



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.

# Genesys Engage cloud Reporting Guide

About Data Export Capability

2/24/2025

---

## Contents

- 1 About Data Export Capability
  - 1.1 What tables are included in the data export?
  - 1.2 Export views
  - 1.3 Schedule
  - 1.4 File/directory structure
  - 1.5 Target database
  - 1.6 Consumption
  - 1.7 Handling Unicode characters
  - 1.8 GDPR compliance

# About Data Export Capability

Data Export capability is enabled in select Genesys Multicloud CX deployments to periodically copy the data that is stored in the Genesys historical database (called the Info Mart database) into local .csv files, so that the data is available for further import into a data warehouse (the *target* database) for the purpose of archiving or custom reporting.

Data Export capability (also known as *BI Data Feed*) exports data from fact and dimension tables that are part of the Genesys Info Mart dimensional model and creates a .zip archive containing individual .csv files, one file per database table. The .csv files are formatted in accordance with RFC 4180 (<https://www.ietf.org/rfc/rfc4180.txt>).

The output data files are encoded using the UTF8 format.

## What tables are included in the data export?

The export does not include aggregate (RAA) tables or internal (GIDB\_\*) tables except for certain configuration tables, as listed **below**. The fact and dimension tables included in your specific data export depend on the details of your Genesys Multicloud CX agreement. The following tables are available for export:

- AGENT\_LOCATION
- ANCHOR\_FLAGS
- ATTEMPT\_DISPOSITION
- BGS\_BOT\_DIM
- BGS\_BOT\_NAME\_DIM
- BGS\_SESSION\_DIM
- BGS\_SESSION\_FACT
- BOT\_ATTRIBUTES
- BOT\_INTENT
- CALL\_RESULT
- CALLBACK\_DIAL\_RESULTS
- CALLBACK\_DIM\_1
- CALLBACK\_DIM\_2
- CALLBACK\_DIM\_3
- CALLBACK\_DIM\_4
- CALLBACK\_FACT
- CALLING\_LIST\_METRIC\_FACT
- CALLING\_LIST\_TO\_CAMP\_FACT (actualized view)
- CAMPAIGN\_GROUP\_SESSION\_FACT
- CAMPAIGN\_GROUP\_STATE
- CAMPAIGN\_GROUP\_STATE\_FACT
- CDR\_DIM1
- CDR\_FACT
- CHAT\_SESSION\_DIM
- CHAT\_SESSION\_FACT
- CHAT\_THREAD\_FACT
- COBROWSE\_END\_REASON
- COBROWSE\_FACT
- COBROWSE\_MODE
- COBROWSE\_PAGE
- COBROWSE\_USER\_AGENT
- CONTACT\_ATTEMPT\_FACT
- CONTACT\_INFO\_TYPE
- DATE\_TIME

- DIALING\_MODE
- GPM\_DIM1
- GPM\_FACT
- GPM\_MODEL
- GPM\_PREDICTOR
- GPM\_RESULT
- GROUP\_ANNEX
- GROUP\_TO\_CAMPAIGN\_FACT (actualized view)
- INTERACTION\_DESCRIPTOR
- INTERACTION\_FACT
- INTERACTION\_RESOURCE\_FACT
- INTERACTION\_RESOURCE\_STATE
- INTERACTION\_TYPE
- IRF\_USER\_DATA\_CUST\_1
- IRF\_USER\_DATA\_GEN\_1
- IRF\_USER\_DATA\_KEYS
- IXN\_RESOURCE\_STATE\_FACT
- LDR\_CAMPAIGN
- LDR\_DEVICE
- LDR\_FACT
- LDR\_GROUP
- LDR\_LIST
- LDR\_POSTAL\_CODE
- LDR\_RECORD
- MEDIA\_ORIGIN
- MEDIA\_TYPE
- MEDIATION\_SEGMENT\_FACT
- PLACE\_GROUP\_FACT (actualized view)
- POST\_CALL\_SURVEY\_DIM\_1
- POST\_CALL\_SURVEY\_DIM\_2
- POST\_CALL\_SURVEY\_DIM\_3
- POST\_CALL\_SURVEY\_DIM\_4
- POST\_CALL\_SURVEY\_DIM\_5
- POST\_CALL\_SURVEY\_DIM\_6
- RECORD\_FIELD\_GROUP\_1
- RECORD\_FIELD\_GROUP\_2
- RECORD\_STATUS
- RECORD\_TYPE
- REQUESTED\_SKILL
- REQUESTED\_SKILL\_COMBINATION
- RESOURCE\_
- RESOURCE\_ANNEX
- RESOURCE\_GROUP\_COMBINATION
- RESOURCE\_GROUP\_FACT (actualized view)
- RESOURCE\_SKILL\_FACT (actualized view)
- RESOURCE\_STATE
- RESOURCE\_STATE\_REASON
- ROUTING\_TARGET
- SDR\_ACTIVITIES\_FACT
- SDR\_ACTIVITY
- SDR\_APPLICATION
- SDR\_BOTS\_FACT
- SDR\_CALL\_DISPOSITION
- SDR\_CALL\_TYPE
- SDR\_CUST\_ATTRIBUTES
- SDR\_CUST\_ATTRIBUTES\_FACT
- SDR\_ENTRY\_POINT
- SDR\_EXIT\_POINT
- SDR\_EXT\_HTTP\_REST
- SDR\_EXT\_REQUEST
- SDR\_EXT\_REQUEST\_FACT
- SDR\_EXT\_REQUEST\_OUTCOME
- SDR\_EXT\_SERVICE\_OUTCOME
- SDR\_GEO\_LOCATION
- SDR\_INPUT
- SDR\_INPUT\_OUTCOME
- SDR\_LANGUAGE
- SDR\_MESSAGE
- SDR\_MILESTONE
- SDR\_SESSION\_FACT

- SDR\_SURVEY\_ANSWERS
- SDR\_SURVEY\_FACT
- SDR\_SURVEY\_I1
- SDR\_SURVEY\_I2
- SDR\_SURVEY\_QUESTIONS
- SDR\_SURVEY\_QUESTIONS\_I1
- SDR\_SURVEY\_QUESTIONS\_I2
- SDR\_SURVEY\_QUESTIONS\_S1
- SDR\_SURVEY\_QUESTIONS\_S2
- SDR\_SURVEY\_S1
- SDR\_SURVEY\_S2
- SDR\_SURVEY\_SCORES
- SDR\_SURVEY\_STATUS
- SDR\_SURVEY\_TRANSCRIPT\_FACT
- SDR\_USER\_INPUT
- SDR\_USER\_INPUTS\_FACT
- SDR\_USER\_MILESTONE\_FACT
- SM\_MEDIA\_NEUTRAL\_STATE\_FACT
- SM\_RES\_SESSION\_FACT
- SM\_RES\_STATE\_FACT
- SM\_RES\_STATE\_REASON\_FACT
- STRATEGY
- TECHNICAL\_DESCRIPTOR
- TIME\_ZONE
- USER\_DATA\_CUST\_DIM\_1
- USER\_DATA\_CUST\_DIM\_2
- USER\_DATA\_GEN\_DIM\_1
- USER\_DATA\_GEN\_DIM\_2
- WORKBIN

The data export for Genesys Multicloud CX customers will also include custom user data tables named IRF\_USER\_DATA\_CUST\_01 (a fact table for high-cardinality attached data) and USER\_DATA\_CUST\_DIM\_01 through USER\_DATA\_CUST\_DIM\_08 (dimension tables for low-cardinality attached data). These tables may be empty, depending on the Genesys Info Mart configuration enabled by your Genesys Multicloud CX agreement.

In addition to the data from the Genesys Info Mart dimensional model tables, configuration details data is exported from the following **GIDB tables** :

- Table GIDB\_GCX\_LOGIN\_INFO
- Table GIDB\_GC\_ANNEX
- Table GIDB\_GC\_CALLING\_LIST
- Table GIDB\_GC\_CAMPAIGN
- Table GIDB\_GC\_FOLDER
- Table GIDB\_GC\_GROUP
- Table GIDB\_GC\_LOGIN
- Table GIDB\_GC\_PLACE
- Table GIDB\_GC\_SKILL
- Table GIDB\_GC\_TENANT

## Export views

Genesys Info Mart exports your data from *export views*, which represent a frozen snapshot of the Info Mart schema at the time the export views were created. For Genesys CX on AWS customers, export

views are created when the Data Export feature is enabled for your account. Using export views means that the export will always include the same tables and columns, regardless of any schema changes that may occur as a result of Genesys Info Mart upgrades and database migrations.

The export views include all the tables listed [above](#), including the custom user-data tables.

Using export views frees you from the need to continually update your target database and consumption queries to ensure consistency with a migrated Info Mart schema. For example, without export views, new columns added to a table that gets exported would break an import query that selects all columns from the exported table, unless you have also added the corresponding columns in the target database.

Be aware that using export views means that the export will not reflect *any* changes that may have occurred in the Info Mart schema since the export views were created, including deletion or renaming of tables or columns, which might affect how Genesys Info Mart populates certain data.

Starting with release 8.5.116.12, for internal reasons the output package includes the **make\_export\_views<db-schema>.sql** script that was used to create your export views. The script is potentially of limited value to you for troubleshooting purposes.

You can request that Genesys personnel update your export views, if you identify that a later version of Genesys Info Mart, configured in accordance with your Genesys Multicloud CX agreement, provides new data that you want to be included in your export. For information about Info Mart schema changes between releases, see [New in the Info Mart Database](#) and [Summary of Info Mart Schema Changes](#).

### Important

Before your export views are refreshed, ensure that your consumption queries and target database are ready to process the additional data. For information about creating or updating your target database schema, see [Target database](#), below.

## Schedule

By default, the export runs at 00:20, 08:20, and 16:20 every day. While Genesys personnel can adjust the schedule as necessary for your cloud deployment, the export schedule should not be any more frequent than every 30 minutes.

Starting with release 8.5.116.20, you can optionally choose to encrypt the Data Export files. Contact your account representative for more information.

## File/directory structure

The export is incremental and uses special audit keys to identify changes in data since the last export. At each export, a chunk of exported data is written into a separate folder that is named

according to the following naming convention: `export_XXX`

where XXX consists of:

- an audit key identifier (*audit key high-water mark*)
- the maximum date of data contained in all previous exports and this export, in GMT time zone, written in the `YYYY_MM_DD_HH_MI_SS` format.

The output folder contains several `.zip` files, as follows:

- **export\_XXX.zip** — zip file with exported data. Each table is stored in a separate file with a file name in the format `<table-name>.csv`—for example, **interaction\_fact.csv**. Within a `.csv` file, a header line identifies the table column names. Note that, within the exported `.csv` files, nulls and empty strings are represented as empty fields. If the Data Export is encrypted, the file is a compressed binary named **export\_XXX.zip.bin**.
- **export\_XXX.zip.sha1** — checksum for **export\_XXX.zip**. The checksum can be validated by `sha1sum` program (<https://en.wikipedia.org/wiki/Sha1sum>) and is used to verify that the `.zip` file is complete on the receiving side. If the Data Export is encrypted, the checksum is **export\_XXX.zip.bin.sha1**.
- **export\_XXX.extracted.xml** — metadata about **export\_XXX.zip**.

Starting with release 8.5.116.12, the output folder also contains RDBMS-specific **sql\_scripts/<dbms>** folders, containing the **update\_target** and **make\_export\_views** SQL scripts described on this page.

### Important

The subfolder **.gim** is reserved for internal use.

Checksums are also generated for each individual table `.csv` file. If a table does not have any changes since the last export, nothing is written for that table.

## Export metadata file

[Link to video](#)

The **export\_XXX.extracted.xml** metadata file includes information about the export file, as shown in the example below.

### Example

```
<info>
<created-ts>1521091600</created-ts>
<gim-schema-version>8.5.009.15</gim-schema-version>
<gim-version>8.5.009.20</gim-version>
<hwm-from audit-key="13" created-ts="1520919983"/>
<hwm-to audit-key="200074" created-ts="1520995485"/>
<max-data-ts>1521006157</max-data-ts>
</info>
```

Where:

- `created-ts` — The UTC timestamp, in seconds since January 1, 1970, for the execution of the export.
- `gim-schema-version` — The version of the Info Mart database schema used to populate the tables; this schema version is not necessarily the same as the schema version reflected by the export views and actually used for the export.
- `gim-version` — The version of Genesys Info Mart Server that created the export files.
- `hwm-from` — The starting point of the data in the export by audit key and the create time, in UTC seconds, of that audit key.
- `hwm-to` — The ending point of the data in the export by audit key and the create time, in UTC seconds, of that audit key.
- `max-data-ts` — The maximum time, in UTC seconds, of the data contained in all previous exports and this export.

The `hwm-to` and `hwm-from` values must match between successive export runs. Use them to verify that no intermediate export file has been missed on the receiving side. For example, the next export following the example `.xml` file above is supposed to have `hwm-from audit-key = 200074`.

The maximum time span of data in any single export file is one day. For example, if historical reporting was not available for two days (because, for instance, the server or database has been down), the export will continue from the last exported high-water mark and move ahead one day in the data. The next export will continue from there, exporting no more than one day at a time, until the export has caught up with the current data.

## Target database

Genesys provides an SQL script, **`update_target_gim_db.sql`**, to assist you in creating a target schema into which to import the exported Info Mart data. Starting with release 8.5.116.12, Genesys Info Mart provides a full suite of **`update_target_gim_db*.sql`** scripts — **`update_target_gim_db.sql`**, **`update_target_gim_db_partitioned.sql`**, **`update_target_gim_db_multilang.sql`**, or **`update_target_gim_db_multilang_partitioned.sql`** — in RDBMS-specific **`sql_scripts/<dbms>`** folders in the Data Export output package. The scripts match the Genesys Info Mart release in effect when the data export was performed. Execute the script against your target database to create a schema consistent with the Info Mart schema.

You can also use the script to migrate your target database if the Info Mart database schema changes after you have set up your target database, and your export views have been updated to include the schema changes. The **`update_target_*.sql`** script enables you to migrate your target database directly from any Info Mart schema version to any later schema version, by updating the target schema with new tables or columns if they are missing.

When to run the **`update_target_*.sql`** script to migrate your target schema following an Info Mart migration depends on your business needs, import processing, and consumption queries. If you decide that you want your export to include new data available in the Info Mart database, first prepare your processing updates to accommodate the changes between the Info Mart versions (see above for links to information about schema changes). When you are ready, co-ordinate with your Genesys Multicloud CX account representative to get your export views updated and, if your deployment uses a Genesys Info Mart release earlier than 8.5.116.12, to obtain the applicable **`update_target_gim_db.sql`** script. Execute the script to migrate your target database before you try to import data from the first export after your export views have been updated.



### Custom user-data tables

In releases earlier than 8.5.011.14, the **update\_target\_gim\_db.sql** script created custom user-data tables named IRF\_USER\_DATA\_CUST\_1, USER\_DATA\_CUSTOM\_DIM\_1, and USER\_DATA\_CUSTOM\_DIM\_2. Starting with release 8.5.011.14, the **update\_target\_gim\_db.sql** script creates or migrates custom user-data tables named IRF\_USER\_DATA\_CUST\_01 and USER\_DATA\_CUST\_DIM\_01 through USER\_DATA\_CUST\_DIM\_08. Starting with release 8.5.014.19, to increase backward compatibility the script creates or migrates custom user-data tables with both forms of the table names.

### Considerations for cross-platform export and import

Genesys Info Mart supports data export and import across RDBMS platforms. For example, you can export data from a PostgreSQL Info Mart database and import it into a Microsoft SQL Server database.

If you are importing data into a Microsoft SQL Server database, note the following considerations:

- **Case-sensitive vs. case-insensitive data** — Starting with release 8.5.015.07, Genesys Info Mart supports either a case-sensitive or a case-insensitive collation for the Info Mart database in Microsoft SQL Server deployments. (In earlier releases, Genesys Info Mart required a case-insensitive collation.) Starting with release 8.5.014.34, the **update\_target\_\*.sql** script for a Microsoft SQL Server target database schema is compatible with a case-sensitive Microsoft SQL Server collation. If you plan to import case-sensitive data from an Oracle or PostgreSQL Info Mart database into a Microsoft SQL Server target database, ensure that your target database is case-sensitive.

#### Important

Data exported from your Info Mart database in the cloud is case-sensitive.

- **Index size limitation** — Remember that a Microsoft SQL Server limitation restricts the maximum length of index keys. In releases earlier than 8.5.014.34, the sizes of many columns in dimension tables in the target database schema defined for Microsoft SQL Server were reduced, to ensure that indexes did not exceed Microsoft SQL Server size limits. Starting with release 8.5.014.34, the sizes of all columns defined in the **update\_target\_\*.sql** scripts are the same across all RDBMS platforms.

For a list of the dimension columns that were modified in the **update\_target\_\*.sql** scripts for Microsoft SQL Server, see [New in Release 8.5.014.34](#).

#### Important

If you are importing Info Mart data into a Microsoft SQL Server database, ensure that your import tool or process is able to handle errors that arise when the sum of the actual values of dimension table columns included in an index exceeds the Microsoft SQL Server limit on index size.

## Consumption

The exported table data typically contains a mix of created and updated rows. For this reason, you should merge newly exported data with existing data loaded from prior exports. For example, first, load the export files into a temporary table and then use an SQL merge statement based on the primary key for the table to merge the data into a permanent target table that holds the cumulative data from prior exports.

Process the export folders in order by folder name.

If necessary, you can restart the export data stream from the beginning or from a fixed date. Also, you can re-export a time span backwards from the most recent export.

## Data decoding

The data is exported into .csv files that are formatted in accordance with RFC 4180 (<https://www.ietf.org/rfc/rfc4180.txt>). The exported data must be decoded properly before it is imported into the target database for custom reporting or archiving purposes. Customers should perform decoding of the exported .csv files according to the guidelines in RFC 4180. Properly decoded data is expected to fit into the target schema that is created by Genesys-provided scripts without the need to increase field sizes.

## Handling Unicode characters

Special considerations are required for data that includes Unicode characters. By default, Genesys Info Mart encodes the exported data using Eight-bit Unicode Transformation Format (UTF8) character encoding. However, both the original Info Mart database and the target database must be set up to accommodate Unicode characters in respective database fields, as follows:

- For Microsoft SQL Server, specify the NVARCHAR(N) data type for the columns that store Unicode data
- For Oracle, specify **AL32UTF8** as the database character set when creating the database
- For PostgreSQL, choose the default encoding, which is **UTF8**, when creating the database

Subsequently, a special "multilanguage" version of the Genesys-provided SQL script is required to create both the Info Mart schema and a target schema with the fields that store data with Unicode characters..

Finally, when exported data is decoded before being imported into the target database, .csv file decoding must be done using UTF8 encoding.

Following the above guidelines will help to avoid issues such as data corruption due to data not being decoded properly or data import due to data length being larger than the column size in the target database.

## GDPR compliance

Genesys Info Mart data complies with General Data Protection Regulation (GDPR) requirements. However, depending on when you export Info Mart data in relation to specific GDPR requests, as well as when and how you process the exported data, your target database or retained export files might contain personally identifiable information (PII) that needs to be redacted.

You are responsible for implementing adequate processes to ensure that any PII in your exported Info Mart data is handled in accordance with GDPR requirements, including using suitable retention periods or redacting data to comply with "forget" requests.

For details about which Info Mart tables potentially contain PII, see the description of the [CTL\\_GDPR\\_HISTORY](#) table. (Note that this table is not included in your data export.)