}	string	no	The name of the business attribute, in order to retrieve its schema only.
{attribute-full-name}	string	yes if ?cv-attribute= is part of the URL	The full name of a Context Services' business attribute in order to retrieve its schema. Examples: <ul style="list-style-type: none"> Service.type, Task.disposition, State.media_type

<references />

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

HTTP code		200	
HTTP message		OK	
Parameter	Type	Mandatory	Description
Body Business Attribute or BusinessAttribute[]			
name	string	yes	The business attribute's name.
dbid	string	yes	The business attribute's database ID.
display_name	string	yes	The business attribute's display name.
description	string	yes	The description.
map_name	boolean	no	<ul style="list-style-type: none"> • true means that the Business Attribute Value Names is returned instead of DB IDs in the responses for GET operations. • false means that DB IDs are returned in the responses for GET operations. See also [business-attributes] Section.
cv_attributes	string[]	no	The associated Context Services attributes.
values	BusinessAttribute[]	no	Nested possible attributes values.

Example

Operation

```
GET /metadata/business-attributes/MediaType
```

Result

```
200 OK {
{
```

```
"display_name" : "Media Type",
"values" : [ {
  "display_name" : "email",
  "description" : "Media EMail",
  "name" : "email",
  "dbid" : 1003
}, {
  "display_name" : "chat",
  "description" : "Media Chat",
  "name" : "chat",
  "dbid" : 1006
}, {
  "display_name" : "voice",
  "description" : "Media Voice",
  "name" : "voice",
  "dbid" : 1001
}, {
  "display_name" : "fax",
  "description" : "Media Fax",
  "name" : "fax",
  "dbid" : 1014
}],
"description" : "Media type identifier",
"name" : "MediaType",
"dbid" : 1001,
"map_name" : false,
"cv_attributes" : [ "Service.media_type", "State.media_type", "Task.media_type" ]
}
}
```

Query Service by ID

GET /services/\${service_id}

Description

Enables your application to query a service if its ID is known. By default, no tasks or states are returned. The example below shows how your application can use optional parameters to customize its request to specify which nested tasks and states of the service should be part of the response.

Operation

ID	CV.WS.SRV.14		
Method	GET		
URL	/genesys/1/cs/services/\${service_id}		
Field Name	Type	Mandatory	Description
URI Parameters			
\${service_id}	integer	yes	The service ID.
active_states	bool	no	<ul style="list-style-type: none"> "true" to include information on this service's active states in the results "false" (default)
completed_states	bool	no	<ul style="list-style-type: none"> "true" to include information on this service's completed states in the results "false" (default)
active_tasks	bool	no	<ul style="list-style-type: none"> "true" to include information on this service's active tasks in the results "false" (default)

ID	CV.WS.SRV.14		
completed_tasks	bool	no	<ul style="list-style-type: none"> "true" to include information on this service's completed tasks in the results "false" (default)
extensions	string	no	<p>List of extension names separated with commas, which will be returned with the service. For example:</p> <p>such as <code>http://localhost:8010/genesys/1/cs/services/383-f9a62aec-e6f2-45cb-aa01-0d9404f905bd?active_states=true</code></p> <p>.</p> <p>You can also prefix the extension name with the a "State" or "Task" entity name followed by dot and extension name. For example:</p> <p><code>http://localhost:8010/genesys/1/cs/services/383-f9a62aec-e6f2-45cb-aa01-0d9404f905bd?active_states=true</code></p> <p>Alternatively, all extensions can be returned with specifying <code>extensions=*</code>, or simply entity extensions with specifying for example <code>extensions=State</code>.</p>

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response

HTTP code	200		
HTTP message	OK		
Field Name	Type	Mandatory	Description

HTTP code	200		
Body: Service. This response includes the representation of the Service associated with the service ID.			
service_type	long or string	yes	The unique ID associated with the service type, typically the DB ID of a value in the Service Type Business Attribute. Refer to Configuration Options for more details about <i>Business Attribute</i> mapping.
service_id	integer	yes	The service's unique ID. Only top-level objects include this field. For instance, if your application retrieves a service which includes nested states and tasks, only the top-level service object contains this ID.
started	Service Start Event	yes	Start event (see Start Service). This event does not include the "est_duration" field, if "est_duration" is specified at the service level of the response.
completed	Service End Event	no	The related end event if the service is completed (see Complete Service). This event does not include the "disposition" and "disposition_desc" fields if they are specified at the service level of the response.
customer_id	string	no	The customer's unique ID. If the service has no customer ID, the service is Anonymous Service . Only top-level objects include this field. For instance, if your application retrieves a service which includes nested states and tasks, only the top-level service object contains this ID.
est_duration	integer	no	The estimated time for completing the service, in seconds.

HTTP code	200		
duration	integer	no	The total duration in milliseconds. Important Only relevant for completed services.
disposition	integer	no	ID of the Business disposition ID. Important Only relevant for completed services.
disposition_desc	string	no	The reason for assigning the business disposition with the service element. Limited to 256 characters.
active_states	State[]	no	The array of the started states which are not completed. The field is returned if the corresponding option was set to true in the request.
completed_states	State[]	no	The array of completed states through which the service has made transitions. The field is returned if the corresponding option was set to true in the request.
active_tasks	Task[]	no	The array of the started tasks which are not completed. The field is returned if the corresponding option was set to true in the request.
completed_tasks	Task[]	no	The array of the tasks which are completed as part of the service delivery. The field is returned if the corresponding option was set to true in the request.
interactions	Interaction[]	no	The array of the interactions which are associated with this service. The field is

HTTP code	200		returned if the corresponding option was set to true in the request.
<extension key>	Any JSON type	no	Service attached data as key-value pairs. You can add as many key-value pairs as needed.

Example

Operation The following operation retrieves the given service (ID=2000) with its active states and tasks to resume the activity.

```
GET http://localhost:8080/genesys/1/cs/services/3005?active_states=trueCleted_states=true~tive_tasks=trueCleted_tasks=true&extensions=ClientInfo,relatedOffers
```

Result

```
{
  "completed_tasks" : [ {
    "service_id" : 3005,
    "duration" : 82937,
    "disposition_desc" : "normal ending",
    "task_id" : 2001,
    "task_type" : 55,
    "est_duration" : 540,
    "started" : {
      "timestamp" : "2010-06-03T08:49:53.053Z",
      "interaction_id" : "587"
    },
    "disposition" : 5,
    "completed" : {
      "timestamp" : "2010-06-03T08:51:15.990Z",
      "interaction_id" : "587"
    }
  } ],
  "service_id" : 3005,
  "active_states" : [ {
    "service_id" : 3005,
    "state_id" : 4000,
    "state_type" : 100,
    "est_duration" : 60,
    "started" : {
      "timestamp" : "2010-06-03T08:48:18.257Z",
      "application_type" : 400,
      "resource_id" : 20,
      "media_type" : 2,
      "resource_type" : 200,
      "application_id" : 40,
      "interaction_id" : "51"
    }
  } ],
  "ClientInfo" : {
    "userAgent" : "Mozilla/5.0 (Windows; U; Windows NT 5.1; fr; rv:1.9.2) Gecko/20100115
```

```

Firefox/3.6 (.NET CLR 3.5.30729)",
  "clientId" : "192.168.1.1",
  "contentType" : "Content-Type : application/json;charset=UTF-8"
},
"active_tasks" : [ {
  "service_id" : 3005,
  "state_id" : 4001,
  "task_id" : 2000,
  "task_type" : 55,
  "est_duration" : 540,
  "started" : {
    "timestamp" : "2010-06-03T08:49:45.943Z",
    "interaction_id" : "587"
  }
} ],
"service_type" : 100,
"est_duration" : 300,
"started" : {
  "timestamp" : "2010-09-07T07:58:16.313Z",
  "application_type" : 400,
  "resource_id" : 10,
  "media_type" : 2,
  "resource_type" : 200,
  "application_id" : 40,
  "interaction_id" : "56"
},
"completed_states" : [ {
  "service_id" : 3005,
  "state_id" : 4001,
  "duration" : 182907,
  "disposition_desc" : "normal ending",
  "state_type" : 200,
  "est_duration" : 300,
  "started" : {
    "timestamp" : "2010-06-03T08:48:51.473Z",
    "application_type" : 400,
    "resource_id" : 6000,
    "media_type" : 3,
    "resource_type" : 100,
    "application_id" : 40,
    "interaction_id" : "8001"
  },
  "disposition" : 5,
  "completed" : {
    "timestamp" : "2010-06-03T08:51:54.380Z",
    "interaction_id" : "1587"
  }
} ],
"relatedOffers" : [ {
  "offer_name" : "VIP credit card black ed.",
  "type" : "9",
  "comments" : "proposed to all client"
}, {
  "offer_name" : "3 times payment GOLD",
  "type" : "4",
  "comments" : "limited offer"
}, {
  "offer_name" : "life insurance",
  "type" : "3",
  "comments" : "healt check to be done before approval"
} ],
"contact_key" : "bob"
}

```


Additional comments:

- The service=2000 does not contain completed values. It is an active service.
- There was no active states for this service.
- There is a single active task.

Query Services

Modified in 8.5.111

- **GET /genesys/1/cs/services/anonymous/\${contact_key}**
- **GET /genesys/1/cs/customers/\${customer_id}/services**

Description

The operations on this page allow users to query for active and/or completed services, either for a specific customer or for anonymous services matching a contact key. The latter case primarily addresses service resumption use cases for services in which the customer is not yet identified. For example, consider the following:

- The customer calls in from phone number 408-832-7799.
- The application starts a service, noting contact key "408-832-7799" for further reference.
- The application is not able to identify the caller based on the phone number.
- IVR identifies that the customer is calling to set up a new wireless account.
- The customer is placed in an agent queue. After 10 minutes, the call is dropped.
- The customer calls back in from the same number. The application still cannot identify the customer based on that number.
- The application looks up whether there are any outstanding services with identification token "408-832-7799"--one is returned.
- The application asks the customer whether he or she is calling back in regards to the previous request or for a new wireless request. Upon confirmation, the customer is placed back in the agent queue.

Operations

Query Anonymous Services

Important

This operation queries **anonymous services** only.

ID	CV.WS.SRV.10		
Method	GET		
URI	<ol style="list-style-type: none"> Active: /genesys/1/cs/services/anonymous/{contact_key}/active Completed: /genesys/1/cs/services/anonymous/{contact_key}/completed All: /genesys/1/cs/services/anonymous/{contact_key} 		
Field Name	Type	Mandatory	Description
URI Parameters			
{contact_key}	string	yes	The "contact key" is supplied at the service creation if the service is started before the customer's identification. This key can be used to look up the service with this method, or for further association once the customer is identified (see Associate Service).
started_from	date/time	no	Filters the services started on or after the given date/time given in ISO 8601 representation: [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z
started_to	date/time	no	Filters the services started prior to the given date/time in ISO 8601 representation: [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z.
completed_from	date/time	no	Filters the services that were completed on or after the given date/time in ISO 8601 representation: [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z. Only applicable when querying service history.
completed_to	date/time	no	Filters the services completed prior to the given date/time. Only applicable when querying service history.

ID	CV.WS.SRV.10		
			Date given in ISO 8601 representation: [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z
service_types	integer	no	Filters the services by the application-defined service type. Refer to Configuration Options for more details on Business Attribute mapping. Your application can specify several types separated by a comma, as follows: <i>service_types=1,3,45</i>
active_states	bool	no	<ul style="list-style-type: none"> "true" to include information on the service's active states in the results. "false" by default.
active_tasks	bool	no	<ul style="list-style-type: none"> "true" to include information on the service's active tasks in the results. "false" (default).
completed_states	bool	no	<ul style="list-style-type: none"> "true" to include information on the service's completed states in the results. "false" (default).
completed_tasks	bool	no	<ul style="list-style-type: none"> "true" to include information on the service's completed tasks in the results. "false" (default).
max_services Added in: 8.5.111	integer	no	Maximum number of services to return.
<extension key>	Any JSON type	no	Service attached data as key-value pairs. You can add as many key-value pairs as needed.

Query Customer Services

Important

This operation queries services which are not anonymous, that is, services which are associated with a customer id.

ID	CV.WS.SRV.10		
Method	GET		
URIs	<ol style="list-style-type: none"> Active Services: /genesys/1/cs/customers/\${customer_id}/services/active Completed Services: /genesys/1/cs/customers/\${customer_id}/services/completed All Services: /genesys/1/cs/customers/\${customer_id}/services 		
Parameter	Type	Mandatory	Description
URI Parameters			
\${customer_id}	string	yes	The Customer ID, to get associated active or completed services. Note that a customer can be associated with a service either at the service creation or later in the process (see Associate Service).
started_from	date/time	no	Filters the services started on or after the given date/time in ISO 8601 representation: [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z
started_to	date/time	no	Filters the services started prior to the given date/time in ISO 8601 representation: [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z
completed_from	date/time	no	Filters the services that were completed on or after the given date/time in ISO 8601 representation: [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z.

ID	CV.WS.SRV.10		
			Only applicable when querying service history.
completed_to	date/time	no	Filters the services completed prior to the given date/time. Only applicable when querying service history.
service_types	long or string	no	Filters the services by the application-defined service type. Refer to Configuration Options for more details on Business Attribute mapping.
active_states	bool	no	<ul style="list-style-type: none"> "true" to include information on the service's active states in the results. "false" by default.
active_tasks	bool	no	<ul style="list-style-type: none"> "true" to include information on the service's active tasks in the results. "false" (default).
completed_states	bool	no	<ul style="list-style-type: none"> "true" to include information on the service's completed states in the results. "false" (default).
completed_tasks	bool	no	<ul style="list-style-type: none"> "true" to include information on the service's completed tasks in the results. "false" (default).
max_services Added in: 8.5.111	integer	no	Maximum number of services to return.
<extension key>	Any JSON type	no	Service attached data as key-value pairs. You can add as many key-

ID	CV.WS.SRV.10				value pairs as needed.
-----------	--------------	--	--	--	------------------------

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response	
HTTP code	200
HTTP message	OK
Body	Service[] .

Example

Operation

```
GET /genesys/1/cs/customers/ABC1234/services/active?active_states=true
```

Response

```
[ // returned in an array
  { "customer_id": "ABC1234",
    "service_id": 4692834,
    "est_duration": 86400,
    "started": {
      "timestamp": "2009-05-07T12:05:20.157",
      // additional Start Event fields
    },
    "active_states":
      [// included given specification of "results" attribute
        { // array of one or more State objects
          "state_id": 5005,
          "state_type": 8, // service delivery
          "started": {
            "timestamp": "2009-05-07T12:08:53.298",
            // additional Start Event fields
          }
        }
      ]
    }
  ]
}]
```

Important

In the above example, the inline State objects do not include active/completed tasks.

Query State by ID

GET /genesys/1/cs/services/\${service_id}/states/\${state_id}

Description

This operation allows you to retrieve a single state for a given service. By default, no task information is returned with the state representation. Optional URI parameters enable to include the state's nested tasks in the response.

Operation

ID	CV.WS.SRV.15		
Method	GET		
URL	/genesys/1/cs/services/\${service_id}/states/\${state_id}		
Field name	Type	Mandatory	Description
URI Parameters			
\${service_id}	integer	yes	The service ID.
\${state_id}	integer	yes	The state ID.
active_tasks	bool	no	<ul style="list-style-type: none"> <i>true</i> to include information on the service's active tasks in the results. <i>false</i> (default)
completed_tasks	bool	no	<ul style="list-style-type: none"> <i>true</i> to include information on the service's completed tasks in the results. <i>false</i> (default)
<extension key>	Any JSON type	no	Service attached data as key-value pairs. You can add as many key-value pairs as needed.

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response

HTTP code	200		
HTTP message	OK		
Field Name	Type	Mandatory	Description
Body: State<ref>This response includes the State representation with the state ID.</ref>			
state_id	integer	yes	The 32-bit integer ID of the state, assigned by the UCS when the state is started.
state_type	long or string	yes	<p>The unique ID associated with the state type. For instance, this ID can be the DB ID of a value in the Service Type Business Attribute.</p> <p>Enumerated values could be one of the following:</p> <ol style="list-style-type: none"> 1. Customer identification 2. Service identification 3. Assign service agent 4. Waiting for service agent 5. Offering another service while waiting for agent 6. Offering callback 7. Callback pending 8. Delivering service 9. Waiting for customer input 10. Offering another service while delivering service <p>Refer to Configuration Options for more details on <i>Business Attribute</i> mapping.</p>

HTTP code		200	
service_id	integer	yes if not nested in service resource	The service's unique ID if the state is not part of a top-level service resource which includes a service ID. <ref name="toplevel">Only top-level representations include this field. For instance, if your application retrieves a service which includes nested states and tasks, only the top-level service resource contains this ID.</ref>
started	State Start Event	yes	Related start event. This event should not contain the "est_duration" field, if it already exists at the state level.
completed	State End Event	no	The related end event if the state is completed. This event should not repeat the "disposition" and "disposition_desc" fields if they already exist at the state level.
customer_id	string	no	The customer's unique ID <ref name="toplevel"/>.
est_duration	integer	no	The estimated time for completing the state, in seconds.
duration	integer	no	The total duration in milliseconds. Important Only relevant for completed states.
disposition	long or string	no	ID of the Business Disposition ID. Important Only relevant for completed states.
disposition_desc	string	no	The reason for assigning the business disposition with the service element. Limited to 256

HTTP code	200		
			characters.
active_tasks	Task[]	no<ref name="option">This field is returned if the corresponding option is set to true in the URI parameters.</ref>	The array of tasks that are started but not yet completed.
completed_tasks	Task[]	no<ref name="option"/>	The array of Tasks that are completed.
<extension key>	Any JSON type	no	Service attached data as key-value pairs. You can add as many key-value pairs as needed.

<references />

Example

Operation

```
GET /genesys/1/cs/services/2/states/
1?active_tasks=true&deleted_tasks=true&extensions=Feedback,Satisfaction
```

Result

```
200 OK
[ // array of one or more state object
{
  "service_id" : 1,
  "state_id" : 0,
  "state_type" : 100,
  "est_duration" : 60,
  "started" : {
    "timestamp" : "2010-05-18T15:23:34.447Z",
    "application_type" : 400,
    "resource_id" : 20,
    "media_type" : 2,
    "resource_type" : 200,
    "application_id" : 40,
    "interaction_id" : "51"
  },
  "Feedback":
  {"FeedbackType":"survey",
  "rating":7,
  "notes":"warm welcome at frontdesk, thanks for the nice trip"}
},
  "Satisfaction": [
  {
    "rating":2,
    "pertinence":8,
    "usefull":true,
    "place":"Terranova mexico resort"
  },

```

```
{
  "rating":8,
  "pertinence":4,
  "usefull":false,
  "place":"Fancy resort Paris"
},
"active_tasks": [
  // included given specification of "results" attribute
  { // array of one or more Task objects
    "task_id": 25080,
    "task_type": 5, // application-defined task type
    "est_duration": 300,
    "started": {
      "timestamp": "2009-05-07T12:08:53.298",
    }
  }
]
"completed_tasks": [
  { "task_id": 24027,
    "task_type": 2, // such as task for processing quote
    "disposition": 10, // for example quote completed
    "disposition_desc": "<text describing quote result>",
    "started": {
      "timestamp": "2009-05-07T12:02:23.715",
      "interaction_id": "123ABC908ABFFD8080"
    },
    "completed": {
      "timestamp": "2009-05-07T12:06:23.715",
      "interaction_id": "157C9A208AFD523D01"
    }
  }
]
},
{
  "service_id" : 1,
  "state_id" : 1,
  "state_type" : 100,
  "est_duration" : 60,
  "started" : {
    "timestamp" : "2010-05-18T15:23:41.977Z",
    "application_type" : 400,
    "resource_id" : 20,
    "media_type" : 2,
    "resource_type" : 200,
    "application_id" : 40,
    "interaction_id" : "51"
  }
}
]
```

Query States

GET /genesys/1/cs/genesys/1/cs/services/\${service_id}/states

Description

This operation queries the states for a given service. By default, all the states are returned, but your application can filter the results to retrieve only active or completed states, based on the specified URI. Additional URI parameters enable to include task information in the response.

Operation

Query states

ID	CV.WS.SRV.12		
Method	GET		
URL	<ol style="list-style-type: none"> All (default): /genesys/1/cs/services/\${service_id}/states Active States: /genesys/1/cs/services/\${service_id}/states/active Completed States: /genesys/1/cs/services/\${service_id}/states/completed 		
Field Name	Type	Mandatory	Description
URI Parameters			
\${service_id}	string	yes	The service ID generated from the Start Service operation.
state_types	integer	no	Filters specific service state types. The possible enumerated values are available in the filter field of the State resources . You can specify several state_type values separated by commas.
active_tasks	bool	no	<ul style="list-style-type: none"> <i>true</i> to include information the service's active tasks in the results. <i>false</i> (default)

ID	CV.WS.SRV.12		
completed_tasks	bool	no	<ul style="list-style-type: none"> <i>true</i> to include information on the service's completed tasks in the results. <i>false</i> (default)
<extension>	Any JSON type	no	State attached data as key-value pairs. You can add as many key-value pairs as needed.

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response	
HTTP code	200
HTTP message	OK
Body	State[]

Example

Operation

```
GET /genesys/1/cs/services/ABC1235XZY/states/active~tive_tasks=trueCleted_tasks=true&extensions=Feedback
```

Result The following output shows a possible response for the previous request:

```
200 OK
[ // array of one or more state object
  {
    "service_id" : 1,
    "state_id" : 0,
    "state_type" : 100,
    "est_duration" : 60,
    "started" : {
      "timestamp" : "2010-05-18T15:23:34.447Z",
      "application_type" : 400,
      "resource_id" : 20,
      "media_type" : 2,
      "resource_type" : 200,
      "application_id" : 40,
      "interaction_id" : "51"
    }
    "Feedback":
```

```
{ "FeedbackType":"survey",
  "rating":7,
  "notes":"warm welcome at frontdesk, thanks for the nice trip"}
},
"active_tasks": [
  // included given specification of "results" attribute
  { // array of one or more Task objects
    "task_id": 25080,
    "task_type": 5, // application-defined task type
    "est_duration": 300,
    "started": {
      "timestamp": "2009-05-07T12:08:53.298",
    }
  }
]
"completed_tasks": [
  { "task_id": 24027,
    "task_type": 2, // such as task for processing quote
    "disposition": 10, // such as quote completed
    "disposition_desc": "<text describing quote result>",
    "started": {
      "timestamp": "2009-05-07T12:02:23.715",
      "interaction_id": "123ABC908ABFFD8080"
    },
    "completed": {
      "timestamp": "2009-05-07T12:06:23.715",
      "interaction_id": "157C9A208AFD523D01"
    }
  }
]
},
{
  "service_id" : 1,
  "state_id" : 1,
  "state_type" : 100,
  "est_duration" : 60,
  "started" : {
    "timestamp" : "2010-05-18T15:23:41.977Z",
  },
  "application_type" : 400,
  "resource_id" : 20,
  "media_type" : 2,
  "resource_type" : 200,
  "application_id" : 40,
  "interaction_id" : "51"
}
}
]
```

Query Task by ID

GET /genesys/1/cs/services/\${service_id}/tasks/\${task_id}

Description

Queries a task by its ID for the given service.

Operation

ID	CV.WS.SRV.16		
Method	GET		
URL	/genesys/1/cs/services/\${service_id}/tasks/\${task_id}		
Name	Type	Mandatory	Description
URI Parameters			
\${service_id}	integer	yes	The service ID.
\${task_id}	integer	yes	The task ID.
<extension key>	Any JSON type	no	Task attached data as key-value pairs. You can add as many key-value pairs as needed.

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response

HTTP code	200		
HTTP message	OK		
Field Name	Type	Mandatory	Description
Body: Task<ref>This response includes the Task representation associated with the task ID.</ref>			
state_id	integer	no	The 32-bit integer ID for the state service associated with the task.

HTTP code	200		
task_id	integer	yes	The 32-bit integer ID of the task, assigned by the UCS when the task is started.
task_type	long or string	yes	The unique ID associated with the type of service<ref name="business">Refer to Configuration Options for more details on <i>Business Attribute</i> mapping.</ref>. For instance, this ID can be the DB ID of a value in the Service Type Business Attribute.
service_id	integer	yes if not specified in top-level resources<ref name="toplevel">Only top-level representations include this field. For instance, if your application retrieves a service which includes nested states and tasks, only the top-level service representations contains this ID.</ref>	The service's unique ID if the task is not part of a top-level state or service resource which includes a service ID.
started	Task Start Event	yes	Related start event. This start event does not contain the "est_duration" field, if this field is already specified at the task level.
completed	Task End Event	no	The related end event if this task is completed. This end event does not include the "disposition" and "disposition_desc" fields if they are already specified at the task level.
customer_id	string	no	The customer's unique ID<ref name="toplevel"/>.
est_duration	integer	no	The estimated time for completing the item, in seconds.
duration	integer	no	The total duration in milliseconds.

HTTP code	200		
			Only relevant for completed tasks.
disposition	long or string	no	ID of the Business Disposition ID<ref name="business"/>. Only relevant for completed tasks.
disposition_desc	string	no	The reason for assigning the business disposition with the service element. Limited to 256 characters.
<extension key>	Any JSON type	no	Task attached data as key-value pairs. You can add as many key-value pairs as needed.

<references />

Example

Operation

GET /genesys/1/cs/services/942/tasks/12?extensions=Survey,Proposal

Result

```
{
  "interaction_id":42,
  "est_duration":460,
  "state_id":24,
  "task_type":55,
  "Survey":
  {
    "url":"http://ourServer/storage/userAnswers",
    "question1":7,
    "question2":true,
    "question3":"will be better with cable tv and on-demand video"
  },
  "Proposal": [
    {
      "car type":"cabriolet",
      "price":25 000,
      "seats":2,
      "comments":"200 cv, hardtop"
    },
    {
      "car type":"S.U.V.",
      "price":70 000,
      "seats":8,
      "comments":"4wd, leather seats"
    }
  ]
}
```

}

Query Tasks

GET /services/\${service_id}/tasks

Description

This operation queries the tasks for a given service. By default, all the tasks are returned, but your application can filter the results to retrieve only active or completed tasks, based on the specified URI. Your application can also filter the returned tasks with additional URI parameters to select one or more task type and a given state.

Operation

ID	CV.WS.SRV.13		
Method	GET		
URL	<ol style="list-style-type: none"> All (default): /genesys/1/cs/services/\${service_id}/tasks Active Tasks: /genesys/1/cs/services/\${service_id}/tasks/active Completed Tasks: /genesys/1/cs/services/\${service_id}/tasks/completed 		
Field Names	Type	Mandatory	Description
URI Parameters			
\${service_id}	string	yes	The service ID generated from the service creation .
state_id	integer	no	Used to filter tasks by state ID. To filter both on the state id and the task_type, your application can also use this field identically to task_type. For example: /genesys/1/cs/services/\${service_id}/tasks/completed?task_types=1,3&state_id=700
task_types	long or string	no	Filters specific service task types separated by commas.
extensions	string	no	Names, separated by commas, of the task extension keys to return

ID	CV.WS.SRV.13		with the matching task resources. By default, no extension is returned.

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response	
HTTP code	200
HTTP message	OK
Body	Task[.].

Example

Retrieve completed tasks

Operation Retrieve completed tasks for service 6001:

```
GET /genesys/1/cs/services/15/tasks/completed?task_types=STT-DefineCustomerNeed%2CSTT-FillCustomerInfo&state_id=5&extensions=Survey
```

Result

```
[ {
  "service_id" : 1,
  "state_id" : 2,
  "task_id" : 0,
  "task_type" : 55,
  "est_duration" : 55,
  "started" : {
    "timestamp" : "2010-05-20T15:41:52.380Z",
    "interaction_id" : "65"
  },
  "Survey":
  { "url": "http://ourServer/storage/userAnswers",
    "question1": 7, "question2": true,
    "question3": "will be better with cable tv and on-demand video"
  },
  "Proposal": [
    {
      "car type": "cabriolet",
      "price": 25 000,
      "seats": 2,
      "comments": "200 cv, hardtop"
    }
  ]
}
```

```

    },
    {
      "car type": "S.U.V.",
      "price": 70 000,
      "seats": 8,
      "comments": "4wd, leather seats"
    }
  ]
}, {
  "service_id" : 1,
  "state_id" : 1,
  "task_id" : 1,
  "task_type" : 12,
  "est_duration" : 900,
  "started" : {
    "timestamp" : "2010-05-20T15:42:57.207Z",
    "interaction_id" : "954"
  }
},
} ]

```

Retrieve active tasks

Operation Retrieve active tasks for service 6001:

```
GET /genesys/1/cs/services/6001/tasks/active
```

Result

```

[ {
  "service_id" : 6001,
  "task_id" : 15001,
  "task_type" : 1433,
  "started" : {
    "timestamp" : "2010-04-25T16:07:12.133Z",
    "application_type" : 1426,
    "media_type" : 1010,
    "resource_type" : 1423
  }
} ]

```

Operation Retrieve all tasks for service 6001:

```
GET /genesys/1/cs/services/6001/tasks
```

Result

```

[ {
  "service_id" : 6001,
  "duration" : 0,
  "task_id" : 15000,
  "task_type" : 1433,
  "started" : {
    "timestamp" : "2010-04-25T16:07:12.133Z",
    "application_type" : 1426,
    "media_type" : 1010,
    "resource_type" : 1423
  },
  "completed" : {
    "timestamp" : "2010-04-25T16:07:12.133Z",
    "application_type" : 1426,
  }
} ]

```

```
    "media_type" : 1010,
    "resource_type" : 1423
  }
}, {
  "service_id" : 6001,
  "task_id" : 15001,
  "task_type" : 1433,
  "started" : {
    "timestamp" : "2010-04-25T16:07:12.133Z",
    "application_type" : 1426,
    "media_type" : 1010,
    "resource_type" : 1423
  }
} ]
```

Start Service

POST /genesys/1/cs/services/start

Description

Starts a service based on:

- The service type passed in the body.
- The contact information available (customer ID or contact key for later lookup).

If your application does not assign a customer ID, your service is anonymous. In that case, your application must assign a contact key. For further information, read the following pages:

- [Anonymous Service](#)
- [Service Start Event](#)

Important

Your application is not responsible for the service ID creation. Context Services will automatically assign 64-bit integer IDs to the created services.

Attached Data

You can attach data to the service by adding key-value pairs to the JSON body of the POST action. Key-value pairs replace 8.1 extensions, do not need schema definition like in 8.1, and ensure backward compatibility with 8.1.

Operation

Start Service

ID	CV.WS.SRV.1		
Method	POST		
URL	/genesys/1/cs/services/start		
Field Name	Type	Mandatory	Description
Body: Service Start Event. The body of this operation is mandatory and makes use of the Service Start Event .			

ID	CV.WS.SRV.1		
customer_id	string	yes if no <i>contact_key</i>	<p>The unique ID of the customer associated with the given service.</p> <p>Important Services with no <i>customer_id</i> are anonymous. In this case, the event should include a <i>contact_key</i>.</p>
contact_key	string	yes if no <i>customer_id</i>	<p>Key for later lookup of the service. This <i>contact_key</i> is mandatory if no customer ID is specified. For example, the application might store a PIN or ANI as the contact key when the service is first started. Later, if the customer is identified then the contact key is used to lookup the existing service record and associate it with the customer.</p>
service_type	long or string	no	<p>The unique ID associated with the type of service, typically the DB ID of a value in the Service Type Business Attribute. Refer to Configuration Options for more details on Business Attribute mapping.</p>
session_id	string	no	<p>The ID of the related session, for instance, the orchestration session, or any other business session.</p>
interaction_id	string	no	<p>The ID of the related Genesys interaction. This ID can be used by other Genesys reporting products such as Stat Server, URS, Composer, and GVP.</p>
application_type	long or string	no	<p>The unique ID associated with the type or class application issuing the service</p>

ID	CV.WS.SRV.1		
			event. May be used to group related applications, potentially across resource types. Refer to Configuration Options for more details on Business Attribute mapping.
application_id	integer	no	The unique ID (Genesys DB ID) for the application issuing the service event, such as a GVP VoiceXML application or an Orchestration SCXML application.
resource_type	long or string	no	The unique ID associated with the type or class of resource which provides the service (such as GVP, Agent Desktop, or Orchestration). Refer to Configuration Options for more details on Business Attribute mapping.
resource_id	integer	no	<p>The unique DB ID for the specific resource which provides the service. For instance:</p> <ul style="list-style-type: none"> the Genesys DB ID of a specific GVP or orchestration platform. the DB ID of a given agent, according to the context.
media_type	long or string	no	The media type applicable to the given service, for instance, email, voice, chat, etc. Refer to Configuration Options for more details on Business Attribute mapping.
est_duration	integer	no	The estimated service duration in seconds.
timestamp	date/time	no	The UTC time at which the event was raised,

ID	CV.WS.SRV.1		
			with a precision of milliseconds, using the ISO 8601 representation: [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z. If the application does not specify this timestamp, the server does it when the event is processed.
auto_closed	boolean	no	If true, creates a completed service; false, by default.
<extension key>	Any JSON type	no	Service attached data as key-value pairs. You can add as many key-value pairs as needed.

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response

HTTP code	201
HTTP message	Created
Header	Location: /genesys/1/cs/services/\${service_id} where: <ul style="list-style-type: none"> • \${service_id} is the service ID.
Body	{"service_id": \${service_id}} where: <ul style="list-style-type: none"> • \${service_id} is the service ID.

Example

Operation

```
POST http://localhost:8080/genesys/1/cs/services/start
{
```

```
"timestamp": "2009-05-12T12:05:12.145Z",
"interaction_id": "123ABCAADFJ1259ACF",
"application_type":400,
"application_id":40,
"est_duration":60,
"contact_key": "42",
"service_type":100,
"media_type": 1,
"resource_id": 5005,
"resource_type": 2,
"disposition": 10,
"coupon": {
  "coupon_name": "DISCOUNTCODE15"
},
"satisfaction": {
  "score": 85,
  "agentID": 2025
},
"relatedOffers": [
  {
    "offer_name":"VIP credit card black ed.",
    "type":9,
    "comments":"proposed to all client"
  },
  {
    "offer_name":"3 times payment GOLD",
    "type":4,
    "comments":"limited offer"
  },
  {
    "offer_name":"life insurance",
    "type":3,
    "comments":"health check to be done before approval"
  }
]
}
```

Result

```
HTTP 201 Created
location: /genesys/1/cs/services/12349857
{"service_id": 12349857}
```

Start State

POST /genesys/1/cs/services/\${service_id}/states/start

Description

Starts a service state of a given type for a given service. UCS assigns auto-incrementing identifiers to the states, just like to services and tasks. States and tasks are assigned 32-bit identifiers.

Operation

ID	CV.WS.SRV.5		
Method	POST		
URL	/genesys/1/cs/services/\${service_id}/states/start		
Name	Type	Mandatory	Description
URI Parameters			
\${service_id}	integer	yes	The service ID is specified in the request URI.
Body: State Start Event. This body contains fields from the State Start Event resource.			
state_type	long or string	no	<p>The unique ID associated with the state type. Typically, the DB ID of a value in the Service Type Business Attribute.</p> <p>Refer to Configuration Options for more details on Business Attribute mapping. Enumerated values could be one of the following:</p> <ol style="list-style-type: none"> 1. Customer identification 2. Service identification 3. Assign service agent 4. Waiting for service agent 5. Offering another service while waiting

ID	CV.WS.SRV.5		
			for agent 6. Offering callback 7. Callback pending 8. Delivering service 9. Waiting for customer input 10. Offering another service while delivering service
previous_state_id	integer	no	The ID of the previous state.
session_id	string	no	The ID of the related session; for instance, the orchestration session or any other business session.
interaction_id	string	no	The ID of the related Genesys interaction. This ID can be used by other Genesys reporting products such as Stat Server, URS, Composer, and GVP.
application_type	long or string	no	The unique ID associated with the type or class of application issuing the service event. May be used to group related applications, potentially across resource types.
application_id	integer	no	The unique ID (Genesys DB ID) for the application issuing the service event, such as a GVP VoiceXML application or an Orchestration SCXML application.
resource_type	long or string	no	The unique ID associated with the type or class of resource which provides the service (such as GVP, Agent Desktop, or Orchestration).
resource_id	integer	no	The unique DB ID for

ID	CV.WS.SRV.5		
			<p>the specific resource which provides the service. For instance:</p> <ul style="list-style-type: none"> the Genesys DB ID of a specific GVP or orchestration platform. the DB ID of a given agent, according to the context.
media_type	long or string	no	The media type applicable to the given state; for instance, email, voice, or chat.
est_duration	integer	no	The estimated state duration in seconds.
timestamp	date/time	no	<p>The UTC time at which the event was raised, with a precision of milliseconds, using the ISO 8601 representation: [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z. If the application does not specify this timestamp, the server does it when the event is processed.</p>
<extension key>	Any JSON type	no	State attached data as key-value pairs. You can add as many key-value pairs as needed.

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response	
HTTP code	201
HTTP message	Created
Header	Location: /genesys/1/cs/services/ {service_id} /states/ {state_id}

HTTP code	201
	<p>where:</p> <ul style="list-style-type: none"> • <code>\${service_id}</code> is the service ID. • <code>\${state_id}</code> is the ID of the created state.
Body	<pre>{ "state_id": \${state_id} }</pre> <p>where:</p> <ul style="list-style-type: none"> • <code>\${state_id}</code> is the stateID.

Example

Operation

POST /genesys/1/cs/services/12345/states/start

```
{
  "interaction_id": "51",
  "application_type": 400,
  "application_id": 40,
  "resource_type": 200,
  "resource_id": 20,
  "media_type": 2,
  "est_duration": 60,
  "state_type": 100,
  "Feedback": {
    "FeedbackType": "survey",
    "rating": 7,
    "notes": "warm welcome at frontdesk, thanks for the nice trip"
  },
  "Satisfaction": [
    {
      "rating": 2,
      "pertinence": 8,
      "usefull": true,
      "place": "Terranova mexico resort"
    },
    {
      "rating": 8,
      "pertinence": 4,
      "usefull": false,
      "place": "Fancy resort Paris"
    }
  ]
}
```

Result

```
{ "state_id": 1258276 }
```


Start Task

POST /genesys/1/cs/services/\${service_id}/tasks/start

Description

This operation starts a task of a given type for the given service. The type is one of the enumerated values recorded for the corresponding business attribute. See [Business Attributes in Context Services](#) for further details. UCS assigns auto-incrementing identifiers to states, services, and tasks. For states and tasks, the assigned identifiers are 32-bit integers; for services the identifier is a 64-bit integer.

Operation

ID	CV.WS.SRV.8		
Method	POST		
URL	/genesys/1/cs/services/\${service_id}/tasks/start		
Field Name	Type	Mandatory	Description
URI Parameters			
\${service_id}	integer	yes	The unique 64-bit ID of the service.
Body: Task Start Event This body contains fields from the Task Start Event resources.			
state_id	integer	no	The 32-bit integer ID of the state. See State .
task_type	long or string	no	The unique ID associated with the type of task, typically the DB ID of a value in a Business Attribute representing customer service tasks for the given organization Refer to Configuration Options for more details on Business Attribute mapping.</ref>.
session_id	string	no	The ID of the related session, for instance, the orchestration session or any other business session.
interaction_id	string	no	The ID of the related

ID	CV.WS.SRV.8		
			Genesys interaction. This ID can be used by other Genesys reporting products such as Stat Server, URS, Composer, and GVP.
application_type	long or string	no	The unique ID associated with the type or class of application issuing the service event. May be used to group related applications, potentially across resource types.
application_id	integer	no	The unique ID (Genesys DB ID) for the application issuing the event, such as a GVP VoiceXML application or an Orchestration SCXML application.
resource_type	long or string	no	The unique ID associated with the type or class of resource which provides the service (such as GVP, Agent Desktop, or Orchestration).
resource_id	integer	no	The unique DB ID for the specific resource which provides the service. For instance: <ul style="list-style-type: none"> the Genesys DB ID of a specific GVP or orchestration platform. the DB ID of a given agent, according to the context.
media_type	long or string	no	The media type applicable to the given task; for instance, email, voice, or chat.
est_duration	integer	no	The estimated task duration in seconds.
timestamp	date/time	no	The UTC time at which the event was raised, with a precision of milliseconds, using the ISO 8601

ID	CV.WS.SRV.8		
			representation : [YYYY] - [MM] - [DD]T[HH] : [mm] : [ss] . [SSS]Z. If the application does not specify this timestamp, the server does it when the event is processed.
<extension key>	Any JSON type	no	Service attached data as key-value pairs. You can add as many key-value pairs as needed.

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response

HTTP code	201
HTTP message	Created
Header	Location: <code>\${base_uri}/service/\${service_id}/tasks/\${task_id}</code> where: <ul style="list-style-type: none"> <code>\${base_uri}</code> is the URI of the created service. <code>\${service_id}</code> is the service ID. <code>\${task_id}</code> is the created ID for the new task.
Body	<code>{"task_id": <code>\${task_id}</code>}</code> where: <ul style="list-style-type: none"> <code>\${task_id}</code> is the task ID.

Example

Operation

```
POST /genesys/1/cs/services/21456878/tasks/start
{
  "interaction_id":42,
  "est_duration":460,
```

```
"state_id":24,
"task_type":customer info,
"Survey":
{
  "url":"http://ourServer/storage/userAnswers",
  "question1":7,
  "question2":true,
  "question3":"will be better with cable tv and on-demand video"
},
"Proposal": [
  {
    "car type":"cabriolet",
    "price":25 000,
    "seats":2,
    "comments":"200 cv, hardtop"
  },
  {
    "car type":"S.U.V.",
    "price":70 000,
    "seats":8,
    "comments":"4wd, leather seats"
  }
]
}
```

Result

```
{"task_id": 15928}
```

Update Expiration Time

PUT /genesys/1/cs/services/{service_id}

Description

You can make a conversation expire by adding a Time-to-live (TTL) or expiration time to a Service by updating your resource with this query. When you update the TTL or service expiration time, you overwrite previous updates and any expiration time **set in your configuration**. You can also use `ttl` and `expiration_time` attributes in **Service Start Events** and **Service End Events**.

Important

Updating the conversation expiration time has a performance impact and uses as much CPU as three event queries.

Operation

Important

You need a WRITE role to use this request.

Method	PUT		
URL	/genesys/1/cs/services/{service_id}		
Parameter	Type	Mandatory	Description
URI Parameters			
{service_id}	string	yes	The id of the service to request.
expiration_time	string	no	Time at which the service should expire, using the ISO 8601 representation: [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z
ttl	integer	no	Time-to-live of the full conversation in seconds from this event time.

Method	PUT		
			This parameter takes precedence over <code>expiration_time</code> parameter which is discarded if you pass both parameters.

Response

The following table shows the correct response for a successful request. In case of error, you will see standard [HTTP response error codes](#).

HTTP code	200
HTTP message	OK
Body	Empty

Example

```
PUT http://localhost:8010/genesys/1/cs/services/115-cc038ec2-c284-41e5-8560-b006743e1935
{
  "ttl": 86400
}
```

Response: 200 OK

Update Service Extension

PUT /services/{service_id}/extensions/{ext_name}

Description

This operation replaces the extension value with a new extension value. The former value of the extension is lost.

Important

In 8.5, extensions are JSON key-value pairs. This ensures backward compatibility and simplifies the management of extensions.

This operation supports the update of multi-valued extensions.

Operation

Updates a service's extension value

ID	CV.WS.SRV.4		
Method	PUT		
URL	/services/{service_id}/extensions/{ext_name}		
Name	Type	Mandatory	Description
URI Parameters			
{service_id}	integer	yes	The ID of the service.
{ext_name}	string	yes	The name of the extension
Body			
<extension> or <extension>[]	Any JSON type	yes	Attached data as key-value pairs or array of key-value pairs.

Response

The Context Management Service API answers with HTTP codes for every request. The following table

shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response	
HTTP code	200
HTTP message	OK

Example

Operation

The following operation updates the single-valued extension "score".

```
PUT /services/8389/extensions/score
{
  "score": 85,
  "agentID": 2025
}
```

Operation

The following operation updates the multi-valued extension "relatedOffers".

```
PUT /services/8389/extensions/relatedOffers
[
  {
    "offer_name": "VIP credit card black ed.",
    "type": 9,
    "comments": "proposed to all client"
  },
  {
    "offer_name": "3 times payment GOLD",
    "type": 4,
    "comments": "limited offer"
  },
  {
    "offer_name": "life insurance",
    "type": 3,
    "comments": "health check to be done before approval"
  }
]
```

Result

```
200 OK
```


Update State Extension

PUT /genesys/1/cs/services/\${service_id}/states/\${state_id}/extensions/\${ext_name}

Description

This operation replaces the extension value with a new extension value. The former value of the extension is lost.

Operation

Updates a state's extension value

ID	CV.WS.SRV.17		
Method	PUT		
URL	/genesys/1/cs/services/\${service_id}/states/\${state_id}/extensions/\${ext_name}		
Name	Type	Mandatory	Description
URI Parameters			
\${service_id}	integer	yes	The ID of the service.
\${state_id}	integer	yes	The ID of the state.
\${ext_name}	string	yes	The name of the extension.
Body			
<extension> or <extension>[\	value	yes	Values for one or more extensions.

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response

HTTP code	200
HTTP message	OK

Example

Operation: Update a multi-valued extension

```
PUT /genesys/1/cs/services/8389/states/1/extensions/Satisfaction
[
  {
    "rating":2,
    "pertinence":8,
    "usefull":true,
    "place":"Terranova mexico resort"
  },
  {
    "rating":8,
    "pertinence":4,
    "usefull":false,
    "place":"Fancy resort Paris"
  }
]
```

Result

204 No Content

Operation: Update a single-valued extension

```
PUT /genesys/1/cs/services/8389/states/1/extensions/Feedback
{
  "FeedbackType":"survey",
  "rating":7,
  "notes":"warm welcome at frontdesk, thanks for the nice trip"
}
```

Result

200 OK

Update Task Extension

POST /genesys/1/cs/
services/{service_id}/tasks/{task_id}/extensions/{extension_name}

Description

This operation replaces the extension value with a new extension value. The former value of the extension is lost.

Operation

Updates a state's extension value

ID	CV.WS.SRV.18		
Method	PUT		
URL	/genesys/1/cs/ services/{service_id}/tasks/{task_id}/extensions/{extension_name}		
Name	Type	Mandatory	Description
URI Parameters			
{service_id}	integer	yes	The ID of the service.
{task_id}	integer	yes	The ID of the task.
{extension_name}	string	yes	The name of the extension
Body			
<extension n>	Any JSON type	yes	Values for one or more extension attached data.

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response

HTTP code	200
HTTP message	OK

Example

Operation

Update a multi-valued extension.

```
PUT /genesys/1/cs/services/8389/tasks/42/extensions/Proposal
[
  {
    "car type":"cabriolet",
    "price":25 000,
    "seats":2,
    "comments":"200 cv, hardtop"
  },
  {
    "car type":"S.U.V.",
    "price":70 000,
    "seats":8,
    "comments":"4wd, leather seats"
  }
]
```

Result

200 OK

Operation

Update a single-valued extension.

```
PUT /services/8389/tasks/42/extensions/Survey
{
  "url":"http://ourServer/storage/userAnswers",
  "question1":7,
  "question2":true,
  "question3":"will be better with cable tv and on-demand video"
}
```

Result

200 OK

Service Resources

Groups the Service resources.

- **End Event List:** List of the end events that your application can receive if a service, state, or task is terminated.
- **Extension:** JSON extension for the service API.
- **State Transition Event:** Your application can receive this event if a service performs a transition between two states.
- **Service:** The service resource, which is a container for nested states and tasks.
- **State:** The state resource.
- **Start Event List:** List of the start events that your application can receive if a service, state, or task is started.
- **Task:** The task resource.

End Event List

Purpose: Lists the End Event resources used to describe the end of a service, state, or task.

Service End Event

Field	Type	Mandatory	Description
service_id	integer	yes	The unique 64-bit ID of the given service.
session_id	string	no	The ID of the related session (for instance, the orchestration session or any other business session). Limited to 32 characters.
interaction_id	string	no	The ID of the interaction at the service event's creation. Limited to 50 characters.
application_type	long or string	no	The unique ID associated with the type or class of application that issued the service event. Refer to Configuration Options for more details on <i>Business Attribute</i> mapping. May be used to group related applications, potentially across resource types.
application_id	integer	no	The Genesys DB ID for the application which issues the service event, such as a GVP VoiceXML application, Orchestration SCXML application, and so on.
resource_type	long or string	no	The unique ID associated with the type or class of resource providing the service (for example, GVP, Agent Desktop, Orchestration).
resource_id	integer	no	The unique DB ID for

Field	Type	Mandatory	Description
			<p>the specific resource which provides the service. For instance:</p> <ul style="list-style-type: none"> The Genesys DB ID of a specific GVP or orchestration platform The DB ID of a given agent, according to the context
media_type	long or string	no	The media type applicable to the given service, such as email, voice, chat, and so on.
disposition	long or string	no	The unique ID for the business disposition assigned to the given service. For instance, this can be the DB ID of a Business Attribute used to enumerate a given organization's business dispositions.
disposition_desc	string	no	A text description that provides additional information about the business disposition. Limited to 64 characters.
timestamp	date/time	no	<p>The UTC time at which the service event was raised, with a precision of milliseconds, using the ISO 8601 representation : [YYYY] - [MM] - [DD]T[HH] : [mm] : [ss] . [SSS]Z. If the application does not specify this timestamp, the server adds it when the service event is processed.</p>
expiration_time	string	no	<p>Time at which the service should expire, using the ISO 8601 representation: [YYYY] - [MM] - [DD]T[HH] : [mm] : [ss] . [SSS]Z. For example : "2016-09-02T08:00:00.000Z"</p>

Field	Type	Mandatory	Description
			<p>Important</p> <p>This expiration time applies to the whole conversation (service, states, tasks), not only to the given service.</p>
ttl	integer	no	<p>Time to live (TTL) in seconds of the full conversation (service, states, tasks) starting from this event time. If the query includes both the ttl and expiration_time parameters, GMS takes into account only the ttl value and ignores the expiration_time.</p> <p>Important</p> <p>This TTL applies to the whole conversation (service, states, tasks), not only to the given service.</p>
<extension>	Any JSON type	no	Task attached data as key-value pairs. You can add as many key-value pairs as needed.

Related Operations

- [Complete Service](#)

State End Event

Field	Type	Mandatory	Description
service_id	integer	yes	The unique 64-bit ID of the given service.
state_id	integer	yes	The 32-bit ID of the given state.
session_id	string	no	The ID of the related session (for instance, the orchestration session or any other business session). Limited to 32 characters.

Field	Type	Mandatory	Description
interaction_id	string	no	The ID of the interaction at the service event's creation. Limited to 50 characters.
application_type	long or string	no	The unique ID associated with the type or class of application which issued the state event. Refer to Configuration Options for more details on <i>Business Attribute</i> mapping. May be used to group related applications, potentially across resource types.
application_id	integer	no	The Genesys DB ID for the application which issues the state event, such as a GVP VoiceXML application, Orchestration SCXML application, and so on.
resource_type	long or string	no	The unique ID associated with the type or class of resource providing the related service (for example, GVP, Agent Desktop, Orchestration).
resource_id	integer	no	The unique DB ID for the specific resource which provides the related service. For instance: <ul style="list-style-type: none"> The Genesys DB ID of a specific GVP or orchestration platform The DB ID of a given agent, according to the context
media_type	long or string	no	The media type applicable to the given state, such as email, voice, chat, and so on.
disposition	long or string	no	The unique ID for the business disposition assigned to the given state. For instance, this

Field	Type	Mandatory	Description
			can be the DB ID of a Business Attribute used to enumerate a given organization's business dispositions.
disposition_desc	string	no	A text description that provides additional information about the business disposition. Limited to 64 characters.
timestamp	date/time	no	The UTC time at which the service event was raised, with a precision of milliseconds, using the ISO 8601 representation : [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z. If the application does not specify this timestamp, the server adds it when the service event is processed.
<extension>	Any JSON type	no	Task attached data as key-value pairs. You can add as many key-value pairs as needed.

Related Operations

- [Complete State](#)

Task End Event

Field	Type	Mandatory	Description
service_id	integer	yes	The unique 64-bit ID of the given service.
state_id	integer	no	The ID of the related state. Important If the task is optional, you have no obligation to associate it with a state. For instance, you can associate this task with a higher service level.

Field	Type	Mandatory	Description
task_id	integer	yes	The ID of the given task.
session_id	string	no	The ID of the related session (for instance, the orchestration session or any other business session). Limited to 32 characters.
interaction_id	string	no	The ID of the interaction at the service event's creation. Limited to 50 characters.
application_type	long or string	no	The unique ID associated with the type or class of application which issued the service event. Refer to Configuration Options for more details on <i>Business Attribute</i> mapping. May be used to group related applications, potentially across resource types.
application_id	integer	no	The Genesys DB ID for the application which issues the service event, such as a GVP VoiceXML application, Orchestration SCXML application, and so on.
resource_type	long or string	no	The unique ID associated with the type or class of resource providing the service (for example, GVP, Agent Desktop, Orchestration).
resource_id	integer	no	The unique DB ID for the specific resource which provides the service. For instance: <ul style="list-style-type: none"> The Genesys DB ID of a specific GVP or orchestration platform The DB ID of a given agent, according to the context
media_type	long or string	no	The media type

Field	Type	Mandatory	Description
			applicable to the given task, such as email, voice, chat, and so on.
disposition	long or string	no	The unique ID for the business disposition assigned to the given task. For instance, this can be the DB ID of a Business Attribute used to enumerate a given organization's business dispositions.
disposition_desc	string	no	A text description that provides additional information about the business disposition. Limited to 64 characters.
timestamp	date/time	no	The UTC time at which the service event was raised, with a precision of milliseconds, using the ISO 8601 representation : [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z. If the application does not specify this timestamp, the server adds it when the service event is processed.
<extension>	Any JSON type	no	Task attached data as key-value pairs. You can add as many key-value pairs as needed.

Related Operations

- [Complete Task](#)

Extension

Purpose: Describes an extension.

Description

An extension is an attached data, available as a key-value pair where the key is a string and the value a JSON type (object, array, numbers, etc.) You can add, remove, and update extensions added to services, states, and tasks. In 8.5 and further releases, there is no schema definition for extensions. Your application can create and add extension records when dealing with standard services, states, and task operations. For instance, if **you start a task**, you can specify extensions in your operation body as shown in the following query:

```
POST /genesys/1/cs/services/21456878/tasks/start
{
  "interaction_id":42,
  "est_duration":460,
  "state_id":24,
  "task_type":customer info,
  "Survey":
  {
    "url":"http://ourServer/storage/userAnswers",
    "question1":7,
    "question2":true,
    "question3":"will be better with cable tv and on-demand video"
  },
  "Proposal": [
    {
      "car type":"cabriolet",
      "price":25 000,
      "seats":2,
      "comments":"200 cv, hardtop"
    },
    {
      "car type":"S.U.V.",
      "price":70 000,
      "seats":8,
      "comments":"4wd, leather seats"
    }
  ]
}
```

There is no specific field for service extensions. In the API reference, you can see that extensions are available when the <extension key> parameter is present, as follows:

Service/State/Task Extension

Field	Type	Mandatory	Description
<extension key>	Any JSON type	no	Attached data as key-value pairs. You can use as many key-value pairs as needed.

<references />

State Transition Event

Purpose: State Transition event used to describe the transition of a state.

Description

Combines state event resources to describe the completion of a given state and the start of a new state in a single request, rather than issuing separate start and end requests.

Resource

Important

When your application deals with this resource, parts of the *from* and *to* fields are shared. See below.

State Transition Event

Field	Type	Mandatory	Description
service_id	integer	yes	The unique ID of the related service.
from	hash	yes	<p>The hash of the following fields describing the state from which the service is transitioning:</p> <ul style="list-style-type: none"> <i>state_id</i> (<i>mandatory</i>): The 32-bit ID of the state to complete. <i>disposition</i> (<i>optional</i>): The unique ID for the business disposition assigned to the given state. For instance, this can be the DB ID of a Business Attribute used to enumerate a

Field	Type	Mandatory	Description
			<p>given organization's business dispositions.</p> <p>Refer to Configuration Options for more details on Business Attribute mapping.</p> <ul style="list-style-type: none"> disposition_desc (optional): The text description which provides additional context on the business disposition. Limited to 64 characters.
to	hash	yes	<p>The hash of the following fields, which describe the state into which the service is transitioning:</p> <ul style="list-style-type: none"> state_type (mandatory): The unique ID associated with the type of service, typically the DB ID of a value in the Service Type Business Attribute. est_duration (optional): The estimated service duration in seconds.
The following fields apply both to the state to complete and the state to start:			
session_id	string	no	The ID of the related session, for instance, the orchestration session, or any other business session.
interaction_id	string	no	The ID of the related Genesys interaction. This ID can be used by other Genesys reporting products such as Stat Server, URS, Composer, and GVP.
application_type	long or string	no	The unique ID associated with the type or class of application issuing the service

Field	Type	Mandatory	Description
			event. May be used to group related applications, potentially across resource types.
application_id	integer	no	The unique ID (such as Genesys DB ID) for the application issuing the service event, such as a GVP VoiceXML application or an Orchestration SCXML application.
resource_type	long or string	no	The unique ID associated with the type or class<ref name="business"/> of resource which provides the service (for example, GVP, Agent Desktop, or Orchestration).
resource_id	integer	no	The unique DB ID for the specific resource which provides the service. For instance: <ul style="list-style-type: none"> the Genesys DB ID of a specific GVP or orchestration platform. the DB ID of a given agent, according to the context.
media_type	long or string	no	The media type applicable to the given service/state/task, for instance, email, voice, chat, etc.
timestamp	date/time	no	The UTC time at which the event was raised, with a precision of milliseconds, using the ISO 8601 representation : [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z. If the application does not specify this timestamp, the server does it when the event is processed.

Field	Type	Mandatory	Description
<extension>	Any JSON type	no	State attached data as key-value pairs. You can add as many key-value pairs as needed.

Related Operations

- [Perform State Transition](#)

Service

Purpose: Describes the Service resource.

Description

Important

Provides a consolidated view of the service information.

Services are customer commitments defined by the business application (IVR, Orchestration, Agent, etc.) which interacts with the customer. Each service potentially spans multiple interactions over a variety of media channels. Each service can be divided into **States** which include a list of **Tasks**. The service is completed once the states and their tasks are completed. Your application is responsible for state transitions and task management. The service resources (creation, start, or end) are managed through **Service Operations**, which update the service information through service events (**Start Event** and **End Events**).

Resource

This table lists the resource fields that your application can deal with through operations. According to the operation, your application may not use the totality of the resource.

Field	Type	Mandatory	Description
service_type	long or string	yes	The unique ID associated with the type of service, typically, the DB ID of a value in the Service Type Business Attribute. Refer to Configuration Options for more details about <i>Business Attribute</i> mapping.
service_id	integer	yes	The service's unique ID.<ref name="toplevel">Only top-level objects include this field. For instance, if your application retrieves a service which includes nested

Field	Type	Mandatory	Description
			states and tasks, only the top-level service object contains this ID.</ref>
started	Service Start Event	yes	Start event (see Start Service). This event should not include the "est_duration" field, if it already exists at the service level.
completed	Service End Event	no	The related end event if the service is completed (see Complete Service). Important This field should not include the "disposition" and "disposition_desc" fields if they already exist at the service level.
customer_id	string	no	The customer's unique ID<ref name="oplevel"/>. If the service has no customer ID, the service is Anonymous Service .
est_duration	integer	no	The estimated time for completing the service, in seconds.
duration	integer	no	The total duration in milliseconds. Only relevant for completed services.
disposition	long or string	no	ID of the Business disposition ID. Only relevant for completed services.
disposition_desc	string	no	The reason for assigning the business disposition with the service element. Limited to 256 characters.
active_states	State[]	no	The array of the started states which are not completed.
completed_states	State[]	no	The array of completed states through which the service has made transitions.
active_tasks	Task[]	no	The array of the started

Field	Type	Mandatory	Description
			tasks which are not complete.
completed_tasks	Task[]	no	The array of the tasks which are complete as part of the service delivery.
interactions	Interaction[]	no	The array of the interactions which are associated with this service.
<extension>	Any JSON type	no	Task attached data as key-value pairs. You can add as many key-value pairs as needed.

<references />

Related Operations

- [Query Services](#)
- [Query Service by ID](#)

State

Purpose: Describes a State Resource.

Description

Describes one of the service states.

Throughout their lifecycle, services go through states such as:

- customer identification
- agent assignment
- service delivery

The exact sequence of states depends on the steps which your customer service application implements (for example, IVR, orchestration, or agent application). Your application is responsible for starting and completing states, but also for managing state transitions. Each State can also contain nested **Task** resources (zero or more) that your application should manage as well.

Resource

This table lists the resource fields that your application can deal with through operations. According to the operation, your application may not use the totality of the resource.

Field	Type	Mandatory	Description
state_id	integer	yes	The 32-bit integer ID of the state, assigned by the UCS when the state is started.
state_type	long or string	yes	The unique ID associated with the state type. For instance, this ID can be the DB ID of a value in the Service Type Business Attribute. Enumerated values could be one of the following: <ol style="list-style-type: none"> 1. Customer identification 2. Service identification

Field	Type	Mandatory	Description
			3. Assign service agent 4. Waiting for service agent 5. Offering another service while waiting for agent 6. Offering callback 7. Callback pending 8. Delivering service 9. Waiting for customer input 10. Offering another service while delivering service Refer to Configuration Options for more details on <i>Business Attribute</i> mapping.
service_id	integer	yes/no	The service's unique ID if the state is not part of a top-level service resource which includes a service ID. <ref name="toplevel"> Only top-level representations include this field. For instance, if your application retrieves a service which includes nested states and tasks, only the top-level service resource contains this ID. .</ref>
started	State Start Event	yes	Related start event. This field should not repeat the "est_duration" field, if it already exists.
completed	State End Event	no	The related end event if the state is completed. This field should not repeat the "disposition" and "disposition_desc" fields if they already exist.
est_duration	integer	no	The estimated time for

Field	Type	Mandatory	Description
			completing the state, in seconds.
duration	integer	no	The total duration, in milliseconds. Only relevant for completed states.
disposition	long or string	no	ID of the Business disposition ID. Only relevant for completed states.
disposition_desc	string	no	The reason for assigning the business disposition with the service element. Limited to 256 characters.
active_tasks	Task[]	no	The array of tasks that are started but not yet complete.
completed_tasks	Task[]	no	The array of Task that are complete.
<extension>	Any JSON type	no	Task attached data as key-value pairs. You can add as many key-value pairs as needed.

<references />

Start Event List

Lists the Start Event resources used to describe the start of a service, state, or task.

Service Start Event

This resource describes information useful to manage the start or the creation of a service.

Field	Type	Mandatory	Description
customer_id	string	yes if no <i>contact_key</i>	The unique ID of the customer associated with the given service. Important Services with no <i>customer_id</i> are anonymous . In this case, the event should include a <i>contact_key</i> .
contact_key	string	yes if no <i>customer_id</i>	Key for later lookup of the service. This <i>contact_key</i> is mandatory if no customer ID is specified. For example, the application might store a PIN, or the ANI, as the contact key when the service is first started. Later, if the customer is identified, the contact key is used to lookup the existing service record and associate it with the customer.
service_type	long or string	yes	The unique ID associated with the type of service, typically the DB ID of a value in the Service Type Business Attribute. Refer to Configuration Options for more details on Business Attribute mapping.
session_id	string	no	The ID of the related

Field	Type	Mandatory	Description
			session, for instance, the orchestration session, or any other business session.
interaction_id	string	no	The ID of the related Genesys interaction. This ID can be used by other Genesys reporting products such as Stat Server, URS, Composer, and GVP.
application_type	long or string	no	The unique ID associated with the type or class application issuing the service event. May be used to group related applications, potentially across resource types.
application_id	integer	no	The unique ID, for example, Genesys DB ID, for the application issuing the service event, such as a GVP VoiceXML application or an Orchestration SCXML application.
resource_type	long or string	no	The unique ID associated with the type or class of resource which provides the service (for example, GVP, Agent Desktop, Orchestration).
resource_id	integer	no	The unique DB ID for the specific resource which provides the service. For instance: <ul style="list-style-type: none"> the Genesys DB ID of a specific GVP or orchestration platform. the DB ID of a given agent, according to the context.
media_type	long or string	no	The media type applicable to the given service, for instance, email, voice, chat, etc.
est_duration	integer	no	The estimated service

Field	Type	Mandatory	Description
			duration, in seconds.
timestamp	date/time	no	The UTC time at which the event was raised, with a precision of milliseconds, using the ISO 8601 representation: [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z. If the application does not specify this timestamp, the server does it when the event is processed.
expiration_time	string	no	Time at which the service should expire, using the ISO 8601 representation: [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z. For example : "2016-09-02T08:00:00.000Z" Important This expiration time applies to the whole conversation (service, states, tasks), not only to the given service.
ttl	integer	no	Time to live (TTL) in seconds of the full conversation (service, states, tasks) starting from this event time. If the query includes both the ttl and expiration_time parameters, GMS takes into account only the ttl value and ignores the expiration_time. Important This TTL applies to the whole conversation (service, states, tasks), not only to the given service.
auto_complete_after	long	no	Time in seconds to automatically complete a service after its last update. If a service or its states or its tasks are not updated within

Field	Type	Mandatory	Description
			the specified period, the service (including states and tasks) is terminated.
last_modified	long	no	Time in milliseconds since the last update of the Service object, or of one of its subordinate States and Tasks.
<extension>	Any JSON type	no	Service attached data as key-value pairs. You can add as many key-value pairs as needed.

Related Operations

- [Start Service](#)
- [Associate Service](#)

State Start Event

This resource describes the start of the service state.

Field	Type	Mandatory	Description
service_id	integer	yes	The unique 64-bit ID of the service.
state_type	long or string	yes	<p>The unique ID associated with the state type, typically, the DB ID of a value in the Service Type Business Attribute.</p> <p>Refer to Configuration Options for more details on Business Attribute mapping. Enumerated values may be from the following:</p> <ol style="list-style-type: none"> 1. Customer identification 2. Service identification 3. Assign service agent 4. Waiting for service agent

Field	Type	Mandatory	Description
			<ul style="list-style-type: none"> 5. Offering another service while waiting for agent 6. Offering callback 7. Callback pending 8. Delivering service 9. Waiting for customer input 10. Offering another service while delivering service
previous_state_id	integer	no	The ID of the previous state.
session_id	string	no	The ID of the related session, for instance, the orchestration session, or any other business session.
interaction_id	string	no	The ID of the related Genesys interaction. This ID can be used by other Genesys reporting products such as Stat Server, URS, Composer, and GVP.
application_type	long or string	no	The unique ID associated with the type or class<ref name="business"/> of application issuing the service event. May be used to group related applications, potentially across resource types.
application_id	integer	no	The unique ID, for example, Genesys DB ID, for the application issuing the service event, such as a GVP VoiceXML application or an Orchestration SCXML application.
resource_type	long or string	no	The unique ID associated with the type or class of resource which provides the service (for example, GVP, Agent Desktop,

Field	Type	Mandatory	Description
			Orchestration).
resource_id	integer	no	The unique DB ID for the specific resource which provides the service, for instance: <ul style="list-style-type: none"> the Genesys DB ID of a specific GVP or orchestration platform. the DB ID of a given agent, according to the context.
media_type	long or string	no	The media type applicable to the given state, for instance, email, voice, chat, etc.
est_duration	integer	no	The estimated state duration in seconds.
timestamp	date/time	no	The UTC time at which the event was raised, with a precision of milliseconds, using the ISO 8601 representation: [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z. If the application does not specify this timestamp, the server does it when the event is processed.
<extension>	Any JSON type	no	State attached data as key-value pairs. You can add as many key-value pairs as needed.

Related Operations

- [Complete State](#)
- [Perform State Transition](#)
- [Query State by ID](#)
- [Query States](#)
- [Start State](#)

Task Start Event

Describes the start of a Task.

Field	Type	Mandatory	Description
service_id	integer	yes	The unique 64-bit ID of the service.
state_id	integer	no	The 32-bit integer ID of the state. See State .
task_type	long or string	yes	The unique ID associated with the type of task, typically the DB ID of a value in a Business Attribute representing customer service tasks for the given organization. Refer to Configuration Options for more details on Business Attribute mapping.
session_id	string	no	The ID of the related session, for instance, the orchestration session, or any other business session.
interaction_id	string	no	The ID of the related Genesys interaction. This ID can be used by other Genesys reporting products such as Stat Server, URS, Composer, and GVP.
application_type	long or string	no	The unique ID associated with the type or class of application issuing the service event. May be used to group related applications, potentially across resource types.
application_id	integer	no	The unique ID (such as Genesys DB ID) for the application issuing the event, such as a GVP VoiceXML application or an Orchestration SCXML application.
resource_type	long or string	no	The unique ID associated with the type or class of resource which provides the

Field	Type	Mandatory	Description
			service (for example, GVP, Agent Desktop, Orchestration).
resource_id	integer	no	The unique DB ID for the specific resource which provides the service. For instance: <ul style="list-style-type: none"> the Genesys DB ID of a specific GVP or orchestration platform. the DB ID of a given agent, according to the context.
media_type	long or string	no	The media type applicable to the given task, for instance, email, voice, chat, etc.
est_duration	integer	no	The estimated task duration, in seconds.
timestamp	date/time	no	The UTC time at which the event was raised, with a precision of milliseconds, using the ISO 8601 representation: [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z. If the application does not specify this timestamp, the server does it when the event is processed.
<extension>	Any JSON type	no	Task attached data as key-value pairs. You can add as many key-value pairs as needed.

Task

Purpose: Represents a Task.

Description

Describes one task.

Tasks allow your application to divide a **State** into a collection of tasks, each with its own disposition. Zero or more tasks can be executed as part of the Service Delivery state. Your application is responsible for managing these tasks.

Important

If the state is optional, your application can associate the task with a service instead of a state.

Resource

This table lists the resource fields that your application can deal with through operations. According to the operation, your application may not use the totality of the resource.

Field	Type	Mandatory	Description
state_id	integer	no	The 32-bit integer ID for the state associated with the task.
task_id	integer	no	The 32-bit integer ID of the task, assigned by the UCS when the task is started.
service_id	integer	no	The service's unique ID if the task is not part of a top-level state or service resource which includes a service ID. <ref name="toplevel">Only top-level representations include this field. For instance, if your application retrieves a service

Field	Type	Mandatory	Description
			which includes nested states and tasks, only the top-level service representations contain this ID.</ref>
started	Task Start Event	yes	Related start event. This event should not include the "est_duration" field, if it already exists at the task level.
completed	Task End Event	no	The related end event if this task is completed. This event should not repeat the "disposition" and "disposition_desc" fields if they already exist at the task level.
customer_id	string	no	The customer's unique ID<ref name="toplevel" />.
est_duration	integer	no	The estimated time for completing the item, in seconds.
duration	integer	no	The total duration, in milliseconds. Only relevant for completed tasks.
disposition	long or string	no	ID or name of the Business Disposition ID associated with the task. See Business Attributes for further details. Only relevant for completed tasks.
disposition_desc	string	no	The reason for assigning the business disposition with the service element. Limited to 256 characters.
<extension>	Any JSON type	no	Task attached data as key-value pairs. You can add as many key-value pairs as needed.

<references />

Related Operations

- [Complete Task](#)
- [Query Task by ID](#)
- [Query Tasks](#)
- [Start Task](#)

Customer Profile API

- Profile Operations
 - [POST /profiles](#)
 - [GET /profiles](#)
 - [POST /profiles/{customer id}/extensions](#)
 - [GET /profiles/{customer id}](#)
 - [PUT /profiles/{customer id}](#)
 - [PUT /profiles/{customer id}/extensions/{ext name}/by/unique](#)
 - [PUT /profiles/{customer id}/extensions/{ext name}/delete/by/unique](#)
 - [PUT profiles/{customer_id}/merge/{src_id}/](#)
- Schema and Identification Key Operations
 - [GET /metadata/profiles/](#)
 - [POST /metadata/profiles/extensions](#)
 - [GET /metadata/profiles/extensions](#)
 - [GET /metadata/identification-keys](#)
 - [POST /metadata/identification-keys](#)
 - [GET {content-type} /metadata](#)
 - [DELETE /metadata/profiles/extensions/{extension-name}](#)
 - [DELETE /metadata/identification-keys/{id_key-name}](#)
- Server Operations
 - [GET /server/status](#)
 - [POST /server/mode](#)
- Interaction Resources
 - [Chat Interaction](#)
 - [Callback Interaction](#)
 - [E-mail Interactions](#)
 - [Phone Call Interaction](#)
- Profile Resources
 - [Customer Profile](#)
 - [Profile Extension](#)

- [Identification Key](#)

Interaction Operations

Important

Prerequisites: You need to **enable** profiles in UCS.

Purpose: Groups operations which handle multimedia interactions.

Interaction Resources

- [Callback Interaction](#)
- [Chat Interaction](#)
- [E-mail Interaction](#)
- [Phone Call Interaction](#)

Interaction Operations

- [GET /customers/{customer id}/interactions](#)
- [GET /services/{service id}/interactions](#)
- [GET /interactions/{interaction id}](#)

Query Interactions

Important

Prerequisites: You need to **enable** profiles in UCS.

- **GET** /customers/{customer_id}/interactions
- **GET** /interactions/{interaction_id}

Description

Retrieves multimedia interactions and their data.

Operation

Queries Multimedia Interactions

ID	CV.WS.IXN.2		
Method	GET		
URL	<ol style="list-style-type: none"> 1. By Customer: /customers/{customer_id}/interactions 2. By Interaction ID: /interactions/{interaction_id} 		
Name	Type	Mandatory	Description
URI Parameters			
{customer_id}	string	yes if query is by customer.	Customer ID, required for URI 1.
{service_id}	integer	yes if query is by service.	Service ID, required for URI 2.
{interaction_id}	integer	yes if query is by interaction.	Service ID, required for URI 3.

Important

The output sample shows a "structured_text" parameter which provides a URL to the

structured text content (which is not returned as part of the query): the user application performs a separate GET to the given URL.

Similarly, the following URLs are available for GETting interaction content:

- /interactions/\${interaction_id}/plain_text
- /interactions/\${interaction_id}/structured_text
- /interactions/\${interaction_id}/binary_content

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response	
HTTP code	200
HTTP message	OK
Body	An array of one or more Interaction Resources.

Example

Operation

GET <http://ucshost:8080/interactions/00001a57JGQ00BW8>

Result

```
{
  "external_id" : "00AF4A7951D60009",
  "established_date" : "2009-08-05T09:33:15.000Z",
  "text" : "http://ucshost:8080/interactions/00001a57JGQ00BW8/plain_text",
  "media_type" : "chat",
  "udata" : {
    "RTargetTypeSelected" : "2",
    "RRequestedSkillCombination" : "",
    "ChatServerHost" : "SUITE80",
    "CustomerSegment" : "default",
    "RTargetObjectSelected" : "Agent Group Everybody",
    "RTargetRuleSelected" : "",
    "RTargetPlaceSelected" : "Place_1002",
    "RTargetAgentGroup" : "Agent Group Everybody",
    "RTargetObjSelDBID" : "123",
    "ServiceType" : "default",
    "_smsSrcNumber" : "+33298143456",
    "RTargetAgentSelected" : "1002",
```



```
"RTenant" : "defaultTenant",
"RVQID" : "",
"CBR-contract_DBIDs" : "",
"CBR-Interaction_cost" : "",
"ChatServerPort" : "23874",
"_smsDestNumber" : "+33298143322",
"RTargetRequested" : "Agent Group Everybody",
"CBR-IT-path_DBIDs" : "",
"RStrategyDBID" : "480",
"CBR-actual_volume" : "",
"RStrategyName" : "sms-session-inbound"
},
"binary_content" : "http://ucshost:8080/interactions/00001a57JGQ00BW8/binary_content",
"structured_text" : "http://ucshost:8080/interactions/00001a57JGQ00BW8/structured_text",
"status" : 3,
"subject" : "SMS chat",
"timeshift" : 1,
"type" : "Inbound",
"interaction_id" : "00001a57JGQ00BW8",
"allow_children" : true,
"end_date" : "2009-08-05T09:34:04.140Z",
"owner_id" : 124,
"sub_type" : "InboundNew",
"released_date" : "2009-08-05T09:33:59.000Z",
"entity_type" : 2,
"creator_app_id" : 175,
"thread_id" : "00001a57JGQ00BW9",
"start_date" : "2009-08-05T09:33:10.000Z",
"structured_text_mime" : "text/xml",
"customer_id" : "00001a57JGQ00BVU"
}
```

Interaction Resources

Important

Prerequisites: You need to **enable** profiles in UCS.

Lists interaction resources.

- [Callback Interaction](#)
- [Chat Interaction](#)
- [E-mail Interaction](#)
- [Phone Call Interaction](#)

Callback Interaction

Purpose

Provides context information relevant to a given multimedia callback interaction. A customer can ask for a callback if he or she is not available for the moment. In that case, based on the application's inputs, the system schedules the callback interaction at a time which meets the user's expectations.

Resource

Field	Type	Mandatory	Description
Callback-related information			
callback_status	integer	no	Status for the callback. <ul style="list-style-type: none"> • 0-UNKNOWN • 1-RECEIVED • 2-TIMER • 3-QUEUED • 4-DISTRIBUTED • 5-SUBMITTED_TO_AGT • 6-REJECTED_BY_AGT • 7-CANCELED_BY_AGT • 8-PROCESSED_BY_AGT • 9-FINAL • 10-NOT_QUEUED • 11-ERROR • 12-REPLACED • 13-AGT_TRANSFER • 14-ENDTIME_EXPIRED
desired_response_type	string	yes	Possible types: <ul style="list-style-type: none"> • Voice • VoIP • Email

Field	Type	Mandatory	Description
start_time	date/time	yes	Time for starting the callback.
end_time	date/time	no	Limit time for executing the callback.
customer_number	string	yes	Contact information for the customer. Depends on the media type (i.e. phone number, IP address for VoIP, or email address).
attempts	integer	yes	The number of callback attempts that have been made.
dn	string	no	The DN to which the Callback server sent the request.
location	string	no	Name of the switch to which the DN belongs.
callback_server_id	string	no	Identifies the Callback server that handled the request.
call_result	integer	no	One of the following results: <ul style="list-style-type: none"> • 0 - OK • 1 - TRANSFERRED • 2 - CONFERENCED • 3 - GENERAL_ERROR • 4 - SYSTEM_ERROR • 5 - REMOTE_RELEASE • 6 - BUSY • 7 - NO_ANSWER • 8 - SIT_DETECTED • 9 - ANSWERING_MACHINE • 10- ALL_TRUNKS_BUSY • 11- SIT_INVALID_NUM • 12- SIT_VACANT • 13- SIT_OPER_INTERCEPT

Field	Type	Mandatory	Description
			<ul style="list-style-type: none"> • 14- SIT_UNKNOWN • 15- SIT_NO_CIRCUIT • 16- SIT_REORDER • 17- SIT_FAX_DETECTED • 18- QUEUE_FULL • 19- CLEARED • 20- OVERFLOWED • 21- ABANDONED • 22- REDIRECTED • 23- FORWARDED • 24- CONSULT • 25- PICKEDUP • 26- DROPPED • 27- DROPPED_NO_ANSWER • 28- UNKNOWN • 29- COVERED • 30- CONVERSE_ON • 31- BRIDGED • 32- SILENCE • 33- ANSWER • 34- NU_TONE • 35- NO_DIAL_TONE • 36- NO_PROGRESS • 37- NO_RING_BACK • 38- NO_ESTABLISHED • 39- PAGER_DETECTED • 40- WRONG_PARTY • 41- DIAL_ERROR • 42- CALL_DROPPED_ERROR

Field	Type	Mandatory	Description
			<ul style="list-style-type: none"> 43- SWITCH_ERROR 44- NO_FREE_PORT_ERROR 45- TRANSFER_ERROR 46- STALE 47- AGT_CALLBACK_ERROR 48- GRP_CALLBACK_ERROR 49- DEAFENED 50- HELD 51- DO_NOT_CALL 52- CANCEL_RECORD 53- WRONG_NUMBER
callback_type	integer	no	Possible types: <ul style="list-style-type: none"> 0-UNKNOWN 1-ASAP 2-SCHEDULED
Common interaction fields			
customer_id	string	yes	The ID of the customer.
interaction_id	string	yes	The Genesys GUID (or Call ID) for the interaction.
status	integer	yes	Possible statuses: <ul style="list-style-type: none"> 0-NEW 1-PENDING 2-IN_PROCESS 3-STOPPED
entity_type	integer	yes	Possible types: <ul style="list-style-type: none"> 0-EMAIL_IN 1-EMAIL_OUT 2-CHAT

Field	Type	Mandatory	Description
			<ul style="list-style-type: none"> • 3-PHONE_CALL • 5-CALLBACK • 6-COBROWSE • 7-INTERACTION (open media)
type	string	yes	The type of the interaction (such as inbound, outbound, etc.)
sub_type	string	no	The application-specific subtype which further classifies the attribute "type".
media_type	string	yes	<p>The media type of the interaction:</p> <ul style="list-style-type: none"> • voice • email • chat • callback • and so on.
parent_id	string	no	The ID of the parent interaction, if applicable.
thread_id	string	no	The ID of the thread to which this interaction should be appended, or the ID of the interaction at the root of the thread.
external_id	string	no	The reference identifier for an external system.
owner_id	integer	no	The DB ID of the Person who owns the interaction. (The Person is an object defined in the Genesys Configuration Server.)
creator_app_id	integer	yes	The DB ID of the application that created the interaction.
start_date	date/time	yes	The UTC time of the interaction's creation, using the ISO 8601 representation<ref

Field	Type	Mandatory	Description
			name="iso8601"> http://en.wikipedia.org/wiki/ISO_8601 </ref>: [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z
end_date	date/time	no	The UTC time of the interaction's end, using the ISO 8601 representation<ref name="iso8601" />: [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z
timeshift	integer	no	The number of minutes to add to or to subtract from the specified UTC<ref name="iso8601" /> time in order to get the local time of the interaction's creation.
category_id	string	no	The category assigned to this interaction by the Classification Server or the desktop's manual override.
allow_children	Boolean	no	True if child interactions can be added. Important Default value is true.
subject	string	no	An arbitrary textual subject for the interaction.
stopped_reason	string	no	The application-specified reason for which the interaction ended.
text	text	no	Plain text for the interaction.
structured_text	url	no	The URL of the interaction's structured text.
structured_text_mime	string	yes if structured_text	The mime type<ref name="mime"> http://www.iana.org/assignments/media-types </ref> for the structured text, if specified (such as text/html). Mandatory if the parameter "structured_text" is specified.

Field	Type	Mandatory	Description
binary_content	url	no	The URL of the the interaction's binary content..
binary_content_mime	string	yes if binary_content	The mime type<ref name="mime"> http://www.iana.org/assignments/media-types </ref> for the binary content, if specified (such as text/html).
binary_content_size	integer	yes if binary_content	The size of the binary content in bytes. Mandatory if the parameter "binary_content" is specified.
udata	hash	no	A hash of key value pairs for the user data attached to the interaction.

References

<references />

Chat Interaction

Purpose

Describes chat data for a multimedia interaction.

Resource

Name	Type	Mandatory	Description
Chat-related information			
established_date	date/time<ref name="iso8601">UTC Time, using the ISO 8601 representation : [YYYY]-[MM]-[DD]T[HH]:[mm]:[ss].[SSS]Z</ref>	no	The date/time when the chat session was established with a contact center resource.
released_date	date/time<ref name="iso8601"/>	no	The date/time when the chat session was released from the contact center resource.
Common Interaction fields			
customer_id	string	yes	The ID of the customer.
interaction_id	string	yes	The Genesys GUID (or Call ID) for the interaction.
status	integer	yes	Possible statuses: <ul style="list-style-type: none"> 0-NEW 1-PENDING 2-IN_PROCESS 3-STOPPED
entity_type	integer	yes	Possible types: <ul style="list-style-type: none"> 0-EMAIL_IN 1-EMAIL_OUT 2-CHAT 3-PHONE_CALL

Name	Type	Mandatory	Description
			<ul style="list-style-type: none"> • 5-CALLBACK • 6-COBROWSE • 7-INTERACTION (open media)
type	string	yes	The type of the interaction (such as inbound, outbound, etc.)
sub_type	string	no	The application-specific subtype which further classifies attribute "type".
media_type	string	yes	The media type of the interaction: <ul style="list-style-type: none"> • voice • email • chat • callback • and so on.
parent_id	string	no	The ID of the parent interaction, if applicable.
thread_id	string	no	The ID of the thread to which this interaction should be appended, or the ID of the interaction at the root of the thread.
external_id	string	no	The reference identifier for an external system.
owner_id	integer	no	The DB ID of the Person who owns the interaction. (The Person is an object defined in the Genesys Configuration Server.)
creator_app_id	integer	yes	The DB ID of the application that created the interaction.
start_date	date/time<ref name="iso8601"/>	yes	The UTC time of the interaction's creation.
end_date	date/time<ref name="iso8601"/>	no	The UTC time of the interaction's end.
timeshift	integer	no	The number of minutes

Name	Type	Mandatory	Description
			to add to or to subtract from the specified UTC<ref name="iso8601" /> time in order to get the local time of the interaction's creation.
category_id	string	no	The category assigned to this interaction by the Classification Server or the desktop's manual override.
allow_children	boolean	no	True if child interactions can be added. Important Default value is true.
subject	string	no	An arbitrary textual subject for the interaction.
stopped_reason	string	no	The application-specified reason for which the interaction ended.
text	text	no	Plain text for the interaction.
structured_text	url	no	The URL of the interaction's structured text.
structured_text_mime	string	yes if structured_text	The mime type<ref name="mime"> http://www.iana.org/assignments/media-types </ref> for the structured text, if specified (such as text/html). Mandatory if the parameter "structured_text" is specified.
binary_content	url	no	The URL of the the interaction's binary content..
binary_content_mime	string	yes if binary_content	The mime type<ref name="mime"> http://www.iana.org/assignments/media-types </ref> for the binary content, if specified (such as text/html).
binary_content_size	integer	yes if binary_content	The size of the binary

Name	Type	Mandatory	Description
			content in bytes. Mandatory if the parameter "binary_content" is specified.
udata	hash	no	A hash of key value pairs for the user data attached to the interaction.

<references />

E-mail Interactions

Purpose

Provides information for email address and interaction management.

E-mail Address

E-Mail Address Fields

Field	Type	Required	Description
address	string	yes	The email address.
personal	string	no	The personal part of the email address, for instance, a displayable such as a localized name.

Inbound E-mail

Inbound emails are email interactions which are received by the application or the Genesys Solution.

E-mail In

Field	Type	Required	Description
Inbound-specific fields			
from	E-mail Address	no	The sender's email address.
reply_to	E-mail Address	no	The address used to reply.
to	E-mail Address[]	no	The list of recipient addresses, as an array of email addresses.
cc	E-mail Address[]	no	The list of addresses for copied recipients, as an array of email addresses.
sent_date	date/time	no	The UTC Wikipedia ISO 8601 date/time at which the email was sent.
Common Interaction fields			

Field	Type	Required	Description
customer_id	string	yes	The ID of the customer.
interaction_id	string	yes	The Genesys GUID (or Call ID) for the interaction.
status	integer	yes	Possible statuses: <ul style="list-style-type: none"> • 0-NEW • 1-PENDING • 2-IN_PROCESS • 3-STOPPED
entity_type	integer	yes	Possible types: <ul style="list-style-type: none"> • 0-EMAIL_IN • 1-EMAIL_OUT • 2-CHAT • 3-PHONE_CALL • 5-CALLBACK • 6-COBROWSE • 7-INTERACTION (open media)
type	string	yes	The type of the interaction (such as inbound, outbound, etc.)
sub_type	string	no	The application-specific subtype which further classifies the attribute "type".
media_type	string	yes	The media type of the interaction: <ul style="list-style-type: none"> • voice • email • chat • callback • and so on.
parent_id	string	no	The ID of the parent interaction, if applicable.
thread_id	string	no	The ID of the thread to which this interaction

Field	Type	Required	Description
			should be appended, or the ID of the interaction at the root of the thread.
external_id	string	no	The reference identifier for an external system.
owner_id	integer	no	The DB ID of the Person who owns the interaction. (The Person is an object defined in the Genesys Configuration Server.)
creator_app_id	integer	yes	The DB ID of the application that created the interaction.
start_date	date/time	yes	The UTC time of the interaction's creation, using the ISO 8601 representation : [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z
end_date	date/time	no	The UTC time of the interaction's end, using the ISO 8601 representation : [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z
timeshift	integer	no	The number of minutes to add to or to subtract from the specified UTC ISO 8601 time in order to get the local time of the interaction's creation.
category_id	string	no	The category assigned to this interaction by the Classification Server or the desktop's manual override.
allow_children	Boolean	no	True if child interactions can be added. Important Default value is true.
subject	string	no	An arbitrary textual subject for the interaction.
stopped_reason	string	no	The application-specified reason for which the interaction

Field	Type	Required	Description
			ended.
text	text	no	Plain text for the interaction.
structured_text	url	no	The URL of the interaction's structured text.
structured_text_mime	string	yes if structured_text	The mime type for the structured text, if specified (such as text/html). Mandatory if the parameter "structured_text" is specified.
binary_content	url	no	The URL of the the interaction's binary content..
binary_content_mime	string	yes if binary_content	The mime type for the binary content, if specified (such as text/html).
binary_content_size	integer	yes if binary_content	The size of the binary content in bytes. Mandatory if the parameter "binary_content" is specified.
udata	hash	no	A hash of key value pairs for the user data attached to the interaction.

Outbound E-mail

Outbound emails are email interactions which are sent from the application or from the Genesys Solution.

E-mail Out

Field	Type	Mandatory	Description
Outbound-Specific fields			
from	E-mail Address	no	The sender's email address.
reply_to	E-mail Address	no	The email address used to send a reply.
to	E-mail Address[]	no	The list of recipient addresses, as an array of email addresses.

Field	Type	Mandatory	Description
cc	E-mail Address[]	no	The list of addresses for copied recipients, as an array of email addresses.
bcc	E-mail Address[]	no	The list of addresses for blind-copied recipients, as an array of email addresses.
sent_date	date/time	no	The UTC<ref> http://en.wikipedia.org/wiki/ISO_8601 </ref> date/time at which the email was sent.
reference_id	integer	no	Contains the external identifier (message ID) of the parent interaction.
reviewer_id	integer	no	The application-specified ID of the Person who reviewed the email.
standard_response_id	string	no	The ID of the standard response item from which this email was generated.
Common Interaction fields			
customer_id	string	yes	The ID of the customer.
interaction_id	string	yes	The Genesys GUID (or Call ID) for the interaction.
status	integer	yes	Possible statuses: <ul style="list-style-type: none"> • 0-NEW • 1-PENDING • 2-IN_PROCESS • 3-STOPPED
entity_type	integer	yes	Possible types: <ul style="list-style-type: none"> • 0-EMAIL_IN • 1-EMAIL_OUT • 2-CHAT • 3-PHONE_CALL • 5-CALLBACK • 6-COBROWSE

Field	Type	Mandatory	Description
			<ul style="list-style-type: none"> 7-INTERACTION (open media)
type	string	yes	The type of the interaction (such as inbound, outbound, etc.)
sub_type	string	no	The application-specific subtype which further classifies the attribute "type".
media_type	string	yes	<p>The media type of the interaction:</p> <ul style="list-style-type: none"> voice email chat callback and so on.
parent_id	string	no	The ID of the parent interaction, if applicable.
thread_id	string	no	The ID of the thread to which this interaction should be appended, or the ID of the interaction at the root of the thread.
external_id	string	no	The reference identifier for an external system.
owner_id	integer	no	The DB ID of the Person who owns the interaction. (The Person is an object defined in the Genesys Configuration Server.)
creator_app_id	integer	yes	The DB ID of the application that created the interaction.
start_date	date/time	yes	<p>The UTC time of the interaction's creation, using the http://en.wikipedia.org/wiki/ISO_8601 representation: [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z</p>

Field	Type	Mandatory	Description
end_date	date/time	no	The UTC time of the interaction's end, using the ISO 8601 representation: [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z
timeshift	integer	no	The number of minutes to add to or to subtract from the specified UTC time in order to get the local time of the interaction's creation.
category_id	string	no	The category assigned to this interaction by the Classification Server or the desktop's manual override.
allow_children	Boolean	no	True if child interactions can be added. Default value is true.
subject	string	no	An arbitrary textual subject for the interaction.
stopped_reason	string	no	The application-specified reason for which the interaction ended.
text	text	no	Plain text for the interaction.
structured_text	url	no	The URL of the interaction's structured text.
structured_text_mime	string	yes if structured_text	The mime type for the structured text, if specified (such as text/html). Mandatory if the parameter "structured_text" is specified.
binary_content	url	no	The URL of the the interaction's binary content..
binary_content_mime	string	yes if binary_content	The mime type for the binary content, if specified (such as text/html).
binary_content_size	integer	yes if binary_content	The size of the binary content in bytes. Mandatory if the parameter

Field	Type	Mandatory	Description
			"binary_content" is specified.
udata	hash	no	A hash of key value pairs for the user data attached to the interaction.

Phone Call Interaction

Important

Prerequisites: You need to **enable** profiles in UCS.

Purpose: Describes phone call fields.

Description

Provides fields for a phone call interaction.

Resource

Field	Type	Mandatory	Description
duration	integer	no	The duration of the call in seconds.
outcome	string	no	The application-specified outcome of the call.
phone_number	string	no	The phone number called.
tconnection_id	string	no	The connection ID assigned to the call by the Genesys T-Server.
customer_id	string	yes	The ID of the customer.
interaction_id	string	yes	The Genesys GUID (or Call ID) for the interaction.
status	integer	yes	Possible statuses: <ul style="list-style-type: none"> • 0-NEW • 1-PENDING • 2-IN_PROCESS • 3-STOPPED
entity_type	integer	yes	Possible types:

Field	Type	Mandatory	Description
			<ul style="list-style-type: none"> • 0-EMAIL_IN • 1-EMAIL_OUT • 2-CHAT • 3-PHONE_CALL • 5-CALLBACK • 6-COBROWSE • 7-INTERACTION (open media)
type	string	yes	The type of the interaction (such as inbound, outbound, and so on).
sub_type	string	no	The application-specific subtype which further classifies attribute "type".
media_type	string	yes	The media type of the interaction, including: <ul style="list-style-type: none"> • voice • email • chat • callback
parent_id	string	no	The ID of the parent interaction, if applicable.
thread_id	string	no	The ID of the thread to which this interaction should be appended, or the ID of the interaction at the root of the thread.
external_id	string	no	The reference identifier for an external system.
owner_id	integer	no	The DB ID of the Person who owns the interaction. (The Person is an object defined in the Genesys Configuration Server.)
creator_app_id	integer	yes	The DB ID of the application that created the interaction.

Field	Type	Mandatory	Description
start_date	date/time	yes	The UTC time of the interaction's creation, using the ISO 8601 representation<ref name="iso8601"> http://en.wikipedia.org/wiki/ISO_8601 </ref>: [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z
end_date	date/time	no	The UTC time of the interaction's end, using the ISO 8601 representation<ref name="iso8601" />: [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z
timeshift	integer	no	The number of minutes to add to or subtract from the specified UTC<ref name="iso8601" /> time in order to get the local time of the interaction's creation.
category_id	string	no	The category assigned to this interaction by the Classification Server or the desktop's manual override.
allow_children	Boolean	no	Default value of true indicates that child interactions can be added.
subject	string	no	An arbitrary textual subject for the interaction.
stopped_reason	string	no	The application-specified reason for which the interaction ended.
text	text	no	Plain text for the interaction.
structured_text	url	no	The URL of the interaction's structured text.
structured_text_mime	string	yes if structured_text	The mime type<ref name="mime"> http://www.iana.org/assignments/media-types </ref> for the structured text, if specified (text/html). Mandatory if the

Field	Type	Mandatory	Description
			parameter "structured_text" is specified.
binary_content	url	no	The URL of the the interaction's binary content.
binary_content_mime	string	yes if binary_content	The mime type<ref name="mime"> http://www.iana.org/assignments/media-types </ref> for the binary content, if specified (text/html).
binary_content_size	integer	yes if binary_content	The size of the binary content in bytes. Mandatory if the parameter "binary_content" is specified.
udata	list	no	A list of key-value pairs for the user data attached to the interaction.

<references />

Profile Operations

Important

Prerequisites: You need to **enable** profiles in UCS.

Purpose: Groups operations which handle profiles.

- [POST /profiles](#)
- [GET /profiles](#)
- [POST /profiles/\\${customer id}/extensions](#)
- [GET /profiles/\\${customer id}](#)
- [PUT /profiles/\\${customer id}](#)
 - [PUT /profiles/\\${customer id}/extensions/\\${ext name}/by/unique](#)
 - [PUT /profiles/\\${customer id}/extensions/\\${ext name}/delete/by/unique](#)
 - [POST /profiles/import](#)
 - [DELETE /profiles/\\${customer_id}](#)
 - [PUT profiles/\\${customer_id}/merge/\\${src_id}/](#)

Bulk Profile Import

Important

Prerequisites: You need to **enable** profiles in UCS.

POST /profiles/import

Description

Imports several customer profiles in a single operation. Your application should use this operation if you need to import a large number of profiles into the UCS database.

- The operation supports the import of .csv files if the file is already uploaded on UCS' local file system.
- The operation manages:
 - Database consistency (data type conversion according to metadata).
 - Propagation of identification keys.
 - Backward compatibility with the APIs.

Warning

You cannot submit the profiles as a JSON array.

The operation's description of the supported format must be compliant with **RFC4180**:

- The first record must contain the column name for each field.
- There is one record per line and each line terminates with a line feed.
- Commas separate fields by default; if you wish to use another separator, set up the delimiter field of your operation.
- Spaces are considered as part of a field and will not be ignored (trimmed).

This operation returns the list of contacts which were not imported, with descriptions of the errors or ambiguities which prevented the server from adding them to the database.

Roles

If **role-based access control** is enabled in your environment, your application will need to be assigned the following roles in order to successfully perform a bulk import:

- UCS.Customer.executeBulkImport (Perform Bulk Import)
- UCS.Customer.createProfile (Create Customer Profile)
- UCS.Customer.createProfileExtension (Create Profile Extension)

Important

Your application must be assigned the `UCS.Customer.createProfileExtension` role only if profile extensions will be created by the bulk import.

Operation

Prerequisites:

- The Customer Profile Schema exists.

Your operation's header must comply with the following requirements:

- Content-Type is set to 'multipart/form-data'.
- Request is composed of a first part of type 'application/json', named 'description' and containing the JSON body as described below.

ID	CV.WS.PROF.IMPORT.1		
Method	POST		
URL	/profiles/import		
Parameter	Type	Mandatory	Description
Body			
format	"JSON" Not supported "CSV"	yes	Defines the format of the submitted data.
profiles	string	no	File path, for instance 'C:/profiles.csv'. Your application is responsible for uploading the file on the server's local filesystem.

ID	CV.WS.PROF.IMPORT.1		
id-key	string	no	A valid identification key. If specified, the server checks to see whether the profile already exists prior to the insertion. If the profile does exist in the database, the record containing the duplicate profile is skipped.
delimiter	char or "," by default	no	Defines the character used to separate the submitted profiles.
batch-size	integer 64 by default	no	The number of records to insert or update before a database COMMIT is issued (default to 64).
max-errors	integer	no	The number of errors to allow before ending the bulk load. An error can indicate that a given record was not parsed correctly, or that an ambiguity exists with an existing customer record (see response below).
encoding	string	no	Encoding character to use for file import. Default is UTF-8<ref> UTF-8 without BOM, see related bug http://bugs.sun.com/view_bug.do;jsessionid=fdff2c415891f7
start-from-index	integer	no	The record number from which to start importing.
create-log	boolean	no	This flag turns extended import results logging on or off. The log data is written to an output file that has the same name as the input file, with an extension of .log appended. The default value is false.

Response

The Context Management Service API answers every request with an HTTP code. The following table

shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

HTTP code	200
HTTP message	OK
Body	<pre> {"errors": [{"index": '\$index_id1', "reason": '\$reason_1'}, {"index": '\$index_id2', "reason": '\$reason_2'}], "ambiguous": [{"index": '\$index_id2', "reason": '\$reason_3'}] } </pre> <p>where:</p> <ul style="list-style-type: none"> • `\${index_idn} is the index of the imported record in the source array. • "Ambiguous" lists the source data items which were ignored because of <i>Unresolved identity</i>: The given identification key is not uniquely identifying the customer; for instance, attempting to use "Name" to identify the customer, but there are more than one customer with the same name. • "Ambiguous" lists the source data items which were invalid for some reason. The reason description is one of the errors which can occur when inserting a profile (see Create Customer Profile).

Examples

Raw HTTP

Operation

```

POST /cv/profiles/import HTTP/1.1
Keep-Alive: 10
Content-Length: 320
Content-Type: multipart/form-data; boundary=zaHf8xb3LfV0XZ0o2KUyqWD4Zlfumh0R9uyMN
Host: localhost:9090
Connection: Keep-Alive
Expect: 100-Continue

```

```

--zaHf8xb3LfV0XZ0o2KUyqWD4Zlfumh0R9uyMN
Content-Disposition: form-data; name="description"
Content-Type: application/json; charset=UTF-8
Content-Transfer-Encoding: 8bit

```

```
{ "delimiter": ",", "profiles": "D:/successContentProfiles.csv", "format": "csv", "max-errors": 20 }
--zaHf8xb3LfV0XZ0o2KUyqWD4ZLfumh0R9uyMN--
```

Here is a sample .csv file containing 10 records:

```
FirstName, LastName, PhoneNumber[2], PhoneNumber[1], EmailAddress, Address.AddressType,
Address.PostCode,Address.Address, Address.County, Address.City, Address.Country
kertuleordi,gusch,1-355-530-4087,1-204-489-2084,kertuleordi@gusch.net,7,4703,"25,
et malesuada vulputate Proin",Pomona,Germany,Fullerton
danye,hna,1-551-483-3810,1-455-658-9039,danye@hna.net,6,5101,"blabl \nDonec",Kailua,East
Timor,
Duluth
almady,vanah,1-215-196-0375,1-632-596-1889,almady@vanah.net,0,4602,facilisis fermentum
lacus.,Irving,Cambodia,Jackson
julaudricelsa,mosessaglowsket,1-363-017-4320,
1-600-591-2531,julaudricelsa@mosessaglowsket.net,
4,2644,id bibendum,Fort Lauderdale,Sudan,Orem
clemara,gonebali,1-334-657-4832,1-226-703-2238,clemara@gonebali.net,6,6552,
aliquet rutrum nonummy pellentesque ac,Chula Vista,Jamaica,Waterbury
lonni,mi,1-021-445-4631,1-716-116-8811,lonni@mi.net,7, 102,Duis vulputate lacus.
felis eu eu et sagittis ut,Allentown,Russian Federation,North Charleston
louignaelgarcecio,jecklettorneimermandetett,1-615-348-3144,1-161-040-
1742,louignaelgarcecio@jecklettorneimermandetett.net,1,6533,
nisl Suspendisse imperdiet,Jacksonville,Italy, Baltimore
jantasia,jachils,1-324-661-2865,1-220-082-1681,jantasia@jachils.net,5,0171,neque lacus.
Donec mauris Donec facilisis metus. est,Columbia,
Central African Rep,Thousand Oaks
hizellaighie,wollosch,1-207-086-4460,1-160-603-4275,hizellaighie@wollosch.net,
4,9653,Vestibulum ipsum eu sit,Albuquerque,Romania,Philadelphia
```

HTML Example

Operation

Consider the following code sample embedding the JSON request, which enables you to select the .csv file:

```
<html>
<body>
  Insert JSON request in textbox below and select your CSV file.
  <form method="post" action="http://localhost:8080/cv/profiles/import">
  <br/>'application/json' import parameters:<br/>
  <TEXTAREA name="description" rows="10" cols="80">
  { profiles:"D:/myProfiles.csv", format:"csv", delimiter:",", batch-size:64, max-errors:20
  }</TEXTAREA>
  <input type="submit" value="Send" />
  </form>
</body>
</html>
```

Result The service might return the following information:

```
{
  "errors": [
    {
      "index": 3,
      "reason": "Bad parameter 'Importing profile' reason : number of profile attributes
cannot be different than declared"
    },
  ],
}
```

```

    {
      "index": 4,
      "reason": "Bad parameter 'Invalid value' reason : \"no-integer\" is an invalid value for
type 'Integer'"
    }
  ]
  "ambiguous": [
    {
      "index": 12,
      "reason": "Some customer(s) are already matching this profile : idProfile1, idProfile2"
    },
    {
      "index": 14,
      "reason": "Some customer(s) are already matching this profile : idProfile15, idProfile232"
    }
  ]
}

```

Apache Example

The following code sample shows how to write an HTTP client for Apache which enables an import.

```

DefaultHttpClient client = new DefaultHttpClient();
HttpRequestRetryHandler retryHandler = new HttpRequestRetryHandler() {
    public boolean retryRequest(IOException exception, int executionCount, HttpContext context) {
        return false;
    }
};
client.setHttpRequestRetryHandler(retryHandler);

```

```

HttpPost httpPost = new HttpPost("http://localhost:9090/profiles/import");

```

```

JSONObject parameters = new JSONObject();
parameters.put("format", "csv");
parameters.put("profiles", "D:/myProfiles.csv");
parameters.put("delimiter", ",");
parameters.put("max-errors", 20);

```

```

MultipartEntity reqEntity = new MultipartEntity(HttpMultipartMode.STRICT);
reqEntity.addPart("description", new StringBody(parameters.toString(),
MediaType.APPLICATION_JSON.toString(), Charset.forName("UTF-8")));
httpPost.setEntity(reqEntity);

```

```

HttpResponse response = client.execute(httpPost);
HttpEntity resEntity = response.getEntity();

```

```

String res = EntityUtils.toString(resEntity, "UTF-8");

```

```

client.getConnectionManager().shutdown();

```

References

<references />

Create Customer Profile

Important

Prerequisites: You need to [enable](#) profiles in UCS.

POST /profiles

Description

Creates a Customer Profile and returns the ID created by UCS.

Operation

Prerequisites: The Customer Profile Schema exists.

ID	CV.WS.PROF.3		
Method	POST		
URL	/profiles		
Parameter	Type	Mandatory	Description
Body: Customer Profile. See Customer Profile for more information about the Customer Profile.			
customer_id	string	yes	The unique ID of the given customer. Limited to 16 characters.
<attribute n>	<ul style="list-style-type: none"> string datetime 	yes	<p>Customer attributes, where <attribute n> is the attribute name. See the configuration options for further details.<. Attributes are not part of a separated array, see the example below.</p> <p>The date/time formats are ISO 8601: [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z. Do not use Unauthorized Strings as attribute values.</p>
<extension n>	Profile Extension or	no	Extensions, where

ID	CV.WS.PROF.3		
	Profile Extension[]		<p><extension n> corresponds to the unique name of the profile extension resource.</p> <ul style="list-style-type: none"> For single-valued extensions, the extension's value is a single extension object (see Profile Extension). For multi-valued extensions, the extension's value is an array of zero or more Profile Extensions.

Response

The Context Management Service API answers every request with an HTTP code. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response	
HTTP code	201
HTTP message	Created
Header	Location: /profiles/\${customer_id} where: <ul style="list-style-type: none"> customer_id is the created customer ID.
Body	{"customer_id": customer_id} where: <ul style="list-style-type: none"> customer_id is the customer ID.

Example

Operation The following operation would create the customer profile for "Bruce Banner," as well as three associated telephone records that are grouped in the "Phone" extension. The email address *bruce.banner@marvelous.com* is the primary email attribute, while the other email address is non-

primary.

```
POST /profiles/
{
  "FirstName": "Bruce",
  "LastName": "Banner",
  "DOB": "1962-05-10",
  "EmailAddress": [
    "bruce.banner@marvelous.com",
    "b.banner@hulk.dom"
  ],
  "Phone": [
    {
      "PhoneType":0,
      "prefix":"+33",
      "PhoneNumber":"3145926535",
      "description":"family phone",
      "start_availability":"2009-12-18T18:30:00.000Z",
      "end_availability":"2009-12-18T21:40:00.000Z"
    },
    {
      "PhoneType":2,
      "prefix":"+33",
      "PhoneNumber":"6543210",
      "description":"business calls only, no sales",
      "start_availability":"2009-12-18T09:30:00.000Z",
      "end_availability":"2009-12-18T17:45:00.000Z"
    },
    {
      "PhoneType":5,
      "prefix":"+33",
      "PhoneNumber":"951357456",
      "description":""
    }
  ]
}
```

Result

```
Content-Location      http://ucsserver.mycompany.com:8080/path/profiles/00038b5SCVPU0007
{"customer_id":"0004Va58A92T0017"}
```

Delete Customer Profile

DELETE /profiles/\${customer_id}

Description

Deletes a customer profile and all its extension records and identifications keys records.

Important

Starting with the 8.5.0 release, service, state, and task data are no longer handled in UCS and therefore will not be deleted.

Operation

ID	CV.WS.PROF.8		
Method	DELETE		
URI	/profiles/\${customer_id}		
Parameter	Type	Mandatory	Description
\${customer_id}	string	yes	The unique ID of the given customer that is specified in the URI.

<references />

Response

The Context Management Service API answers with HTTP codes for every request. The following tables show the correct response for a successful request, and the response if the customer cannot be deleted. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response	
HTTP code	200
HTTP message	Success

Response

HTTP code	404
HTTP message	Not Found

Example

Operation The following operation deletes a particular customer:

```
DELETE /profiles/0004Va58A92T0018
```

Result

200 Success

Delete Record From Profile Extension

Important

Prerequisites: You need to **enable** profiles in UCS.

PUT /profiles/{customer_id}/extensions/{ext_name}/delete/by/unique

Description

Deletes a single record in a multi-valued extension. You must specify the list of 'unique' attributes which identify the record to delete. See [Unique Attributes](#) for further information.

Operation

Prerequisites: The extension record exists.

ID	CV.WS.PROF.7		
Method	PUT		
URL	/profiles/{customer_id}/extensions/{ext_name}/delete/by/unique		
Parameter	Type	Mandatory	Description
URI Parameters			
{customer_id}	string	yes	The unique ID of the given customer. Limited to 16 characters.
{ext_name}	string	yes	The name of the Profile Extension to modify.
Body Unique attributes of the Profile Extension record .			
<attribute n>	<ul style="list-style-type: none"> string integer long double date datetime 	yes	<attribute n> corresponds to the attribute name defined 'unique' in the Extension Schema.

ID	CV.WS.PROF.7		
	<ul style="list-style-type: none"> • currency • boolean 		

<references />

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response	
HTTP code	204
HTTP message	No Content

Example

Operation The following operation updates the Phone record associated with the unique "PhoneNumber":"3145926535".

```
PUT /profiles/0000Sb5U97XE000Y/extensions/Phone/delete/by/unique
{"PhoneType":0, "prefix":"+33", "PhoneNumber":"3145926535",
"description":"family phone", "start_availability":"2009-12-18T18:30:00.000Z",
"end_availability":"2009-12-18T21:40:00.000Z"}
```

Result

204 No Content

Identify Customer

Important

Prerequisites: You need to **enable** profiles in UCS.

GET /profiles

Description

Identifies a customer based on the **Identification Key** information submitted with the query. In addition to the identified customer ID, your application can retrieve profile and extension information based on the query fields. Query strings should be **URL Encoded**. If the "id_key" parameter is not specified, then UCS uses the following algorithm:

1. Starting with the identification key with the lowest ID, UCS determines the number of attributes in the key which are covered by the data specified in the "expr" parameter.
2. If all attributes are covered, then use this key. Otherwise, record the number of sequential attributes covered, starting from the left. For example, if key 1 consists of attributes (a, b, c), and "expr" contains (a, b) then the score is 2.
3. If no key is selected after iterating through all the defined keys, choose the key with the highest score from Step 2. In the event of a tie, UCS selects the key with the smallest ID.

Operation

Prerequisites

- The identification keys exist.

Important

Use the **Create Identification Key** operation to create Identification Keys.

ID	CV.WS.CUSTID.1
Method	GET
URL	/profiles

ID	CV.WS.CUSTID.1		
Field Name	Type	Mandatory	Description
URI Parameters			
id_key	string	no	The key name used for identification. UCS can infer the key based on the attributes included in the "expr" parameter. However, explicitly specifying the identification key with your application improves performance.
include_profile	token	no	<ul style="list-style-type: none"> "no" (default) to return the list of customers IDs without their profile attributes. "yes" to return all the profile attributes of the identified customers.
extensions	string	no	The names of the extension to return with the customer profiles. Your application can specify multiple values separated with comas, such as: <i>extensions=contacts,purchases</i>
include_extensions	token	no	<ul style="list-style-type: none"> "always" (default) to return the extensions specified with the field "extension" if one or more customers are identified. "unique" to return the extensions identified in parameter "extensions" if a unique customer is identified.
<attribute name>	string	yes	One or more identification keys used to identify the customer, for instance: <i>last_name=Doe&first_name=Jane</i>

ID	CV.WS.CUSTID.1
	Attribute names with no prefix are supposed to be profile attributes. To refer to the attributes of an extension, your application must use the extension name as a prefix for the attribute: <ext-name>.<attribute-name>

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response	
HTTP code	200
HTTP message	OK
Body	<ul style="list-style-type: none"> • One result: Customer Profile • Several results: Customer Profile[] • No result: []

Example

Operation

```
GET /profiles/
contacts.phone_number=408-888-3214&extensions=contacts,purchases&exclude_profile=yes
&exclude_extensions=unique
```

Result: Multiple profiles

If multiple customers called from the specified phone number, then the output contains profiles for each matched customer:

```
200 OK
[{"customer_id": "2DC255C02AF", "name": "Doe, John", "birthdate": "1976-05-10"},
{"customer_id": "DDF295802AF", "name": "Doe, Jane", "birthdate": "1978-02-25"}]
```

Result: One profile

If a single customer is called from the specified phone number, then the complete profile for that customer is returned.

```
200 OK
{ "customer_id": "0004Va58A92T0017 ",
```

```
"name": "Doe, John",
"birthdate": "1976-05-10",
"EmailAddress": [
  "john.doe@genesyslab.com",
  "jd34@hotmail.dom"],
"contacts": [
  {"phone": "408-555-1234", "ext": "1234"},
  {"phone": "408-832-7712"} ],
"preferences": { "newsletter": "y", "new_offers": "n", "email": "html" }
}
```

Result: No result

If no customers match the specified criteria, then the output is an empty array:

```
[]
```

Insert Extension Records

Important

Prerequisites: You need to **enable** profiles in UCS.

POST /profiles/\${customer_id}/extensions

Description

This operation inserts one or more extension records, which replace former records if they already exist.

Operation

ID	CV.WS.PROF.4		
Method	POST		
URL	/profiles/\${customer_id}/extensions		
Parameter	Type	Mandatory	Description
URI Parameters			
\${customer_id}	string	yes	The unique ID of the given customer. Limited to 16 characters.
Body			
<extension n>	Profile Extension or Profile Extension[]	no	<p>Profile extensions, where <extension n> corresponds to the unique name of the profile extension resource.</p> <ul style="list-style-type: none"> For single-valued extensions, the extension's value is a single Profile extension. For multi-valued extensions, the extension's value is

ID	CV.WS.PROF.4			an an array of zero or more Profile Extension .
-----------	--------------	--	--	--

<references />

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response	
HTTP code	200
HTTP message	OK

Example

Operation

```
POST /profiles/0004Va58A92T0017/extensions
{
  "customer_id": "0004Va58A92T0017",
  "Phone": [
    { "PhoneType": 0,
      "prefix": "+33",
      "number": "3145926535",
      "description": "family phone",
      "start_availability": "2009-12-18T18:30:00.000Z",
      "end_availability": "2009-12-18T21:40:00.000Z"},
    { "PhoneType": 2,
      "prefix": "+33",
      "number": "6543210",
      "description": "business calls only, no sales",
      "start_availability": "2009-12-18T09:30:00.000Z",
      "end_availability": "2009-12-18T17:45:00.000Z"},
    { "PhoneType": 5,
      "prefix": "+33",
      "number": "951357456",
      "description": ""},
  ] }
}
```

Result

200 OK

UCS added the phone extension to the extensions of the specified customer.

Merge Customer Profile

PUT /profiles/{customer_id}/merge/{src_id}

Description

Merges two profiles together, transferring all relative data from one to the other (attributes and all of their extensions).

Important

This operation is not reversible and the source profile will be deleted. Starting with the 8.5.0 release, service, state, and task data are no longer handled in UCS and therefore will not be merged.

Operation

ID	CV.WS.PROF.MERGE.1		
Method	PUT		
URI	/profiles/{customer_id}/merge/{src_id}		
Parameter	Type	Mandatory	Description
URI Parameters			
{customer_id}	string	yes	The customer ID where the data is to be merged.
{src_id}	string	yes	The customer ID of the source customer data to be merged. This ID will be deleted once the merge is complete.
Body Parameters			
keep_uniq_from_src	boolean	no	<p>If there are duplicate attributes, determines which attribute to keep:</p> <ul style="list-style-type: none"> true—Keeps the attribute from the source.

ID	CV.WS.PROF.MERGE.1		
			<ul style="list-style-type: none"> • false—Keeps the attribute from the destination (default).
keep_active_service_only	boolean	no	Determines whether to merge the active records: <ul style="list-style-type: none"> • true—Merge active service records only. • false—Merge active and completed records (default).

For single valued profile extensions, and profile extension with unique attributes, the default is to keep the values from the destination profile. If `keep_uniq_from_src` is set to `true`, the source values will be kept, and the destination values will be discarded. Multi-valued extension content is copied to destination profile.

At the end of the operation, the source profile will be deleted with all completed services if the `keep_active_service_only` is set to `true`. The default operation is to transfer all active and completed services to the destination profile. Duplicate core attributes will be discarded. Core attribute duplication is determined by a case sensitive comparison of their values.

If mixed behavior is needed where some unique/single extension values must be kept from the source and others from the destination, it is recommended to update the destination profile before merging the profiles.

Important

Profile merging is transactional, either the whole operation is a success or nothing is changed. If a source profile has a lot of services assigned (hundreds) to it, the merge operation can take a while, set your client timeouts accordingly.

Example

Operation

```
/profiles/00001b8BBKDX000D/merge/00001b8BBKDX000H
Method: PUT
{
  "keep_active_service_only": false,
  "keep_uniq_from_src": false
}
```

Result

Status: Status (204) - The server has fulfilled the request but does not need to return an entity-body, and might want to return updated meta-information
Reference: <http://localhost:8222/profiles/00001b8BBKDX000D>
Transfer-Encoding: null
Content-Type: application/octet-stream

Query Customer Profile

Important

Prerequisites: You need to [enable](#) profiles in UCS.

GET /profiles/\${customer_id}

Description

Queries the Customer Profile based on the customer ID parameter. According to the query's fields, your application can retrieve profile and extension attributes in addition to the identified customer ID. The response always contains all attributes of the customer profile resource, and all attributes of the specified extensions, to avoid ambiguity in further updates (see [Update Customer Profile](#)).

Important

To get the schema of the profile and extension attributes, use the [Query Profile Schema](#) operation.

Operation

Prerequisites: A Customer Profile Schema (a list of [Attribute Schema](#)) is available.

ID	CV.WS.PROF.1		
Method	GET		
URL	/profiles/\${customer_id}		
Field Name	Type	Mandatory	Description
URI Parameters			
\${customer_id}	string	yes	Specified in the URI, the ID of the customer.
extensions	string	no	The list of extensions to return within the result. Your application can specify multiple extension names, separated with comas.

ID	CV.WS.PROF.1		For instance: <i>extensions=contacts,purchases</i>
-----------	--------------	--	---

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [\[\[HTTPResponseCodesandErrors\] HTTP Response Codes and Errors\]](#) for further details on the possible codes that this operation can return.

Response

HTTP code	200		
HTTP message	OK		
Field Name	Type	Mandatory	Description
Body: Customer Profile. The returned result contains parts of the profile or the complete profile according to the parameters of the request. See Customer Profile for further information on the resource.			
customer_id	string	yes	The unique ID of the given customer. Limited to 16 characters.
<attribute n>	<ul style="list-style-type: none"> string datetime 	yes	Customer attributes, where <attribute n> is the attribute name. See the configuration options for further details. Attributes are not part of a separated array, see the example below. The date/time formats are ISO 8601 : [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z.
<extension n>	Extension or Extension[]	no	By default, no extension is returned. The returned extensions are specified in the parameters of the request. For further details, see Extension .

<references />

Example

For example, the customer profile (id=0004Va58A92T0017) has three attributes: "name", "birthdate" and "EmailAddress". This profile also has two defined extensions:

- contacts: one or more phone numbers (a multi-valued extension).
- preferences: a single-valued extension representing customer preferences for receiving a weekly newsletter, receiving notification of new offers, and the desired email type.

Operation

```
GET /profiles/0004Va58A92T0017?extensions=contacts,preferences
```

Result

```
200 OK
{ "customer_id": "0004Va58A92T0017 ",
  "name": "Doe, John",
  "birthdate": "1976-05-10",
  "EmailAddress": [
    "john.doe@genesyslab.com",
    "jd34@hotmail.dom" ],
  "contacts": [
    {"phone": "408-555-1234", "ext": "1234"},
    {"phone": "408-832-7712"} ],
  "preferences": { "newsletter": "y", "new_offers": "n", "email": "html" }
}
```

Update Customer Profile

Important

Prerequisites: You need to [enable](#) profiles in UCS.

PUT `/profiles/${customer_id}`

Description

Overrides the attributes of the customer profile: prior attribute values are lost and replaced with the attribute list specified in the body of the request. This operation uses the standard convention for HTTP PUT, which requires you to update the profile with the complete profile to avoid losing information. Your application should use this operation in conjunction with the [Query Customer Profile](#) operation:

1. Query the Customer Profile,
2. Modify the Customer Profile data,
3. Update the Customer Profile.

Extensions can be part of the profile used for the update:

- If your application does not include any extensions in the body, no modification occurs on extensions.
- If your application specifies extension values, extension information is overridden. Your application must specify the complete extension set to avoid losing data.

Operation

Prerequisites

- The profile schema exists (see [Query Profile Schema](#)).
- Your application retrieved the Customer Profile (see [Query Customer Profile](#)).

ID	CV.WS.PROF.2		
Method	PUT		
URL	<code>/profiles/\${customer_id}</code>		
Parameter	Type	Mandatory	Description

ID	CV.WS.PROF.2		
URI Parameters			
<code>{customer_id}</code>	string	yes	The ID of the customer. Limited to 16 characters.
Body: The customer profile information <ref>The complete Customer Profile representation, compliant with the defined profile schema (see Query Profile Schema).</ref>			
customer_id	string	yes	The unique ID of the given customer.
<attribute n>	<ul style="list-style-type: none"> string datetime 	yes	<p>The complete set of customer attributes, where <attribute n> is the attribute name. See the configuration options for further details. The date/time formats are ISO 8601: [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z.</p> <p>Attributes will be overridden. This representation must contain all the profile attributes' values, including those that weren't modified.</p>
<extension n>	Profile Extension or Profile Extension[]	no	<p>Extensions, where <extension n> corresponds to the unique name of the profile extension resource.</p> <p>If you specify extensions, extensions are overridden. If you wish to modify an extension value, this representation must contain all the profile extensions, including those that weren't modified.</p>

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response	
HTTP code	200
HTTP message	OK

Example

The following example assumes that:

- FirstName, LastName, DOB are existing attributes.
- Address is an extension.

Operation

```
PUT http://ucserver.mycompany.com:8080/path/profiles/00027a52JCGY000M
Content-Location /profiles/00027a52JCGY000M
Content-Type: application/json
{
  "FirstName": "Bruce",
  "LastName": "Banner",
  "DOB": "1962-05-10",
  "EmailAddress": [
    "bruce.banner@marvelous.com",
    "b.banner@hulk.dom"
  ],
  "Address": { "Type":1, "Address":"21 JumpStreet", "City":"Hollywood",
    "County":"Santa Barbara", "PostCode":"555", "Country":"United States" }
}
```

Result

The above representation is now the profile of the customer whose id is 00027a52JCGY000M.

```
HTTP 200 OK
[]
```

Update Record In Profile Extension

Important

Prerequisites: You need to **enable** profiles in UCS.

PUT /profiles/{customer_id}/extensions/{ext_name}/by/unique

Description

Updates the content of a single record in a multi-valued extension. The attributes which are part of the 'unique' list specified at the **Extension Schema** creation are passed in the body and used to find the correct record to update.

This update cannot change the values of the attributes which are part of the 'unique' list of the **Extension Schema**; 'unique' attributes are used as identifiers for the given record. **Workaround:** If you wish to update a value for an attribute stamped as unique, you must **delete the concerned record**, then **recreate this record with its correct values**.

Operation

Prerequisites: The extension record exists.

ID	CV.WS.PROF.6		
Method	PUT		
URL	/profiles/{customer_id}/extensions/{ext_name}/by/unique		
Parameter	Type	Mandatory	Description
URI Parameters			
{customer_id}	string	yes	The unique ID of the given customer. Limited to 16 characters.
{ext_name}	string	yes	The name of the Profile Extension to update.
Body Record for the Profile Extension .			
<attribute n>	<ul style="list-style-type: none"> string integer 	yes	Individual attributes defined in the Profile Extension Schema . <attribute n> corresponds to the

ID	CV.WS.PROF.6		
	<ul style="list-style-type: none"> • long • double • date • datetime • currency • boolean 		<p>attribute name defined in the schema.</p> <ul style="list-style-type: none"> • If the attribute is unique, its value is used to select the record to update. It is not updated. • If the attribute is not unique, its value is updated.

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response	
HTTP code	204
HTTP message	No Content

Example

Operation The following operation updates the Phone record associated with the unique "PhoneNumber":"3145926535".

```
PUT /profiles/0000Sb5U97XE000Y/extensions/Phone/by/unique
{
  "PhoneType":0,
  "prefix":"+33",
  "PhoneNumber":"3145926535",
  "description":"family phone",
  "start_availability":"2009-12-18T18:30:00.000Z",
  "end_availability":"2009-12-18T21:40:00.000Z"
}
```

Result

204 No Content with empty body

Profile Resources

Important

Prerequisites: You need to **enable** profiles in UCS.

Purpose: Groups profile resources, which include metadata to manage attributes and extensions.

- [Customer Profile](#)
- [Profile Attribute Schema](#)
- [Profile Extension](#)

Customer Profile

Description

Describes a customer profile. Customer profiles are built on top of legacy UCS Contact Attributes. They are multi-valued and support the primary attributes. If an attribute has several values, the first one of the JSON array is the primary attribute, the others are non-primary. Each customer profile consists in a list of attributes defined via business attributes, see [the configuration options](#) for further details. These attributes share the same level with the customer_id field (provided by the UCS to identify the customer).

In addition to these attributes, your application can add extensions (whose schema is defined with [Schema Operations](#)). See the table and example below.

Resource

Field	Type	Mandatory	Description
customer_id	string	yes	The unique ID of the given customer. Limited to 16 characters.
<attribute n>	<ul style="list-style-type: none"> string datetime 	yes	<p>Customer attributes, where <attribute n> is the attribute name<ref>See the configuration options for further details.</ref>. Attributes are not part of a separated array, see the example below.</p> <p>The date/time formats are ISO 8601<ref>Wikipedia ISO 8601</ref>: [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z.</p>
<extension n>	Profile extension or Profile extension[]	no	<p>Profile extensions, where <extension n> corresponds to the unique name of the profile extension resource.</p> <ul style="list-style-type: none"> For single-valued extensions, the result is a single Profile Extension.

Field	Type	Mandatory	Description
			<ul style="list-style-type: none">For multi-valued extensions, the field value is an an array of zero or more Profile extension.

<references />

Example

The following output sample presents the representation for the customer *John Doe*. It includes 2 attributes : `FirstName` and `LastName`. The `EmailAddress` is an extension which contains two contact records.

```
{
  "FirstName": "John",
  "LastName": "Doe",
  "EmailAddress": [
    "john.doe@genesyslab.com",
    "jd34@hotmail.dom"
  ]
}
```

Profile Attribute Schema

Important

Prerequisites: You need to **enable** profiles in UCS.

Purpose: Describes a Profile Attribute schema.

Description

Describes a Profile Attribute. Each attribute field of the profile corresponds to a recorded schema which details the attribute name, type, and content. Select in "Tenant" the item *specific configuration*, and under "Business Attributes" 'Contact Attributes'. Then, create a new Business Attribute Value. By default the new Attribute is of type string. To assign another type to the schema, in Annex, create a section "settings", and a key "type" with value "string" or "date".

Predefined Attributes Schema

Default Attributes visible in the Configuration Server are the following:

- First Name (string)
- Last Name (string)
- Phone Number (string)
- E-mail Address (string)
- Title (string)

Default 'internal' Attributes (not visible in the Configuration Server) are the following:

- LastCalledAgent_EmployeeID (string)
- LastCalledAgent_TimeStamp (date)
- PreferredAgent_EmployeeID (string)

Additional attributes are available for each active media defined in the Business Attribute "Media Type" of the Configuration Manager:

- LCA_EmplID_<media> (string)
- LCA_TimeStamp_<media> (date)
- Pref_EmplID_<media> (string)

Attributes of type date/time are formatted in UTC ISO 8601: "yyyy-MM-dd'T'HH:mm:ss.SSS'Z'"
 Example: "1981-03-17T02:00:00.000Z"

Resource

Field	Type	Mandatory	Description
name	string	yes	<p>The name is unique for each attribute within the given profile.</p> <ul style="list-style-type: none"> Starts with a letter, and can be followed with letters, numbers, or underscores. Is restricted to a maximum of 30 characters (Maximum RDBMS shared limit) <div style="border: 1px solid orange; padding: 5px; margin-top: 10px;"> <p>Important In profile attribute schemas, the name is not case-sensitive.</p> </div>
type	token	yes	<p>The attribute type, which can be one of the following types:</p> <ul style="list-style-type: none"> Boolean String Integer Long Double Date DateTime Currency <p>Notes:</p> <ol style="list-style-type: none"> The constraints on these types (the minimum and maximum values supported for the integer type) vary with the RDBMS.

Field	Type	Mandatory	Description
			<p>2. Customer profiles are built on top of UCS' Contact Attributes, and support only String and DateTime attribute types. However, all attribute types are supported for profile extensions.</p>
default	numeric or string	no	<p>Provides a default value for the attribute. For numeric attributes, you do not need quotation marks. Quotation marks are for string, date, and datetime attributes. By default, the date/time format is ISO 8601 UTC: [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z.</p>
length	integer	no	<p>Reserved to attributes of type String. Defines the maximum number of characters for the string.</p> <ul style="list-style-type: none"> • Default length is 256; • Maximum length is 4000.
encrypt	Boolean	no	<ul style="list-style-type: none"> • false by default. • true if the server must encrypt the attribute value before saving it to the UCS database.
mandatory	Boolean	no	<ul style="list-style-type: none"> • false by default; • true if the attribute is mandatory when inserting a new Profile Extension record.

Related Operation

- [Query Profile Schema](#)

Profile Extension

Important

Prerequisites: You need to [enable](#) profiles in UCS.

Purpose: Describes an extension.

Description


Each extension fulfills the contract of an [Extension Schema](#) that your application created in anterior operations. Your application can create schemas for the following extensions:

- Profile extensions, created through the [Create Profile Extension Schema](#) operation, which provides extensions for profiles.

Once the extension schema is defined, your application can create and add extension records when dealing with standard profile, services, states, and task operations. For instance, if [you start a task](#), you can specify extension records in your operation body as shown in the example section.

Profile Extension

Profile Extension

Field	Type	Mandatory	Description
customer_id	string	yes if not nested in a profile	The unique ID of the given customer.  This field is mandatory if the extension is not nested inside a Customer Profile .
name	string	yes	The unique, case-insensitive name of the extension. <ul style="list-style-type: none">• Starts with a letter to which you can append letters, numbers, or underscores.

Field	Type	Mandatory	Description
			<ul style="list-style-type: none"> Restricted to a maximum of 26 characters. (Maximum RDBMS shared limit on index creation.)
<attribute n>	<ul style="list-style-type: none"> string integer long double date datetime currency boolean 	yes	Individual attributes, defined in the Extension Schema . <attribute n> corresponds to the attribute name defined in the schema.

<references />

Schema Operations

Important

Prerequisites: You need to **enable** profiles in UCS.

Purpose: Groups operations which handle schemas for both identification keys and profile extensions.

- Schema Operations
 - [GET /metadata/profiles/](#)
 - [POST /metadata/profiles/extensions](#)
 - [GET /metadata/profiles/extensions](#)
 - [GET /metadata/identification-keys](#)
 - [POST /metadata/identification-keys](#)
 - [GET \\${content-type} /metadata](#)
 - [DELETE /metadata/profiles/extensions/\\${extension-name}](#)
 - [DELETE /metadata/identification-keys/\\${id_key-name}](#)

Create Identification Key

POST /metadata/identification-keys

Important

Prerequisites: You need to **enable** profiles in UCS.

Description

Creates the key used to identify customers. Your application must define the **Identification Key** prior to any customer identification attempt. You can build identification keys as a combination of attributes:

- One or more attributes of the customer profile.
- One or more attributes of an extension.

Important

Your application cannot create Identification keys with a blend of profile attributes and extension attributes.

The following identification key representation identifies customers based on the `PhoneNumber` attribute.

```
{ "name": "idPhone",
  "attributes": ["PhoneNumber "]}
```

Let's imagine that the phone number is part of the **contactExt** extension; the corresponding object is:

```
{ "name": "idPhone",
  "source": "contactExt",
  "attributes": ["PhoneNumber "]}
```

This example assumes that one or more customers can share the same phone number, which could apply to members of a family for instance, and makes the identification key not unique across the calling customers. To make it unique, use the "unique" field of the identification key (see the **Identification Key** for further details).

Operation

Prerequisites: The server is in **Maintenance mode**.

 Use the **Set Server Mode** operation to switch modes.

Create Identification Key

ID	CV.WS.PROF.META.7		
Method	POST		
URL	/metadata/identification-keys		
Parameter	Type	Mandatory	Description
Body: Includes the identification key to create. This body contains the Identification Key resource.			
name	string	yes	The unique, case-insensitive name of the key. The name must start with a letter, and can be followed with letters, numbers, or underscores. The name is restricted to a maximum of 26 characters. (Maximum RDBMS shared limit on creation of index.)
source	string	no	Can take on one of the following values: <ul style="list-style-type: none"> "profile" (default value): indicates that the attributes specified in parameter "attribute" belong to the core customer profile. The unique name of the given Profile Extension, used to create the Identification Key.
attributes	string[]	yes	The array names of one or more attributes of the extension or of the profile.
unique	bool	no	<ul style="list-style-type: none"> "false" by default. "true" to indicate that the given

ID	CV.WS.PROF.META.7		
			attributes of the profile or extension are unique across all customers.

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response	
HTTP code	201
HTTP message	Created
Header	Location: metadata/identification-keys/ \${key-name} where: <ul style="list-style-type: none"> \${key-name} is the name specified in the submitted identification key resource.
Body	{"name": " \${key-name} "} where: <ul style="list-style-type: none"> \${key-name} is the name of the created key.

Example

The following sample creates a key which identifies the customer based on the provided phone ID. After this key is created, the further profile creations are indexed based on the Customer Phone ID.

Operation

```
POST /metadata/identification-keys
{
  "name": "idPhone",
  "extension": "contactExt",
  "attributes": ["PhoneNumber"]
}
```

Result

```
HTTP 201 Created
{"name": "idPhone"}
```

Create Profile Extension Schema

Important

Prerequisites: You need to [enable](#) profiles in UCS.

POST /metadata/profiles/extensions

Description

Creates a profile extension schema resource, which is mandatory to query and use profile extension schemas.

- [Read more about Extensions.](#)

Operation

Create Profile Extensions Schema

ID	CV.WS.PROF.META.4		
Method	POST		
URL	/metadata/profiles/extensions		
Parameter	Type	Mandatory	Description
Body: The schema to create.<ref>This is the Profile Extension Schema to create.</ref>			
name	string	yes	The unique, case-insensitive name of the extension. The name must start with a letter, and can be followed with letters, numbers, or underscores. The name is restricted to a maximum of 26 characters (Maximum RDBMS shared limit on creation of index).
type	token <ul style="list-style-type: none"> • "single-valued" • "multi-valued" 	yes	Extensions come in the following forms: <ul style="list-style-type: none"> • Single-valued: a single extension

ID	CV.WS.PROF.META.4		
			record is associated with a given customer. <ul style="list-style-type: none"> Multi-valued: multiple extension records can be associated with a given customer.
attributes	Attribute Schema[]	no	The array of <i>zero or more</i> attributes. Empty when type="classifier".
unique	string[]	no	Lists attributes which are unique in the scope of a given customer. The customer profile cannot include several extension records with identical values for these attributes. For example, let's consider a contact extension with the attributes "phone_num", "ext", "time_of_day". To ensure that a given customer does not have two contact extensions with the same phone number, set the following in the extension schema: unique = ["phone_num", "ext"]

<references />

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response	
HTTP code	201
HTTP message	Created
Header	Location: `\${base_uri}` /profiles/

HTTP code	201
	<p>extensions/\${extension-name}</p> <p>where</p> <ul style="list-style-type: none"> • \${extension-name} is the name of the extension provided in the request's entity body. • \${base_uri} is the base URI of Context Services.
Body	<p>{ "name": "\${extension-name}" }</p> <p>where:</p> <ul style="list-style-type: none"> • \${extension-name} is the name of the created extension.

Example

Operation

POST <http://ucsserver.mycompany.com:8080/cms/profiles/extensions>

```
{
  "name": "Address",
  "type": "single-valued",
  "attributes": [
    {"name": "AddressType", "type": "integer", "default": 0},
    {"name": "Address", "type": "string", "length": 256},
    {"name": "City", "type": "string", "length": 32},
    {"name": "County", "type": "string", "length": 32},
    {"name": "PostCode", "type": "string", "length": 10},
    {"name": "Country", "type": "string", "length": 32}
  ]
}
```

Result

201 Created
 Content-Location <http://ucsserver.mycompany.com:8080/cms/profiles/extensions/Address>
 {"name": "Address"}

Change Metadata Cache

Important

Prerequisites: You need to **enable** profiles in UCS.

PUT /metadata/cache/

Description

Enables the reload of metadata cache. Metadata cache holds information for the definition of identification keys.

Operation

ID	CV.WS.META.CACHE.PUT		
URI	/metadata/cache		
Method	PUT		
Parameter	Type	Mandatory	Description
reload	token	yes	Any value triggers a reload.

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response

HTTP code	200
HTTP message	No Content

Example

The following operation reloads the metadata cache from the database.

Operation

```
PUT /cv/metadata/cache/  
{reload:true}
```

Result

HTTP/1.1 200 OK with an empty body.

<references />

Delete Metadata Identification Keys

Important

Prerequisites: You need to **enable** profiles in UCS.

`/metadata/identification-keys/${id_key-name}`

Description

Allows the deletion of Identification keys. The records in metadata table are removed, and the related table is dropped (if any). An identification key can be created based on the core profile attribute, but in this case a table is not created. Therefore, no table is dropped when it is deleted.

Operation

ID	CV.WS.PROF.META.8		
URI	/metadata/identification-keys/\${id_key-name}		
Method	DELETE		
Parameter	Type	Mandatory	Description
URI Parameter			
\${id_key-name}	string	yes	The metadata identification key name to delete. This name is not case-sensitive.

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response	
HTTP code	204
HTTP message	Success No Content

Example

The following operation deletes an identification key.

Operation

```
DELETE /metadata/identification-keys/idNickName
```

Result

```
HTTP/1.1 204 OK  
Success No Content
```

Important

If nothing was deleted, HTTP status code 404 ("Not Found") is sent.

Delete Metadata Profile Extensions

Important

Prerequisites: You need to **enable** profiles in UCS.

`/metadata/profiles/extensions/${extension-name}`

Description

Allows the deletion of a profile extension. The records in the profile extension table are deleted, the table is deleted, and the definition of the profile extension is erased. If identification keys exist based on the specified extension, it will not be deleted, and an error will be returned.

Operation

ID	CV.WS.PROF.META.5		
URI	/metadata/profiles/extensions/\${extension-name}		
Method	DELETE		
Parameter	Type	Mandatory	Description
URI Parameter			
\${extension-name}	string	yes	The metadata extension name to delete. Important This name is not case-sensitive.

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response	
HTTP code	204
HTTP message	Success No Content

Example

The following operation deletes a profile extension.

Operation

```
DELETE /metadata/profiles/extensions/CustomName
```

Result

```
HTTP/1.1 204 OK  
Success No Content
```

Important

If nothing was deleted, HTTP status code 404 ("Not Found") is sent.

Get Identification Keys

Important

Prerequisites: You need to **enable** profiles in UCS.

GET /metadata/identification-keys

Description

Returns the identification keys.

Operation

Get Identification Keys

ID	CV.WS.PROF.META.6		
Method	GET		
URL	<ul style="list-style-type: none"> All: /metadata/identification-keys Key: /metadata/identification-keys/{key-name} 		
Name	Type	Mandatory	Description
URI Parameter			
{key-name}	string	no	<p>The unique, case-insensitive name of the key. The name must start with a letter, and can be followed with letters, numbers, or underscores. The name is restricted to a maximum of 26 characters. (Maximum RDBMS shared limit on creation of index.)</p> <p>Important If you do not specify this parameter, all keys are returned.</p>

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response	
HTTP code	200
HTTP message	OK
Body	<ul style="list-style-type: none"> • /metadata/identification-keys: Identification Key[] <ref>Length can be 0.</ref> • /metadata/identification-keys/\${key-name}: Identification Key

<references />

Example

The following request retrieves all the identification keys: **Operation**

```
GET /metadata/identification-keys
```

Result

Let's consider a customer profile which consists of the customer's name, the date of birth, and the external ID (the customer's ID in an external system such as a CRM). Further assume that a single extension, Contacts, include various attributes such as the phone_number. If you set up the Context Management Server to allow the customer identification via name+birthdate, external ID, or phone number, then the returned result is the following:

```
[{"name": "idNameBD", "attributes": ["name", "birthdate"]},
{"name": "idExt", "attributes": ["external_id", "unique": true},
{"name": "idPhone", "attributes": ["phone_number"]}
{"name": "idExtension", "attributes": ["ext.code", "ext.date"]}]
```

Important

If the identification key was created off an extension, attributes that are part of the extension are prefixed with the extension name. In the previous example, "idExtension" was built on the extension "ext".

Get Metadata

Important

Prerequisites: You need to **enable** profiles in UCS.

GET `${content-type} /metadata`

Description

Returns the location for the list of available metadata resources.

Operation

Get Metadata

ID	CV.WS.META.1		
Method	GET		
URL	/metadata		
HEADER	<code>\${content-type}</code>		
Parameter	Type	Mandatory	Description
<code>\${content-type}</code>	<ul style="list-style-type: none"> "Content-type:text/html" "Content-type:application/json" 	yes	The formatting for the returned result.

<references />

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response	
HTTP code	200
HTTP message	OK
Body	GET "Content-type:text/html" /metadata
	Returns an HTML page, containing a list of resource links as follows: <pre><html><body><h1>Available Resources</h1>
 \${resource-name}
 </body></html></pre>
	where:
	<ul style="list-style-type: none"> • <code>\${resource}</code> is the location of the resource. • <code>\${resource-name}</code>, the name of the resource.
Body	GET "Content-type:application/json" /metadata
	Returns a JSON table, containing a list of resource links as follows: <pre>{\${resource-name-1}:\${resource-1}, \${resource-name-2}:\${resource-2}, ..., \${resource-name-n}:\${resource-n} }</pre>
	where:
	<ul style="list-style-type: none"> • <code>\${resource_i}</code> is the location of the resource-i. • <code>\${resource-name_i}</code> is the name of the resource-i.

Example

Operation

```
GET "Content-type:text/html" /metadata
```

Result

```
200 OK {
<html><body><h1>Available Resources</h1><br/>
<a href="http://localhost:8080/metadata/identification-keys">
identification-keys schemas<br/>
<a href="http://localhost:8080/metadata/tasks/extensions">Tasks Extensions<br/>
<a href="http://localhost:8080/metadata/states/extensions">States Extensions<br/>
<a href="http://localhost:8080/metadata/services/extensions">Services Extensions<br/>
<a href="http://localhost:8080/metadata/business-attributes">business-attributes mapping<br/>
</body></html>
}
```

Operation

GET "Content-type:application/json" /metadata

Result

```
200 OK
{"identification-keys schemas":"http://localhost:8080/metadata/identification-keys",
  "Tasks Extensions":"http://localhost:8080/metadata/tasks/extensions",
  "Services Extensions":"http://localhost:8080/metadata/services/extensions",
  "States Extensions":"http://localhost:8080/metadata/states/extensions",
  "business-attributes mapping":"http://localhost:8080/metadata/business-attributes"}
```

Query Metadata Roles

Important

Prerequisites: You need to **enable** profiles in UCS.

GET /metadata/roles/

Description

This method provides convenient access to roles as configured in Configuration Server.

Operation

-

ID	CV.WS.SRV.META.x
URI	/metadata/roles
Method	GET

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response

HTTP code	200	
HTTP message	OK	
Name	Type	Description
role_name	String	The name of the given role.
role_dbid	String	The unique ID of the given role.
role_enabled	Boolean	<ul style="list-style-type: none"> enabled disabled

HTTP code	200	
task	JSON Array	<p>A task is made up of the following:</p> <ul style="list-style-type: none"> • Name--The name of the task. • Capability--Indicates whether the role can perform the task.

<references />

Example

The following operation returns the configured roles with the state and capability.

Operation

GET /metadata/roles

Result

On success, the operation returns an array of zero or more JSON objects.

```
[ {
  "role_DBID" : 108,
  "role_enabled" : true,
  "tasks" : [ {
    "capableOf" : true,
    "name" : "UCS.Customer.createProfile"
  } ],
  "role_name" : "Ucs_BigBrother"
}, {
  "role_DBID" : 110,
  "role_enabled" : true,
  "tasks" : [ {
    "capableOf" : true,
    "name" : "UCS.SchemaMgt.readServiceExtensionSchema"
  }, {
    "capableOf" : true,
    "name" : "UCS.Service.readServiceExtension"
  }, {
    "capableOf" : true,
    "name" : "UCS.Service.createServiceExtension"
  }, {
    "capableOf" : true,
    "name" : "UCS.Service.deleteServiceExtension"
  }, {
    "capableOf" : true,
    "name" : "UCS.SchemaMgt.readTaskExtensionSchema"
  }, {
    "capableOf" : true,
    "name" : "UCS.Service.startService"
  }, {
```

```
    "capableOf" : true,
    "name" : "UCS.SystemMgt.readServerInfo"
  }, {
    "capableOf" : true,
    "name" : "UCS.Service.readService"
  }, {
    "capableOf" : true,
    "name" : "UCS.SystemMgt.changeServerMode"
  } ],
  "role_name" : "ucs_roleForAgents"
}, {
  "role_DBID" : 111,
  "role_enabled" : true,
  "tasks" : [ {
    "capableOf" : true,
    "name" : "UCS.SchemaMgt.handleMetadata"
  }, {
    "capableOf" : true,
    "name" : "UCS.SystemMgt.readServerInfo"
  }, {
    "capableOf" : true,
    "name" : "UCS.SchemaMgt.readGenAdminRole"
  } ],
  "role_name" : "uselessRole"
}, {
  "role_DBID" : 112,
  "role_enabled" : true,
  "tasks" : [ {
    "capableOf" : true,
    "name" : "UCS.SystemMgt.readServerInfo"
  } ],
  "role_name" : "anotherUselessRole"
} ]
```

<references />

Query Profile Extension Schema

Important

Prerequisites: You need to **enable** profiles in UCS.

GET /metadata/profiles/extensions

Description

Queries the schemas of the given profile extension.

Operation

Prerequisites: The Profile Extension Schema resource exists.

Important

To create the profile extension schema, use the **Create Profile Extension Schema** operation.

Query Profile Extensions Schema

ID	CV.WS.PROF.META.3		
Method	GET		
URL	<ul style="list-style-type: none"> All: /metadata/profiles/extensions /metadata/profiles/extensions/\${extension-name} 		
Name	Type	Mandatory	Description
URI Parameter			
\${extension-name}	string	no	The extension name, if a specific extension should be returned. <div style="border: 1px solid orange; padding: 5px; margin-top: 5px;"> Important This name is not case- </div>

ID	CV.WS.PROF.META.3
	sensitive.

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response	
HTTP code	200
HTTP message	OK
Body	/metadata/profiles/extensions
	Returns an array of the following fields: <ul style="list-style-type: none"> name: the unique extension name. type: "single-valued" or "multi-valued". required: true or false attributes: Attribute Schema[].
	/metadata/profiles/extensions/\${extension-name} Extension Schema or Extension Schema[]

Example

Get all extension schemas

Operation

```
GET /metadata/profiles/extensions
```

Result

```
200 OK
[
{
  "name": "Phone",
  "type": "multi-valued",
  "attributes": [
    {"name": "PhoneType", "type": "integer", "default": 0, "mandatory": "true"},
    {"name": "prefix", "type": "string", "length": 3, "default": "555",},
    {"name": "PhoneNumber", "type": "integer", "length": 15, "mandatory": "true"},
    {"name": "description", "type": "string", "length": 32, "mandatory": "true"},
    {"name": "start_availability", "type": "datetime"},
  ]
}
```



```
        {"name":"end_availability","type":"datetime", "mandatory":"false"}
    ],
},
{
  "name":"Address",
  "type":"single-valued",
  "attributes": [
    {"name":"AddressType","type":"integer","default":0},
    {"name":"Address","type":"string","length":256},
    {"name":"City","type":"string","length":32},
    {"name":"County","type":"string","length":32},
    {"name":"PostCode","type":"string", "length":10},
    {"name":"Country","type":"string","length":32}
  ]
} ]
```

Get a single extension schema

Operation

GET /metadata/profiles/extensions/Phone

Result

```
200 OK
{
  "name":"Phone",
  "type":"multi-valued",
  "attributes": [
    {"name":"PhoneType","type":"integer","default":0,"mandatory":"true"},
    {"name":"prefix","type":"string","length":3,"default":"555",},
    {"name":"PhoneNumber","type":"integer","length":15,"mandatory":"true"},
    {"name":"description","type":"string","length":32,"mandatory":"true"},
    {"name":"start_availability","type":"datetime"},
    {"name":"end_availability","type":"datetime", "mandatory":"false"}
  ]
}
```

Query Profile Schema

Important

Prerequisites: You need to [enable](#) profiles in UCS.

GET /metadata/profiles/

Description

Queries the schema which describes the customer profiles. If the profile schema is not defined, this operation returns the HTTP Status Code 404 (Not Found). See [Configuration Options](#) for further details.

Operation

Prerequisites:

- The profile schema is already defined.

Query Profile Schema

ID	CV.WS.PROF.META.1
Method	GET
URL	/metadata/profiles/

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response

HTTP code	200
HTTP message	OK
Body	Attribute Schema[]

Example

Operation

GET /metadata/profiles/

Result

200 OK

```
[{"encrypt":false,"name":"PIN","length":256,"type":"string"},
{"encrypt":false,"name":"Title","length":256,"type":"string"},
{"encrypt":false,"name":"CustomerSegment","length":256,"type":"string"},
{"encrypt":false,"name":"LastName","length":256,"type":"string"},
{"encrypt":false,"name":"AccountNumber","length":256,"type":"string"},
{"encrypt":false,"name":"FirstName","length":256,"type":"string"},
{"encrypt":false,"name":"PhoneNumber","length":256,"type":"string"},
{"encrypt":false,"name":"ContactId","length":256,"type":"string"},
{"encrypt":false,"name":"EmailAddress","length":256,"type":"string"},
{"encrypt":false,"name":"Title","length":256,"type":"string"},
{"encrypt":false,"name":"PIN","length":256,"type":"string"},
{"encrypt":false,"name":"ContactId","length":256,"type":"string"},
{"encrypt":false,"name":"homeaddress","length":256,"type":"string"},
{"encrypt":false,"name":"EmailAddress","length":256,"type":"string"},
{"encrypt":false,"name":"PhoneNumber","length":256,"type":"string"},
{"encrypt":false,"name":"AccountNumber","length":256,"type":"string"},
{"encrypt":false,"name":"LastName","length":256,"type":"string"},
{"encrypt":false,"name":"FirstName","length":256,"type":"string"},
{"encrypt":false,"name":"CustomerSegment","length":256,"type":"string"},
{"encrypt":false,"name":"LastCalledAgent_EmployeeID","length":256,"type":"string"},
{"encrypt":false,"name":"LastCalledAgent_TimeStamp","length":0,"type":"datetime"},
{"encrypt":false,"name":"PreferredAgent_EmployeeID","length":256,"type":"string"},
{"encrypt":false,"name":"LCA_EmplID_workitem","length":256,"type":"string"},
{"encrypt":false,"name":"LCA_TimeStamp_workitem","length":0,"type":"datetime"},
{"encrypt":false,"name":"Pref_EmplID_workitem","length":256,"type":"string"},
{"encrypt":false,"name":"LCA_EmplID_auxwork","length":256,"type":"string"},
{"encrypt":false,"name":"LCA_TimeStamp_auxwork","length":0,"type":"datetime"},
{"encrypt":false,"name":"Pref_EmplID_auxwork","length":256,"type":"string"},
{"encrypt":false,"name":"LCA_EmplID_imchat","length":256,"type":"string"},
{"encrypt":false,"name":"LCA_TimeStamp_imchat","length":0,"type":"datetime"},
{"encrypt":false,"name":"Pref_EmplID_imchat","length":256,"type":"string"},
{"encrypt":false,"name":"LCA_EmplID_outboundpreview","length":256,"type":"string"},
{"encrypt":false,"name":"LCA_TimeStamp_outboundpreview","length":0,"type":"datetime"},
{"encrypt":false,"name":"Pref_EmplID_outboundpreview","length":256,"type":"string"},
{"encrypt":false,"name":"LCA_EmplID_vmail","length":256,"type":"string"},
{"encrypt":false,"name":"LCA_TimeStamp_vmail","length":0,"type":"datetime"},
{"encrypt":false,"name":"Pref_EmplID_vmail","length":256,"type":"string"},
{"encrypt":false,"name":"LCA_EmplID_smail","length":256,"type":"string"},
{"encrypt":false,"name":"LCA_TimeStamp_smail","length":0,"type":"datetime"},
{"encrypt":false,"name":"Pref_EmplID_smail","length":256,"type":"string"},
{"encrypt":false,"name":"LCA_EmplID_cobrowsing","length":256,"type":"string"},
{"encrypt":false,"name":"LCA_TimeStamp_cobrowsing","length":0,"type":"datetime"},
{"encrypt":false,"name":"Pref_EmplID_cobrowsing","length":256,"type":"string"},
{"encrypt":false,"name":"LCA_EmplID_webform","length":256,"type":"string"},
{"encrypt":false,"name":"LCA_TimeStamp_webform","length":0,"type":"datetime"},
{"encrypt":false,"name":"Pref_EmplID_webform","length":256,"type":"string"},
{"encrypt":false,"name":"LCA_EmplID_busevent","length":256,"type":"string"},
{"encrypt":false,"name":"LCA_TimeStamp_busevent","length":0,"type":"datetime"},
{"encrypt":false,"name":"Pref_EmplID_busevent","length":256,"type":"string"},
{"encrypt":false,"name":"LCA_EmplID_voice","length":256,"type":"string"},
{"encrypt":false,"name":"LCA_TimeStamp_voice","length":0,"type":"datetime"},
{"encrypt":false,"name":"Pref_EmplID_voice","length":256,"type":"string"}]
```

```
{
  "encrypt": false, "name": "LCA_EmplID_appsharing", "length": 256, "type": "string"},
  {"encrypt": false, "name": "LCA_TimeStamp_appsharing", "length": 0, "type": "datetime"},
  {"encrypt": false, "name": "Pref_EmplID_appsharing", "length": 256, "type": "string"},
  {"encrypt": false, "name": "LCA_EmplID_email", "length": 256, "type": "string"},
  {"encrypt": false, "name": "LCA_TimeStamp_email", "length": 0, "type": "datetime"},
  {"encrypt": false, "name": "Pref_EmplID_email", "length": 256, "type": "string"},
  {"encrypt": false, "name": "LCA_EmplID_chat", "length": 256, "type": "string"},
  {"encrypt": false, "name": "LCA_TimeStamp_chat", "length": 0, "type": "datetime"},
  {"encrypt": false, "name": "Pref_EmplID_chat", "length": 256, "type": "string"},
  {"encrypt": false, "name": "LCA_EmplID_any", "length": 256, "type": "string"},
  {"encrypt": false, "name": "LCA_TimeStamp_any", "length": 0, "type": "datetime"},
  {"encrypt": false, "name": "Pref_EmplID_any", "length": 256, "type": "string"},
  {"encrypt": false, "name": "LCA_EmplID_fax", "length": 256, "type": "string"},
  {"encrypt": false, "name": "LCA_TimeStamp_fax", "length": 0, "type": "datetime"},
  {"encrypt": false, "name": "Pref_EmplID_fax", "length": 256, "type": "string"},
  {"encrypt": false, "name": "LCA_EmplID_video", "length": 256, "type": "string"},
  {"encrypt": false, "name": "LCA_TimeStamp_video", "length": 0, "type": "datetime"},
  {"encrypt": false, "name": "Pref_EmplID_video", "length": 256, "type": "string"},
  {"encrypt": false, "name": "LCA_EmplID_whiteboard", "length": 256, "type": "string"},
  {"encrypt": false, "name": "LCA_TimeStamp_whiteboard", "length": 0, "type": "datetime"},
  {"encrypt": false, "name": "Pref_EmplID_whiteboard", "length": 256, "type": "string"},
  {"encrypt": false, "name": "LCA_EmplID_alert", "length": 256, "type": "string"},
  {"encrypt": false, "name": "LCA_TimeStamp_alert", "length": 0, "type": "datetime"},
  {"encrypt": false, "name": "Pref_EmplID_alert", "length": 256, "type": "string"},
  {"encrypt": false, "name": "LCA_EmplID_voip", "length": 256, "type": "string"},
  {"encrypt": false, "name": "LCA_TimeStamp_voip", "length": 0, "type": "datetime"},
  {"encrypt": false, "name": "Pref_EmplID_voip", "length": 256, "type": "string"},
  {"encrypt": false, "name": "LCA_EmplID_sms", "length": 256, "type": "string"},
  {"encrypt": false, "name": "LCA_TimeStamp_sms", "length": 0, "type": "datetime"},
  {"encrypt": false, "name": "Pref_EmplID_sms", "length": 256, "type": "string"},
  {"encrypt": false, "name": "LCA_EmplID_callback", "length": 256, "type": "string"},
  {"encrypt": false, "name": "LCA_TimeStamp_callback", "length": 0, "type": "datetime"},
  {"encrypt": false, "name": "Pref_EmplID_callback", "length": 256, "type": "string"}
}
```

Schema Resources

Important

Prerequisites: You need to **enable** profiles in UCS.

Lists the schema resources available for profile management.

- [Attribute Schema](#)
- [Identification Key](#)
- [Profile Extension Schema](#)

Attribute Schema

Important

Prerequisites: You need to **enable** profiles in UCS.

Purpose: Describes an attribute schema, which can be part of an Extension Schema or of a Profile Schema.

Description

Your application can retrieve or manage the definitions of the attributes that compose some resources of the Context Services. Each attribute schema defines an attribute, its name, its type, and associated properties such as the default value of the attribute (if it exists). For instance, the following attribute schema defines that the Address Type attribute accepts integer values and is equal to 0 by default:

```
{"name": "AddressType", "type": "integer", "default": 0}
```

Profile Attributes

Profile attributes correspond to a recorded schema which details the attribute name, type, and content.

Select in "Tenant" the item *specific configuration*, and under "Business Attributes" the item 'Contact Attributes'. Then, create a new Business Attribute Value. By default, the new Attribute is of type string. To assign another type to the schema, in Annex, create a section "settings", and a key "type" with the value "string" or "date". The default predefined attributes visible in the Configuration Server are the following:

- First Name (string)
- Last Name (string)
- Phone Number (string)
- E-mail Address (string)
- Title (string)

Default 'internal' Attributes (not visible in the Configuration Server) are the following:

- LastCalledAgent_EmployeeID (string)
- LastCalledAgent_TimeStamp (date)
- PreferredAgent_EmployeeID (string)

Additional attributes are available for each active media defined in the Business Attribute "Media Type" of the Configuration Manager:

- LCA_EmplID_<media> (string)
- LCA_TimeStamp_<media> (date)
- Pref_EmplID_<media> (string)

Attributes of type date/time are formatted in **ISO 8601** UTC: [YYYY]-[MM]-[DD]T[HH]:[mm]:[ss].[SSS]Z
 Example: "1981-03-17T02:00:00.000Z"

Service, State, and Task extension attributes

Your application can define service, states, and tasks extension resources, based on extension schema which use an array of Attribute Schemas to define the content of the extension. See [Extension Schema](#) for further details.

Resource

Attribute Schema

Field	Type	Mandatory	Description
name	string	yes	The name is unique for each attribute within the given extension. <ul style="list-style-type: none"> • Starts with a letter, and can be followed with letters, numbers, or underscores. • Is restricted to a maximum of 30 characters (Maximum RDBMS shared limit.) <div style="border: 1px solid orange; padding: 5px; margin-top: 10px;"> Important In profile attribute schemas, the name is not case-sensitive. </div>
type	token <ul style="list-style-type: none"> • boolean • string • integer • long 	yes	1. The constraints on these types (the minimum and maximum values supported for the integer type) vary with the RDBMS.

Field	Type	Mandatory	Description
	<ul style="list-style-type: none"> double date dateTime currency 		<p>2. Customer profiles are built on top of UCS' Contact Attributes, and support only String and DateTime attribute types. However, all attribute types are supported for profile extensions.</p>
default	numeric or string	no	<p>Provides a default value for the attribute. For numeric attributes, you do not need quotation marks. Quotation marks are for string, date, and datetime attributes. By default, the date/time format is ISO 8601 UTC: [YYYY] - [MM] - [DD]T[HH]:[mm]:[ss].[SSS]Z.</p>
length	integer	no	<p>Reserved to attributes of type String. Defines the maximum number of characters for the string.</p> <ul style="list-style-type: none"> Default length is 256 Maximum length is 4000.
mandatory	boolean	no	<ul style="list-style-type: none"> <i>false</i> by default; <i>true</i> if the attribute is mandatory when inserting a new Profile Extension record.

Identification Key

Purpose

Describes the Identification Key Resource.

Description

The Identification Key is a combination of attributes used to identify a customer. These attributes (one or more) belong to the Customer Profile or to its extensions. Your application is responsible for creating the identification keys that are alter use to identify customers. See [Create Identification Key](#) for further details.

Important

Identification keys cannot be issued from both the profile and its extensions.

Some identification key samples:

- An identification key consisting of attributes "name" and "birthdate" in the Profile.
- A key based on attribute "pin" of a single-valued extension used to hold customer pass codes.
- A key based on attribute "number" of a multi-valued extension used to record the phone numbers we have seen a customer call from.

When the UCS receives a new customer profile and the associated extension data, it builds the indexing structures based on the specified identification keys, in order to ensure efficient customer identification.

Resource

Identification Key

Field	Type	Mandatory	Description
name	string	yes	The unique, case-insensitive name of the key. The name must start with a letter, and can be followed with letters, numbers, or

Field	Type	Mandatory	Description
			underscores. The name is restricted to a maximum of 26 characters. (Maximum RDBMS shared limit on creation of index.)
source	string	no	<p>Can take on one of the following values:</p> <ul style="list-style-type: none"> <i>profile</i> (default value): indicates that the attributes specified in parameter "attribute" belong to the core customer profile. The unique name of the given Extension, used to create the Identification Key. If none, it means that the attributes specified in the parameter <i>attributes</i> are not sourced from either the profile or an extension. In this case, you must manage manually the identification table in UCS.
attributes	string[]	yes	The array names of one or more attributes of the extension or of the profile.
unique	bool	no	<ul style="list-style-type: none"> <i>false</i> by default. <i>true</i> to indicate that the given attributes of the profile or extension are unique across all customers.

Example

The following identification key is a phone number.

```
{ "name": "idPhone",  
  "attributes": ["phone_number"]}
```

The following sample defines "FirstName" and "LastName" as a "primary key" instead of "indexed key".

```
{ "name": "idFirstNameLastName", "unique": true, "attributes": ["FirstName", "LastName"]}
```

The combination of both parameters is unique. If your application tries to insert twice the same values, the operation will return an error.

Important

The use of a "primary key" instead of an "indexed key" makes the identification queries faster. It also prevents from creating two profiles with the same email address for example.

Related operations

- [Get Identification Keys](#)
- [Create Identification Key](#)

Profile Extension Schema

Important

Prerequisites: You need to **enable** profiles in UCS.

Purpose: Describes the schema for any service, state, or task extension.

Description

Describes one of the **ProfileExtension** resources.

- Profile extensions are supported. Before your application can start adding extensions to a given profile, your application must define this extension's schema, which includes its nested attributes.
- Service extensions are supported and multi-valued extensions are allowed.

Resource

Field	Type	Mandatory	Description
name	string	yes	<p>The unique, case-insensitive name of the extension. The name must start with a letter, and can be followed with letters, numbers, or underscores (_).</p> <p>The name is restricted to a maximum of 26 characters (Maximum RDBMS shared limit on creation of index).</p>
type	token <ul style="list-style-type: none"> • "single-valued" • "multi-valued" • "classifier" 	yes	<p>Extensions come in the following forms:</p> <ul style="list-style-type: none"> • Single-valued: a single extension record is associated with a given customer. • Multi-valued: multiple extension

Field	Type	Mandatory	Description
			<p>records can be associated with a given customer.</p> <ul style="list-style-type: none"> Classifier: the single-valued extension has no attribute.
attributes	Attribute Schema[]	no	The array of <i>zero or more</i> attributes. Empty when type="classifier".
unique	string[]	no	<p>Lists attributes which are unique in the scope of a given customer. The customer profile cannot include several extension records with identical values for these attributes.</p> <p>For example, let's consider a contact extension with the attributes "phone_num", "ext", "time_of_day". To ensure that a given customer does not have two contact extensions with the same phone number, set the following in the extension schema: unique = ["phone_num", "ext"] Each item of this array is either a single attribute, or an array of two or more attributes.</p>

Examples

Schema of a Single-Valued Extension

```
{
  "attributes": [
    {"name": "FeedbackType", "type": "string", "length": "10", "mandatory": "true"},
    {"name": "rating", "type": "integer", "mandatory": "true"},
    {"name": "notes", "type": "string", "length": "256", "mandatory": "false"}
  ],
  "name": "Feedback",
  "type": "single-valued"
}
```

Schema of a Multi-valued Extension

```
{
```

```
"attributes": [  
  {"name": "car type", "type": "string", "length": "50", "mandatory": "true"},  
  {"name": "price", "type": "integer", "mandatory": "true"},  
  {"name": "seats", "type": "integer"},  
  {"name": "comments", "type": "string", "length": "1024"}  
],  
"name": "Proposal",  
"type": "multi-valued"  
}
```

Schema of a Profile Extension

The following schema describes the Address extension.

```
"name": "Address",  
"type": "single-valued",  
"attributes": [  
  {"name": "AddressType", "type": "integer", "default": 0},  
  {"name": "Address", "type": "string", "length": 256},  
  {"name": "City", "type": "string", "length": 32},  
  {"name": "County", "type": "string", "length": 32},  
  {"name": "PostCode", "type": "string", "length": 10},  
  {"name": "Country", "type": "string", "length": 32}  
]
```

Server Operations



Purpose: Lists the operations that your application can perform on UCS.

Server Operations

- [GET /server/status](#)
- [POST /server/mode](#)

Set Server Mode



POST /server/mode

Description

Manages the UCS Mode for the Context Services. Two modes are available: Maintenance and Production.

- If the server switches from maintenance to production mode, the system waits for a configured amount of time (default 1 minute) for the completion of all schema management requests (see [Schema Operations](#)).
- If the server switches from production to maintenance mode, the system waits for the outstanding profile or service-related functions to complete. While the system is attempting to switch modes, all web service requests are denied with HTTP Status Code 503 (Service Unavailable).

Important

Read the [Server Mode](#) page of the 8.5.0 developer's guide for further details about maintenance and production mode.

Operation

ID	CV.WS.MODE		
Method	POST		
URL	/server/mode		
Parameter	Type	Mandatory	Description
mode	token <ul style="list-style-type: none"> • maintenance • production 	yes	The mode to set (case-insensitive).

Response

The UCS Server status is returned as a JSON object with the following fields: The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response	
HTTP code	200
HTTP message	OK
Header	Location /server/status
Body	<pre>{"server changed":"\${mode}"}</pre> where: <ul style="list-style-type: none">• <code>\${mode}</code> is the string for the new UCS mode (production or maintenance).

Example


Operation

```
POST /server/mode
{"mode":"production"}
```

Result

```
{"server changed":"production"}
```

Query Server Status

	GET /server/status
	Available since 8.0.100.00

Description

Queries the status of the Universal Contact Server (UCS). The status that is returned includes in particular the **Server Mode** (production or maintenance), but it also includes additional information and valuable counters to analyse the activity of the UCS.

Important

Some counters, such as "request_today", "client_errors_today", and "server_errors_today", are not persistent. They are automatically reset if the UCS is restarted.

Operation

Query Server Status

ID	CV.WS.STAT
Method	GET
URL	/server/status

Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response

HTTP code	200		
HTTP message	OK		
Name	Type	Description	
Body			

HTTP code	200	
version	string	The version of the UCS.
system_time	date/time	The system time. The date/time format is "yyyy-MM-ddTHH:mm:ss.SSSZ".
mode	token <ul style="list-style-type: none"> • production • maintenance 	The Server Mode .
max_memory	long	The maximum amount of memory the JVM will attempt to use, in KB.
allocated_memory	integer	The amount of memory currently in use by the JVM, in KB.
free_memory	integer	The total memory available to the JVM, in KB.
started	date/time<ref name="format"/>	Start date of the UCS.
last_request	date/time	The time of the last web service request.

Example

Operation

```
GET /server/status
```

Result

```
"max_memory" : 518464,
"server_errors_today" : 0,
"client_errors_today" : 0,
"request_today" : 0,
"system_time" : "2010-02-24T15:56:46.369Z",
"used_memory" : 7947,
"allocated_memory" : 63936,
"started" : "2010-02-24T14:35:28.668Z",
"free_memory" : 55989,
"last_request" : "2010-02-24T14:35:27.964Z",
"mode" : "production",
"version" : "8.0.100.11"
```