Genesys Mobile Services Deployment Guide

Configure Historical Reporting

12/14/2025

Configure Historical Reporting

Genesys Mobile Engagement can work together with other Genesys products to provide Callback Historical reporting features. This page details how you can configure Callback to implement reporting.

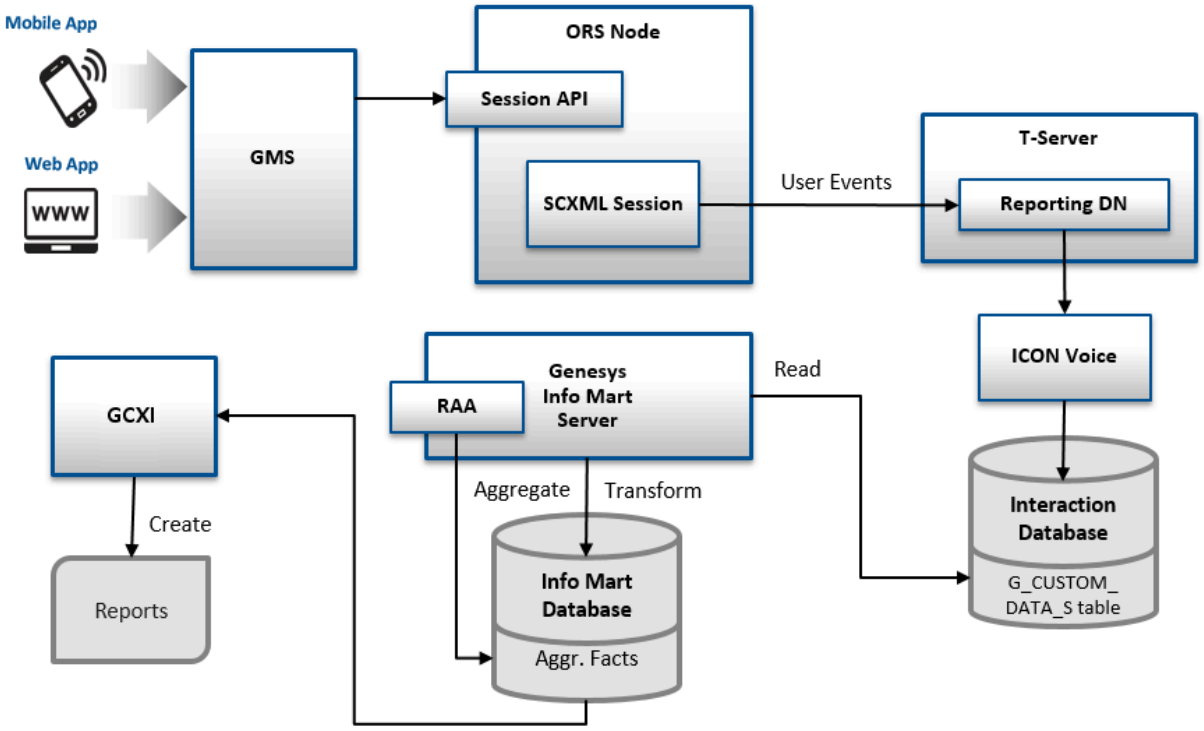
Prerequisites

Mandatory Genesys Components

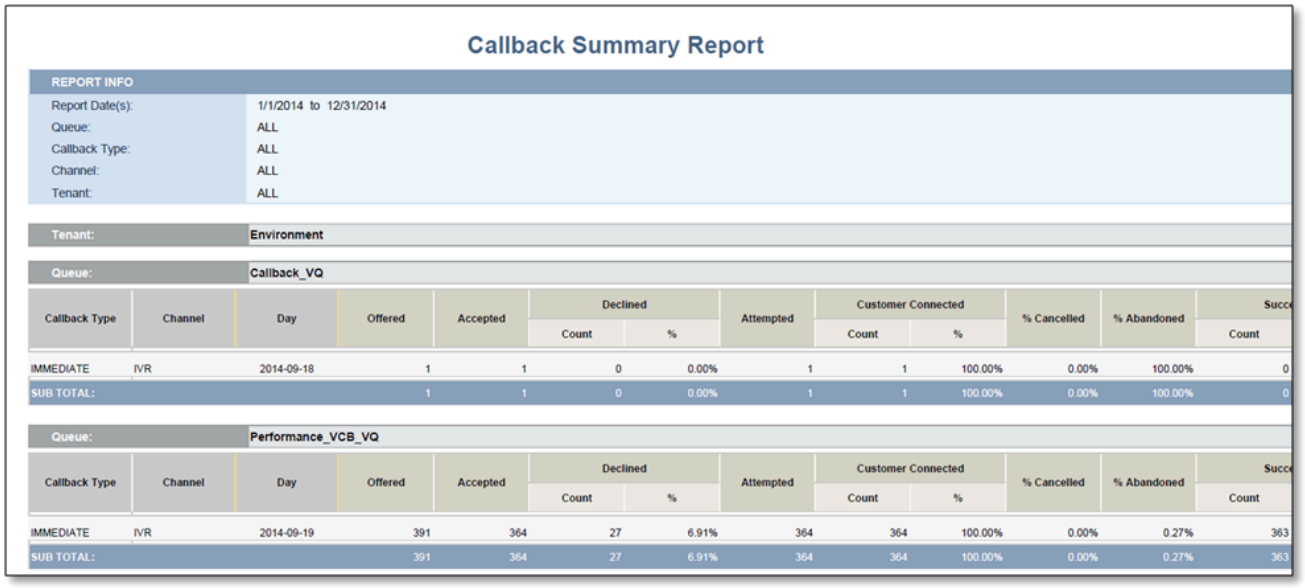
Component	Minimum Version
Orchestration Server	8.1.400.24
Universal Routing Server	8.1.400.22
Interaction Concentrator	8.1.506.07
Genesys Info Mart	8.5.005 (GA)
Reporting and Analytics Aggregates (RAA)	8.5.000.02
Genesys CX Insights (GCXI)	9.0.007.03

Historical Reporting Architecture

Reporting on Genesys Callback relies on the user-event mechanism to provide Callback-related metrics and requires Interaction Concentrator, Genesys Info Mart, and Reporting and Analytics Aggregates (RAA) to collect and organize data to produce a database from which Genesys CX Insights (GCXI) can rapidly extract the needed data.



1. Genesys Callback reports callback metrics through `UserEvents` to the configured DN. SCXML strategies that you load through the templates in the Service Management UI collect metrics and then pass the metrics as user data (KVPs) with two `UserEvent` events, one sent at the start of the session and another, at the end of the session. Genesys Info Mart has certain minimum requirements for the KVPs that must be sent. The out-of-box templates include these KVPs, as well as other KVPs that Genesys Info Mart requires for meaningful reporting. See Genesys Info Mart KVP Requirements for details.
2. Interaction Concentrator (ICON) stores the user data (KVPs) attached to these events into the `G_CUSTOM_DATA_S` table of the Interaction Database (IDB).
3. Genesys Info Mart transforms the data into the `CALLBACK_FACT` table of the Info Mart database; this format can be more quickly loaded into reports.
4. Reporting and Analytics Aggregates (RAA) aggregates the data; in other words, RAA summarizes and organizes the data from Genesys Info Mart in such a way that Genesys CX Insights (GCXI) can extract meaning.
5. Genesys CX Insights (GCXI) then presents two out-of-box callback reports: Callback Summary Report and Callback Details Report.



Callback Summary Report

REPORT INFO	
Report Date(s):	1/1/2014 to 12/31/2014
Queue:	ALL
Callback Type:	ALL
Channel:	ALL
Tenant:	ALL

Tenant:	Environment

Queue:	Callback_VQ

Callback Type	Channel	Day	Offered	Accepted	Declined		Attempted	Customer Connected		% Cancelled	% Abandoned	Succ
					Count	%		Count	%			Count
IMMEDIATE	IVR	2014-09-18	1	1	0	0.00%	1	1	100.00%	0.00%	100.00%	0
SUB TOTAL:			1	1	0	0.00%	1	1	100.00%	0.00%	100.00%	0

Queue:	Performance_VCB_VQ

Callback Type	Channel	Day	Offered	Accepted	Declined		Attempted	Customer Connected		% Cancelled	% Abandoned	Succ
					Count	%		Count	%			Count
IMMEDIATE	IVR	2014-09-19	391	364	27	6.91%	364	364	100.00%	0.00%	0.27%	363
SUB TOTAL:			391	364	27	6.91%	364	364	100.00%	0.00%	0.27%	363

For example, for reporting purposes, the following are some of the keys that GMS sends in `UserEvents` related to Outbound calls:

- _CB_T_SERVICE_START
- _CB_SERVICE_ID
- _CB_D_CALLBACK_OFFER
- _CB_N_CALLBACK_OFFERED
- _CB_T_CALLBACK_OFFERED
- _CB_T_CALLBACK_ACCEPTED
- _CB_T_CUSTOMER_CONNECTED
- _CB_N_IS_SNOOZED
- _CB_T_NEXT_REDIAL_ATTEMPT
- _CB_N_CALLBACK_MEDIA_ATTEMPTS
- _CB_T_LAST_DIAL_ATTEMPT
- _CB_N_AGENT_ADDED_TO_IXN

The keys that GMS sends depend on the scenario. To get a complete list of the keys that might be sent, refer to the Callback KVPs reference on this page.

> ## Important
>
> If the _CB_T_CALLBACK_OFFERED and _CB_T_CALLBACK_ACCEPTED KVPs must be added to the original session that initiated the callback request, the callback request must include the _originating_interaction_id option. In this scenario, in the callback request,

set the _originating_interaction_id value to the interaction ID of the inbound call that is managed by the ORS session.

Genesys Info Mart KVP Requirements

The following KVPs are mandatory. Genesys Info Mart will not create a record for the callback event if the KVP is missing from the `UserEvent`.

- _CB_SERVICE_ID
- _CB_T_SERVICE_START
- _CB_D_CALLBACK_OFFER
- _CB_N_CALLBACK_OFFERED
- _CB_T_CALLBACK_OFFERED

Important

If the `_CB_T_CALLBACK_OFFERED` and `_CB_T_CALLBACK_ACCEPTED` KVPs must be added to the original session that initiated the callback request, the callback request must include the _originating_interaction_id option. In this scenario, in the callback request, set the _originating_interaction_id value to the interaction ID of the inbound call that is managed by the ORS session.

The following four KVPs need to be sent in both `UserEvents` and as call-based attached data in `TEvents`.The duplicated KVPs enable Genesys Info Mart to associate the callback event with interaction data.

- `_CB_T_CALLBACK_ACCEPTED`
- `_CB_T_SERVICE_START`
- `_CB_SERVICE_ID`
- `_CB_T_CUSTOMER_CONNECTED`

For meaningful reporting, Genesys Info Mart requires several other KVPs, depending on the callback scenario. See the Callback KVPs reference, below, for the complete list.

Important

- The `_CB_SERVICE_ID` is returned by the GMS API in response to the callback request.
- For Inbound Calls, where the in-queue callback offer was presented and accepted, _CB_T_CALLBACK_ACCEPTED, _CB_T_SERVICE_START, and _CB_SERVICE_ID must be attached at the time at which the callback was accepted.
- For Virtual and Outbound Calls, _CB_T_CUSTOMER_CONNECTED must be attached at the time at which the customer was connected.

Virtual Queues

As a best practice, Genesys recommends creating virtual queues associated with the following interaction types:

- Virtual Queue for Inbound calls—This queue is where the regular inbound calls are going to be reported. Those calls are callbacks that were not offered or, offered and rejected.
- Virtual Queue for Virtual callbacks—This queue is where the virtual callbacks are going to be waiting for an agent.
- Virtual Queue for Outbound calls—This is where the callback application will place the real outbound call when it gets confirmation that the right person is connected. The call is removed from this queue after it is successfully delivered to an agent or is abandoned by the customer.

Important

Virtual queues (VQ) that are used for reporting will make metrics effective, but they are not used for routing in this context.

Related Resources for Historical Reporting

You may also be interested in reading:

- The Genesys Info Mart Physical Data Model documentation for your RDBMS.
- The Reporting and Analytics Aggregates Physical Data Model documentation for your RDBMS.

Configure Historical Reporting

Important

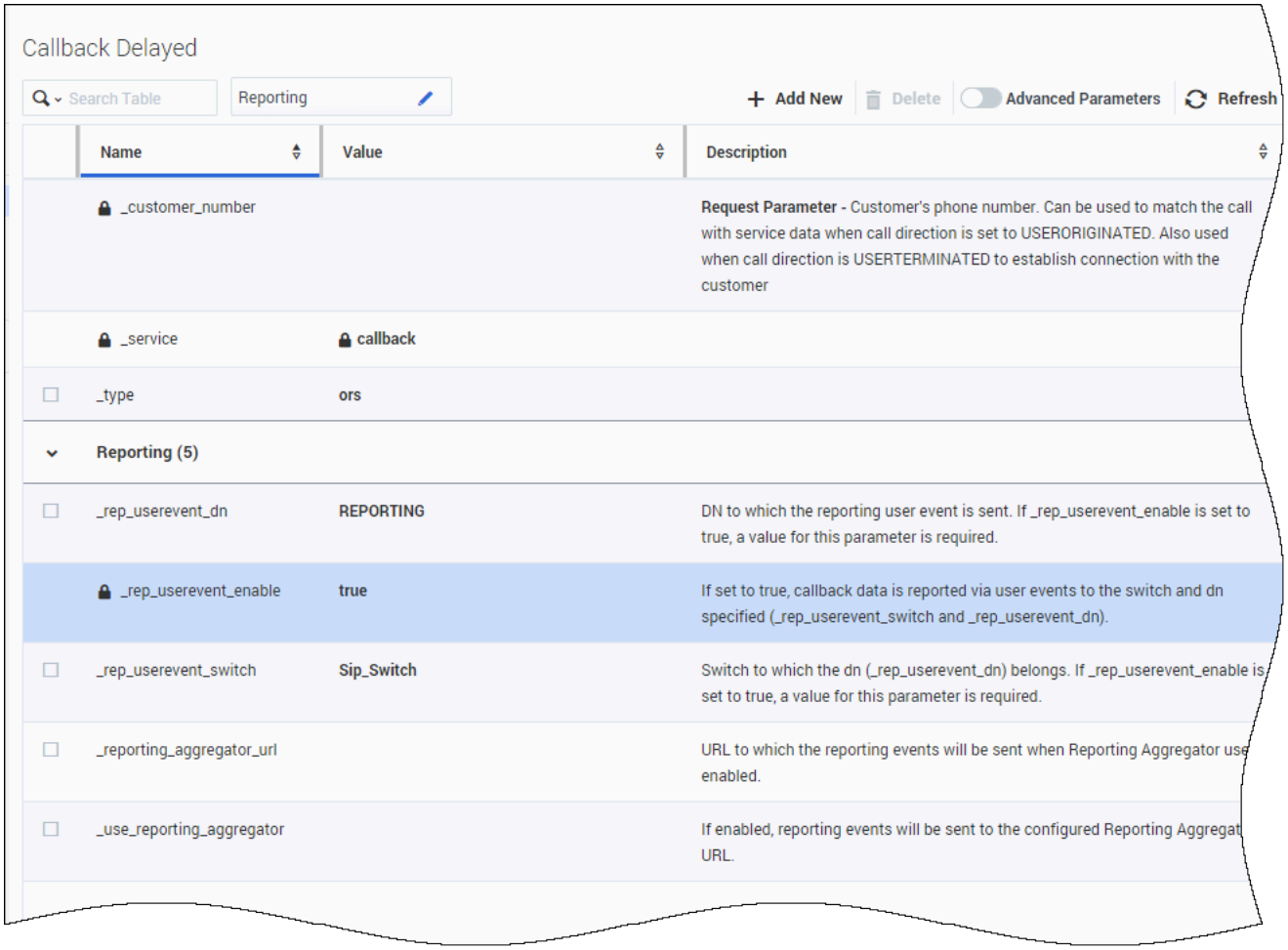
Genesys Info Mart and Genesys CX Insights (GCXI) support for callback offered

through GMS is provided out-of-box, with no additional configuration required. To see callback data in GCXI reports, however, you need to modify the configuration for other products as explained in this section.

Configure a Reporting DN

Open Genesys Administrator or Configuration Manager and create a new DN of type **Trunk Group DN**. The name of the DN is used inside SCXML scripts, so it should be meaningful and recognizable. For example: `Sip_Switch > DN > REPORTING`

Configure your Callback Service



Edit your callback service in **Callback and Mobile Engagement > Configured Services**, expand the **Reporting** section:

- Set the `_rep_userevent_enable` option to true to enable reporting.
- Set the `_rep_userevent_dn` option to the Trunk Group DN that you created previously, used as destination DN of the reporting events.
- Set the `_rep_userevent_switch` option to the Switch name where you created this DN. This is Switch used to report the events.

Configure Orchestration Server

In the connections of your Orchestration Server application, add the T-Server used to define the reporting Switch and DN in the GMS service configuration. For example, `Sip_Switch`.

Configure Interaction Concentrator

To make Callback reporting work, you need to configure Interaction Concentrator (ICON) for Voice. See here for details.

Set the KVP list

- Configure ICON to store the KVP data provided in the `UserData` section of `EventUserEvents`. ICON will store this data in the `G_CUSTOM_DATA_S` table of the Interaction Database (IDB):
  ```
  ICON > Options > custom-states/store-event-data=all
  ```
  By default, `store-event-data` is set to `none`.
- Configure ICON to store required duplicate KVP data provided in the `UserData` attribute of `TEvents`. ICON will store this data in the `G_USERDATA_HISTORY` table. To enable this storage, modify your `ccon_adata_spec.xml` file to capture the four `TEvents` KVPs described in the Callback KVPs reference below:
  - `_CB_T_CALLBACK_ACCEPTED`
  - `_CB_T_CUSTOMER_CONNECTED`
  - `_CB_T_SERVICE_START`
  - `_CB_SERVICE_ID`

> **Tip**
>
> See the `ccon_adata_spec_GIM_example.xml` file in the Genesys Info Mart installation package for an example of the required modification.

Check Interaction Concentrator Connections

Make sure that Interaction Concentrator is connected to the T-Server that is servicing the switch specified in the Callback Service.
For example: `Sip_Switch`.

Start Interaction Concentrator and use logs to verify that it registered on the `REPORTING DN`.

> **Important**
> Interaction Concentrator does not produce historical records for virtual interactions.

Configure Reporting and Analytics Aggregates

Edit the Genesys Info Mart application to enable the `agg-feature\enable-callback` option:
`agg-feature\enable-callback=yes`

> **Tip**
> See here for details about the configuration of your RAA application.

Configure Workspace

Important: In a Callback use case with preview, reporting user data is attached to the call that appears on Agent Desktop (WDE). Once the callback is finished, from a GMS Callback point of view, the agent is managing wrap-up operations for the call and sends a user request to the reporting server using the callback user data. The reporting server sees this data as an additional reporting operation.

To avoid sending this additional reporting data, the agent desktop application can configure the following option in the `interaction-workspace` section:

```
interaction.disposition.use-attached-data=false
```

Verify Reporting Data

1. Run your scenario by triggering Genesys Mobile Services and Orchestration Server (ORS) APIs directly.
2. Make sure user events are being delivered to Interaction Concentrator applications by checking T-Server logs. You should see something like this:

```
00:34:20.757 Int 04543 Interaction message "RequestDistributeUserEvent" received from
516 ("OrchestrationServer")
  -- Absent ThisDN, REPORTING was used
  @00:34:20.7570 [0] 8.1.000.62 send_to_client: message EventACK
          AttributeEventSequenceNumber        0000000000000ef8
          AttributeCustomerID        'Environment'
          AttributeTimeinuSecs        757000
          AttributeTimeinSecs        1348817660 (00:34:20)
          AttributeReferenceID        431
          AttributeThisDN        'REPORTING'
          AttributeUserEvent        RequestDistributeUserEvent
  00:34:20.757 Trc 04542 EventACK sent to [516] (00000003 OrchestrationServer 192.168.27.50:40727)
  @00:34:20.7570 [0] 8.1.000.62 distribute_user_event: message EventUserEvent
          AttributeEventSequenceNumber        0000000000000ef9
          AttributeCustomerID        'Environment'
          AttributeTimeinuSecs        757000
          AttributeTimeinSecs        1348817660 (00:34:20)
          AttributeUserEvent        EventUserEvent
          AttributeUserData        [347] 00 0c 00 00..
                  'gms_AgentAvailable'        '1348817660755'
                  'gms_AgentConnected'        ''
                  'gms_IxnCompleted'        ''
                  'gms_ServiceName'        'inbound-delay'
                  'gms_ServiceStartAt'        '1348817660553'
                  'gms_ServiceStoppedAt'        ''
                  'gms_SessionEventSeq'        3
                  'gms_SessionId'        '65UA6ISSJH76R340BNDQ2DG0DG000036'
                  'gms_UserConnected'        ''
                  'gms_UserId'        ''
                  'gms_WaitingForAgent'        '1348817660744'
                  'gms_externalId'        ''
          AttributeANI        '777'
          AttributeDNIS        '333'
          AttributeReferenceID        431
          AttributeThisDN        'REPORTING'
  00:34:20.758 Trc 04542 EventUserEvent sent to [508] (0000000c Icon_Voice 192.168.27.50:42678)
  00:34:20.758 Trc 04542 EventUserEvent sent to [588] (00000004 Stat_Server 192.168.27.50:40728)
  00:34:20.758 Trc 04542 EventUserEvent sent to [592] (00000005 Universal_Routing_Server
 192.168.27.50:40744)
```

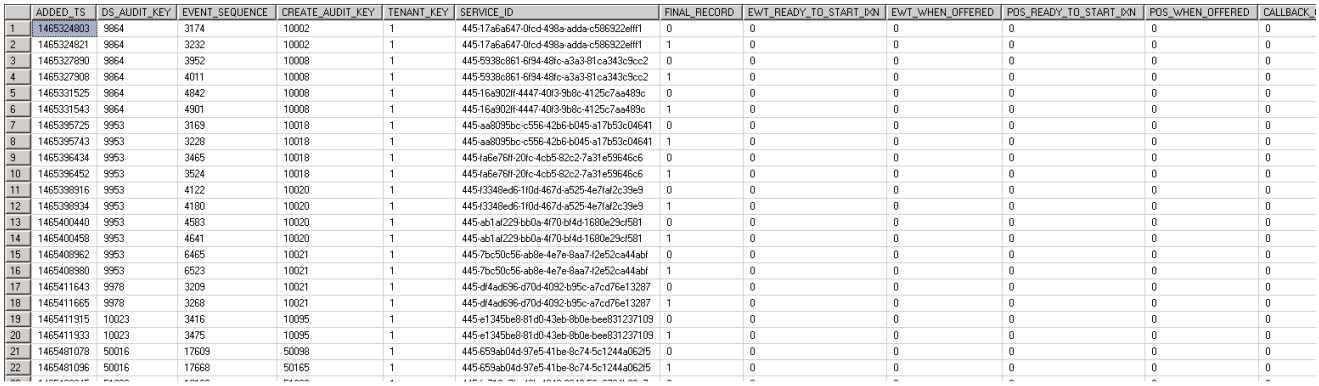
3. Check your Interaction Concentrator logs and the G_CUSTOM_DATA_S table in Interaction Database and make sure that data is recorded properly.

For example, you should see in Interaction Concentrator logs:

```
00:39:19.569 Int 04543 Interaction message "EventUserEvent" received from 65200 ("SIP_Server@REPORTING")
00:39:19.751 Int 04543 Interaction message "EventUserEvent" received from 65200 ("SIP_Server@REPORTING")
00:39:19.766 Int 04543 Interaction message "EventUserEvent" received from 65200 ("SIP_Server@REPORTING")
00:39:19.987 Trc 25016 Persistent Queue GUD: transaction 10929 is committed. 5 records written
into the queue
00:39:19.987 Trc 25003 Database queue [GUD]: persistent queue transaction 10929 is being processed.
00:39:20.001 Trc 25004 Database queue [GUD]: persistent queue transaction 10929 is processed, committed
 and removed. 5 records are written.
```

4. Optionally, you can also check the content of the CALLBACK_FACT table in the Info Mart database to make sure that the transformation process is correctly executed as well. For example, you can try the following query:

```
SELECT * FROM dbo. CALLBACK_FACT
```



	ADDED_TS	DS_AUDIT_KEY	EVENT_SEQUENCE	CREATE_AUDIT_KEY	TENANT_KEY	SERVICE_ID	FINAL_RECORD	EWT_READY_TO_START_IXN	EWT_WHEN_OFFERED	POS_READY_TO_START_IXN	POS_WHEN_OFFERED	CALLBACK_
1	1465324803	9864	3174	10002	1	445-17a6a647-0fcd-498a-adda-c586922efff1	0	0	0	0	0	0
2	1465324821	9864	3232	10002	1	445-17a6a647-0fcd-498a-adda-c586922efff1	1	0	0	0	0	0
3	1465327890	9864	3952	10008	1	445-5938c861-6f94-48fc-a3a3-81ca343c9cc2	0	0	0	0	0	0
4	1465327908	9864	4011	10008	1	445-5938c861-6f94-48fc-a3a3-81ca343c9cc2	1	0	0	0	0	0
5	1465331525	9864	4842	10008	1	445-16a902ff-4447-40f3-9b8c-4125c7aa489c	0	0	0	0	0	0
6	1465331543	9864	4901	10008	1	445-16a902ff-4447-40f3-9b8c-4125c7aa489c	1	0	0	0	0	0
7	1465395725	9953	3169	10018	1	445-aa8095bc-c556-42b6-b045-a17b53c04641	0	0	0	0	0	0
8	1465395743	9953	3228	10018	1	445-aa8095bc-c556-42b6-b045-a17b53c04641	1	0	0	0	0	0
9	1465396434	9953	3465	10018	1	445-fa6e76ff-20fc-4cb5-82c2-7a31e59646c6	0	0	0	0	0	0
10	1465396452	9953	3524	10018	1	445-fa6e76ff-20fc-4cb5-82c2-7a31e59646c6	1	0	0	0	0	0
11	1465398916	9953	4122	10020	1	445-f3348ed6-1f0d-467d-a525-4e7faf2c39e9	0	0	0	0	0	0
12	1465398934	9953	4180	10020	1	445-f3348ed6-1f0d-467d-a525-4e7faf2c39e9	1	0	0	0	0	0
13	1465400440	9953	4583	10020	1	445-ab1af229-bb0a-4f70-bf4d-1680e29cf581	0	0	0	0	0	0
14	1465400458	9953	4641	10020	1	445-ab1af229-bb0a-4f70-bf4d-1680e29cf581	1	0	0	0	0	0
15	1465408962	9953	6465	10021	1	445-7bc50c56-ab8e-4e7e-8aa7-f2e52ca44abf	0	0	0	0	0	0
16	1465408980	9953	6523	10021	1	445-7bc50c56-ab8e-4e7e-8aa7-f2e52ca44abf	1	0	0	0	0	0
17	1465411643	9978	3209	10021	1	445-df4ad696-d70d-4092-b95c-a7cd76e13287	0	0	0	0	0	0
18	1465411665	9978	3268	10021	1	445-df4ad696-d70d-4092-b95c-a7cd76e13287	1	0	0	0	0	0
19	1465411915	10023	3416	10095	1	445-e1345be8-81d0-43eb-8b0e-bee831237109	0	0	0	0	0	0
20	1465411933	10023	3475	10095	1	445-e1345be8-81d0-43eb-8b0e-bee831237109	1	0	0	0	0	0
21	1465481078	50016	17609	50098	1	445-659ab04d-97e5-41be-8c74-5c1244a062f5	0	0	0	0	0	0
22	1465481096	50016	17668	50165	1	445-659ab04d-97e5-41be-8c74-5c1244a062f5	1	0	0	0	0	0

How to Pass Reporting KVPs of the Inbound Call in the Callback Request

Some historical reporting KVP values are known only by the IVR or application that requests the callback service. Including these KVPs in the historical reporting is optional. If you want to include them, the values can be passed in the HTTP request that starts the Callback service. The following is the list of the KVP parameters that can be passed in the HTTP request. Each maps to the corresponding _CB_*X* KVP.

- _cb_t_callback_offered
- _cb_d_callback_offer
- _cb_ewt_when_callback_was_offered
- _cb_pos_when_callback_was_offered
- _cb_t_callback_accepted
- _cb_dim_channel
- _cb_dim_callback_offer_type
- _cb_dim_offer_timing

- _cb_n_callback_offers_per_session
- _cb_d_last_callback_offer

> **Important**
> If the agent submits the completed reason in the disposition result, the system will set the reporting key _CB_DISPOSITION to the provided COMPLETED reason.

> **Important**
> If the _cb_t_callback_offered and _cb_t_callback_accepted KVPs must be added to the original session that initiated the callback request, the callback request must include the _originating_interaction_id option. In this scenario, in the callback request, set the _originating_interaction_id value to the interaction ID of the inbound call that is managed by the ORS session.

Reference: Callback KVPs

The following table describes the KVPs that, if sent by GMS in UserEvents, Genesys Info Mart uses to enable Callback reporting.

The following four KVPs must also be sent as call-based attached data.

- _CB_SERVICE_ID
- _CB_T_SERVICE_START
- _CB_T_CALLBACK_ACCEPTED
- _CB_T_CUSTOMER_CONNECTED

> **Important**
> The sample attached-data specification file in the Genesys Info Mart IP includes these four KVPs by default.

KVP	Description	Info Mart Database Target
_CB_TENANT_DBID	The Tenant DBID.	CALLBACK_FACT.TENANT_KEY
_CB_DISPOSITION	Callback state using the format `<state>.<sub state>` where: • `<state>` can be set to: `SCHEDULED, QUEUED, ROUTING, PROCESSING, COMPLETED`. • `<sub state>` can be set: `REDIAL_LIMIT_REACHED, CANCELLED, AGENT, ABANDONED_IN_QUEUE, REJECTED, PUSH_SEND, PUSH_DELIVERY_CONFIRMED, PUSH_SEND_ERROR, FAILED, CONNECTED, TRANSFERRED_TO_RP`.	CALLBACK_DIM_3.DISPOSITION (referenced through CALLBACK_FACT.CALLBACK_DIM_3_KEY)
_CB_SERVICE_ID*	The ID of the callback service request. Depending on the scenario, the value equals the ID of the GMS service instance or ID of the ORS session.	CALLBACK_FACT.SERVICE_ID
_CB_ORIGINATION_IXN_ID **Introduced:** GMS 8.5.200.07	The ID of the inbound call where the callback was originally offered and accepted. You must pass the `_cb_origination_ixn_id` parameter in your Start Callback query when creating a callback request. If you do not pass the `_cb_origination_ixn_id` parameter, the value of `_CB_ORIGINATION_IXN_ID` will be `undefined`. For chat scenarios, this ID should be the chat interaction ID.	CALLBACK_FACT.ORIGINATION_IXN_ID
_CB_FIRST_OUT_IXN_ID **Introduced:** GMS 8.5.200.07	The call ID of the first outbound call that the callback service created.	CALLBACK_FACT.FIRST_OUT_IXN_ID
_CB_LAST_OUT_IXN_ID **Introduced:** GMS 8.5.200.07	The call ID of the last outbound call that the callback service created.	CALLBACK_FACT.LAST_OUT_IXN_ID

KVP	Description	Info Mart Database Target
_CB_DIAL_1_RESULT **Introduced:** GMS 8.5.200.07	The result of the first callback dialing attempt. One of the following values: • CREATE_CALL_ERROR • BUSY • NO_ANSWER • ANSWERING_MACHINE • ERROR_TONE • FAX • PERSON • CONNECTED • FAILED_TO_ESTABLISH_CUSTOMER_ORIGINATED_MEDIA • PUSH_DELIVERY_CONFIRMED • PUSH_SEND_ERROR • PUSH_DELIVERY_NOT_CONFIRMED • USERORIGINATED_CONNECTED **Notes:** FAILED_TO_ESTABLISH_CUSTOMER_ORIGINATED_MEDIA is a result that must be reported by the user application; otherwise, there is no CTI data that will enable Genesys Callback to identify this result.	CALLBACK_DIAL_RESULTS.DIAL_1_RESULT (referenced through CALLBACK_FACT.CALLBACK_DIAL_RESULTS_KEY)
_CB_DIAL_2_RESULT **Introduced:** GMS 8.5.200.07	The result of the second callback dialing attempt. See _CB_DIAL_1_RESULT for possible values.	CALLBACK_DIAL_RESULTS.DIAL_2_RESULT (referenced through CALLBACK_FACT.CALLBACK_DIAL_RESULTS_KEY)
_CB_DIAL_3_RESULT	The result of the third callback dialing attempt. See _CB_DIAL_1_RESULT for possible values.	CALLBACK_DIAL_RESULTS.DIAL_3_RESULT (referenced through

KVP	Description	Info Mart Database Target
Introduced: GMS 8.5.200.07		CALLBACK_FACT.CALLBACK_DIAL_RESULTS_KEY)
_CB_DIAL_4_RESULT **Introduced:** GMS 8.5.200.07	The result of the fourth callback dialing attempt. See _CB_DIAL_1_RESULT for possible values.	CALLBACK_DIAL_RESULTS.DIAL_4_RESULT (referenced through CALLBACK_FACT.CALLBACK_DIAL_RESULTS_KEY)
_CB_DIAL_5_RESULT **Introduced:** GMS 8.5.200.07	The result of the fifth callback dialing attempt. See _CB_DIAL_1_RESULT for possible values.	CALLBACK_DIAL_RESULTS.DIAL_5_RESULT (referenced through CALLBACK_FACT.CALLBACK_DIAL_RESULTS_KEY)
_CB_T_DIAL_1 **Introduced:** GMS 8.5.200.07	UTC Timestamp of the first dialing attempt.	CALLBACK_FACT.DIAL_1_TS
_CB_T_DIAL_2 **Introduced:** GMS 8.5.200.07	UTC Timestamp of the second dialing attempt.	CALLBACK_FACT.DIAL_2_TS
_CB_T_DIAL_3 **Introduced:** GMS 8.5.200.07	UTC Timestamp of the third dialing attempt.	CALLBACK_FACT.DIAL_3_TS
_CB_T_DIAL_4 **Introduced:** GMS 8.5.200.07	UTC Timestamp of the fourth dialing attempt.	CALLBACK_FACT.DIAL_4_TS
_CB_T_DIAL_5 **Introduced:** GMS 8.5.200.07	UTC Timestamp of the fifth dialing attempt.	CALLBACK_FACT.DIAL_5_TS
_CB_IXN_START_IGNORING_AVAILABILITY	For premise callback, `_CB_IXN_START_IGNORING_AVAILABILITY` will	CALLBACK_DIM_4.DIAL_IGNORING_AVAILABILITY

KVP	Description	Info Mart Database Target
Introduced: GMS 8.5.200.07	always be 0.	
_CB_FINAL_RECORD	Indicates whether this is a final record about this callback service: 0 = No, 1 = Yes.	CALLBACK_FACT.FINAL_RECORD
_CB_EWT_WHEN_READY_TO_START_MEDIA_IXN	The value of Expected Wait Time (EWT), in seconds, for the service request when the contact center was ready to start the first callback interaction, such as an outbound dialing attempt.	CALLBACK_FACT.EWT_READY_TO_START_IXN
_CB_EWT_WHEN_CALLBACK_WAS_OFFERED	The value of EWT, in seconds, at the time the callback was offered.	CALLBACK_FACT.EWT_WHEN_OFFERED
_CB_POS_WHEN_READY_TO_START_MEDIA_IXN	The customer position in the queue when the contact center was ready to start the first callback interaction, such as an outbound dialing attempt.	CALLBACK_FACT.POS_READY_TO_START_IXN
_CB_POS_WHEN_CALLBACK_WAS_OFFERED	The customer position in the queue when callback was offered.	CALLBACK_FACT.POS_WHEN_OFFERED
_CB_D_CALLBACK_OFFER	The duration of the callback offer, in seconds.	CALLBACK_FACT.CALLBACK_OFFER_TIME
_CB_OFFER_EWT_INBOUND_VQ **Introduced:** GMS 8.5.111.04	Estimated Wait Time for the queue where rejected calls and not offered callbacks are being placed. This value is identical to _CB_EWT_WHEN_CALLBACK_WAS_OFFERED if the same Virtual Queue is used to place accepted callbacks.	CALLBACK_FACT.EWT_WHEN_REJECTED
_CB_N_ABANDONED_DURING_CALLBACK_OFFER **Introduced:** GMS 8.5.111.04	Indicates whether the caller dropped the call without explicitly accepting or rejecting the callback offer: 0 = No, 1 = Yes.	CALLBACK_DIM_4.ABANDONED_DURING_CB_OFFER (referenced through CALLBACK_FACT.CALLBACK_DIM_4_KEY)
_CB_CUSTOMER_ANI	ANI of the customer for in-queue scenarios. This value can match _CB_CUSTOMER_PHONE_NUMBER if the same	CALLBACK_FACT.CUSTOMER_ANI

KVP	Description	Info Mart Database Target
Introduced: GMS 8.5.111.04	number is confirmed or entered. Could also be empty if the ANI is not detected.	
_CB_T_SERVICE_END **Introduced:** GMS 8.5.111.04	UTC timestamp for when service was completed or terminated.	CALLBACK_FACT.SERVICE_END_TS
_CB_D_CUSTOMER_WAITED_BEFORE_OFFER **Introduced:** GMS 8.5.106.14	The amount of time, in seconds, the customer waited in the queue before a callback was offered.	CALLBACK_FACT.WAITED_BEFORE_OFFER_TIME
_CB_D_WAITING_FOR_AGENT_OFFLINE	The amount of time, in seconds, the customer was waiting offline for an agent to become available.	CALLBACK_FACT.WAIT_AGENT_OFFLINE_TIME
_CB_D_ESTABLISH_MEDIA_IXN	The amount of time, in seconds, it took to establish the callback interaction, such as an outbound call.	CALLBACK_FACT.ESTABLISH_MEDIA_IXN_TIME
_CB_D_CUSTOMER_CONNECTED_WAITING_FOR_AGENT	The amount of time, in seconds, the customer was waiting to be connected to the agent after the callback interaction was established.	CALLBACK_FACT.CONN_WAITING_AGENT_TIME
_CB_T_CALLBACK_ACCEPTED*	The UTC timestamp when the callback offer was accepted.	CALLBACK_FACT.CALLBACK_ACCEPTED_TS
_CB_T_CALLBACK_OFFERED	The UTC timestamp when the callback was offered.	CALLBACK_FACT.CALLBACK_OFFERED_TS
_CB_T_READY_TO_START_MEDIA_IXN	The UTC timestamp when the contact center was ready to start the callback interaction. The value matches the time of either an outbound dialing attempt or a push notification prompting the customer to start a call or chat session. **Note:** Set this value only once, before the first dial attempt.	CALLBACK_FACT.READY_START_MEDIA_IXN_TS
_CB_T_CUSTOMER_CONNECTED*	The UTC timestamp when the customer was reconnected to the contact center and started	CALLBACK_FACT.CUSTOMER_CONNECTED_TS

KVP	Description	Info Mart Database Target
	waiting for an agent to be connected.	
_CB_N_AGENT_ADDED_TO_IXN	Indicates whether the agent was successfully added to the callback interaction: 0 = No, 1 = Yes.	CALLBACK_FACT.AGENT_ADDED_TO_IXN
_CB_N_TRANSFER_TO_AGENT_FAILED	Number of times the callback interaction failed to transfer to the agent.	CALLBACK_FACT.XFER_TO_AGENT_FAILED
_CB_N_CUSTOMER_ABANDONED_WHILE_WAITING_FOR_AGENT	Indicates whether the customer abandoned the callback interaction while waiting to be connected to an agent: 0 = No, 1 = Yes.	CALLBACK_FACT.ABANDONED_WAITING
_CB_N_TIMEOUT_WHILE_WAITING_FOR_AGENT	Indicates whether the customer was disconnected because the timeout for waiting for an agent was reached: 0 = No, 1 = Yes.	CALLBACK_FACT.TIMEOUT_WAITING
_CB_N_IXN_REQ_AGENT	Indicates whether the interaction required agent assistance: 0 = No, 1 = Yes.	CALLBACK_FACT.IXN_REQ_AGENT
_CB_N_CALLBACK_OFFERED	Indicates whether callback was offered, at least once, during the session: 0 = No, 1 = Yes.	CALLBACK_FACT.CALLBACK_OFFERED
_CB_N_CALLBACK_ACCEPTED	Indicates whether a callback offer was accepted: 0 = No, 1 = Yes.	CALLBACK_FACT.CALLBACK_ACCEPTED
_CB_N_CALLBACK_MEDIA_ATTEMPTS	The total number of callback attempts or notifications, both successful and unsuccessful.	CALLBACK_FACT.CALLBACK_ATTEMPTS
_CB_T_SERVICE_START*	The UTC timestamp when the callback service started. This value represents either the time of the callback request or the time that the callback offer was played, depending on deployment.	CALLBACK_FACT.SERVICE_START_TS, CALLBACK_FACT.START_DATE_TIME_KEY
_CB_DIM_VQ_DBID	The DBID of the virtual queue used to find the target agent. Genesys Info Mart uses this value in combination to identify the RESOURCE_KEY to use.	CALLBACK_FACT.RESOURCE_KEY
VQ_CFG_TYPE_ID	The configuration type ID of the virtual queue used to find the target agent. Genesys Info Mart uses this value in combination to identify the	CALLBACK_FACT.RESOURCE_KEY

KVP	Description	Info Mart Database Target
	RESOURCE_KEY to use.	
VQ_CFG_TYPE	The configuration type of the virtual queue used to find the target agent. Genesys Info Mart uses this value in combination to identify the RESOURCE_KEY to use.	CALLBACK_FACT.RESOURCE_KEY
_CB_DIM_VQ	The virtual queue used to find the target agent. Genesys Info Mart uses this value in combination to identify the RESOURCE_KEY to use.	CALLBACK_FACT.RESOURCE_KEY
_CB_DIM_CHANNEL	The interaction channel from which the callback originated. One of the following values: • IVR • WEB • MOBILE	CALLBACK_DIM_1.CHANNEL (referenced through CALLBACK_FACT.CALLBACK_DIM_1_KEY)
_CB_DIM_CALLBACK_OFFER_TYPE	The type of callback offer that was presented to the customer. For example, after business hours, SCHEDULED is the only available option; during business hours, business rules might allow only the WAIT_FOR_AGENT option or a combination of SCHEDULED and WAIT_FOR_AGENT. One of the following values: • SCHEDULED • WAIT_FOR_AGENT • COMBINED_SCHEDULED_AND_WAIT_FOR_AGENT • IMMEDIATE	CALLBACK_DIM_1.CALLBACK_OFFER_TYPE (referenced through CALLBACK_FACT.CALLBACK_DIM_1_KEY)
_CB_DIM_TYPE	The type of callback the customer requested. One of the following values:	CALLBACK_DIM_1.CALLBACK_TYPE (referenced through CALLBACK_FACT.CALLBACK_DIM_1_KEY)

KVP	Description	Info Mart Database Target
	• IMMEDIATE - The interaction is created right away while the customer is waiting for the agent (in an online chat session or waiting for a voice call). • WAIT_FOR_AGENT - The interaction is delayed until the agent is about to become available or actually becomes available (as in an agent first scenario). • SCHEDULED - The time for the callback interaction is negotiated with the customer.	
_CB_DIM_CONNECT_ORDER	The order in which the final callback interaction was connected. One of the following values: • CUSTOMER_FIRST • AGENT_FIRST_PREVIEW • AGENT_FIRST_NO_PREVIEW	CALLBACK_DIM_1.CONNECT_ORDER (referenced through CALLBACK_FACT.CALLBACK_DIM_1_KEY)
_CB_DIM_DIAL_DIALOG_RESULT	The result of the final dialog for the callback. One of the following values: • RIGHT_PERSON • RESCHEDULED • CANCELLED • TRANSFERRED_TO_RP • PERSON • CANCEL • ERROR_TONE	CALLBACK_DIM_2.DIAL_DIALOG_RESULT (referenced through CALLBACK_FACT.CALLBACK_DIM_2_KEY)

KVP	Description	Info Mart Database Target
	Important: If an error occurs during the callback outbound call, the value of _CB_DIM_FINAL_DIAL_RESULT might overlap with _CB_DIM_DIAL_DIALOG_RESULT.	
_CB_DIM_CALL_DIRECTION	The direction of the final callback interaction. One of the following values: • CUSTOMER_TERMINATED - Outbound Callback scenarios in which the contact center is dialing out to the customer's number. • CUSTOMER_ORIGINATED - Inbound Callback scenarios in which the contact center notifies the customer-facing application that it is time for the callback interaction, after which the application creates the interaction (such as a call or chat), obtaining the phone number if necessary. In this scenario, a customer call comes into the contact center as a regular inbound call, but it is recognized as the callback interaction.	CALLBACK_DIM_2.CALL_DIRECTION (referenced through CALLBACK_FACT.CALLBACK_DIM_2_KEY)
_CB_DIM_FINAL_DIAL_RESULT	The result of the final callback dialing attempt. One of the following values: • CREATE_CALL_ERROR • BUSY • NO_ANSWER • ANSWERING_MACHINE • ERROR_TONE • FAX • PERSON	CALLBACK_DIM_2.FINAL_DIAL_RESULT (referenced through CALLBACK_FACT.CALLBACK_DIM_2_KEY)

KVP	Description	Info Mart Database Target
	• CANCEL • CONNECTED • FAILED_TO_ESTABLISH_CUSTOMER_ORIGINATED_MEDIA • PUSH_DELIVERY_CONFIRMED • PUSH_SEND_ERROR • PUSH_DELIVERY_NOT_CONFIRMED • USERORIGINATED_CONNECTED • REDIAL_LIMIT_REACHED • ABANDONED_IN_QUEUE • FAIL • UNKNOWN • RESCHEDULED • FAIL_FAX_REACHED **Notes:** 1. FAILED_TO_ESTABLISH_CUSTOMER_ORIGINATED_MEDIA is a result that must be reported by the user application; otherwise, there is no CTI data that will enable Genesys Callback to identify this result. 2. CANCEL is set when the on_dial plugin returned action=CANCEL.	
_CB_DIM_OFFER_TIMING	Specifies whether the callback offer was made during operational (business) or non-operational hours. One of the following values: • ON-HOURS	CALLBACK_DIM_2.OFFER_TIMING (referenced through CALLBACK_FACT.CALLBACK_DIM_2_KEY)

KVP	Description	Info Mart Database Target
	• OFF-HOURS	
_CB_DIM_FINAL_TARGET	The routing target that was used to find the agent.	CALLBACK_DIM_3.FINAL_TARGET (referenced through CALLBACK_FACT.CALLBACK_DIM_3_KEY)
_CB_ORS_SESSION_ID **Introduced:** GMS 8.5.114.09	The Orchestration Server (ORS) session ID used to manage the callback. If multiple sessions were used (for example, because an ORS session terminated unexpectedly during the callback), the last session ID is reported.	CALLBACK_FACT.ORS_SESSION_ID
_CB_EWT_WHEN_READY_TO_START_LAST_MEDIA_IXN **Introduced:** GMS 8.5.200.07	Estimated Wait Time in seconds when the last dial attempt was made or the last push notification sent.	CALLBACK_FACT.EWT_WHEN_LAST_DIAL
_CB_POS_WHEN_READY_TO_START_LAST_MEDIA_IXN **Introduced:** GMS 8.5.200.07	Position in queue when the last dial attempt was made or the last push notification sent.	CALLBACK_FACT.POS_WHEN_LAST_DIAL
_CB_PRIORITY_WHEN_CALLBACK_ACCEPTED **Introduced:** GMS 8.5.200.07	Priority of the interaction (real or virtual) when the callback offer was accepted.	CALLBACK_FACT.PRIORITY_WHEN_CB_ACCEPTED
_CB_PRIORITY_WHEN_CUSTOMER_CONNECTED **Introduced:** GMS 8.5.200.07	Priority of the virtual interaction when the customer was connected.	CALLBACK_FACT.PRIORITY_WHEN_C_CONNECTED
_CB_PRIORITY_AT_THE_END_OF_ONLINE_WAIT **Introduced:** GMS 8.5.200.07	Priority of the virtual interaction when the customer was connected to the agent. If the customer abandoned while waiting in queue, then this value is the priority of the call when the customer disconnected.	CALLBACK_FACT.PRIORITY_WHEN_A_CONNECTED

KVP	Description	Info Mart Database Target
_CB_EWT_THRESHOLD_WHEN_OFFERED **Introduced:** GMS 8.5.200.07	Value of the EWT threshold used to decide whether the callback offer should be made or not. Pass this value as an argument of the application that is responsible for making the callback offer.	CALLBACK_FACT.EWT_THRESHOLD_WHEN_OFFERED

*This KVP must be sent twice -- as call-based attached data in a TEvent and as UserEvent-based user data.