CX Contact Deployment Guide

Integrating CX Contact with Genesys Historical Reporting

12/18/2025

Integrating CX Contact with Genesys Historical Reporting

This page describes the component and configuration requirements to enable historical reporting on unattempted records.

Overview: Historical Reporting on Unattempted Records

While Outbound Contact Server (OCS) reports on the outcome of all attempted records and records failed pre-dial validation checks, it does not report on records belonging to a contact suppression list for the campaign group. This comes from CX Contact. The process is as follows:

1. When the campaign group is activated, CX Contact writes all information related to unattempted records to an Elasticsearch index (it writes one Elasticsearch document for each suppressed record).

2. As part of the regular ETL cycle, Genesys Info Mart extracts the data from Elasticsearch and transforms it into Genesys Info Mart **LDR_*** tables, which you can join with OCS-sourced data on that campaign group's attempted records.

For more information about the Genesys Info Mart database tables, see the *Genesys Info Mart Physical Data Model* for your RDBMS. For more information about managing the Genesys Info Mart ETL jobs, see the *Genesys Info Mart Operations Guide*.

Defining Unattempted Records

In this context, an unattempted record refers to a record belonging to a contact suppression list. Records excluded from a campaign because of defined filtering criteria or compliance rules are not considered unattempted records in this context.

The following table summarizes the ways in which records are reported on:

Record Type	Reporting Source
Dialed/attempted records	OCS > ICON > Genesys Info Mart
Records belonging to a contact suppression list (unattempted records)	CX Contact > Elasticsearch > Genesys Info Mart
Records that failed pre-dial validation checks (unattempted records)	OCS > ICON > Genesys Info Mart

Enabling Historical Reporting on Unattempted Records

Prerequisites

The following table summarizes the minimum release requirements for the Genesys and third-party components that enable CX Contact historical reporting.

Component	Minimum release
CX Contact	9.0.000.09
Elasticsearch	6.3.1
Genesys Info Mart	8.5.012.15
ICON	8.1.514.11 (Recommended minimum for Genesys Info Mart; Required for OCS historical reporting)

Setting up Historical Reporting

To set up historical reporting of unattempted records:

1. Deploy Elasticsearch version 6.3.1. Once this is successfully deployed, CX Contact can write all required indexes to Elasticsearch. No explicit CX Contact configuration is required.

   Important

   There are index properties that contain personally identifiable information (PII) and therefore need to be considered for the EU General Data Protection Regulation (GDPR). Ensure you configure the Elasticsearch data-retention settings so that indexes are purged before 30 days.

2. Configure Genesys Info Mart to extract the CX Contact reporting data from Elasticsearch, as follows:

   1. On the **Options** tab of the Genesys Info Mart application object, create a new configuration section, called **elasticsearch-ldr0**.

   2. Add the client option. For example: **elasticsearch-ldr0/client=rest(host.domain.com)**

   3. Add the g:tenant-prefix option. For example: **elasticsearch-ldr0/g:tenant-prefix=-2115**

   Important

   Genesys expects that CX Contact reporting on unattempted records will be used to supplement existing Outbound Contact reporting sourced from OCS. Ensure that your deployment has been configured as required for Genesys Info Mart to support Outbound Contact reporting. For more information, see Enabling Reporting on Outbound Contact Activity in the *Genesys Info Mart Deployment Guide*.

Elasticsearch Index Properties

The following table describes the Elasticsearch index properties, in which CX Contact stores the data about unattempted records. Note the following:

- The **Index property** column represents the XPath term Genesys Info Mart uses to extract and map the data.

- The **Info Mart Database Target** column indicates the Info Mart database table and column to which the property is mapped.

Index property	Description	Info Mart Database Target
campaignGroupId	The DBID of the campaign group as assigned by Configuration Server.	LDR_CAMPAIGN.CAMPAIGN_GROUP_ID (referenced through LDR_FACT.LDR_CAMPAIGN_KEY)
campaignGroupName	The name of the campaign group.	LDR_CAMPAIGN.CAMPAIGN_GROUP_NAME (referenced through LDR_FACT.LDR_CAMPAIGN_KEY)
campaignTemplateName	The name of the campaign template on which the campaign group is based.	LDR_CAMPAIGN.CAMPAIGN_TEMPLATE_NAME (referenced through LDR_FACT.LDR_CAMPAIGN_KEY)
chainId	The chain identifier of the record from the contact list.	LDR_FACT.CHAIN_ID
chainN	The order of the contact list record within the chain.	LDR_FACT.CHAIN_NUMBER
clientId	The unique client identifier of the contact from the contact list.	LDR_FACT.CLIENT_ID
contact_info	The contact information (device) for the contact from the contact list.	LDR_FACT.CONTACT_INFO
contact_info_type	The type of the contact device. This field is set to one of the following values: **Valid values:** • No Contact Type • Home Phone • Direct Business Phone • Business With Extension • Mobile • Vacation Phone • Pager • Modem	LDR_RECORD.CONTACT_INFO_TYPE (referenced through LDR_FACT.LDR_RECORD_KEY)
Index property	**Description**	**Info Mart Database Target**

Index property	Description	Info Mart Database Target
	• Voice Mail • Pin Pager • E-Mail Address • Instant Messaging	
deviceAreaCode	The area code of the record from the contact list.	LDR_DEVICE.DEVICE_AREA_CODE (referenced through LDR_FACT.LDR_DEVICE_KEY)
deviceCountryCode	The country code of the record from the contact list.	LDR_DEVICE.DEVICE_COUNTRY_CODE (referenced through LDR_FACT.LDR_DEVICE_KEY)
deviceMask	The bit mask of the record from the contact list.	LDR_FACT.DEVICE_MASK
deviceStateCode	The state code (or country code) of the record from the contact list.	LDR_DEVICE.DEVICE_STATE_CODE (referenced through LDR_FACT.LDR_DEVICE_KEY)
deviceTimezone	The time zone indicated in the record from the contact list.	LDR_DEVICE.DEVICE_TIMEZONE (referenced through LDR_FACT.LDR_DEVICE_KEY)
disposition	The reason for filtering out the record from the campaign during the pre-loading phase, as reported by CX Contact.	LDR_RECORD.DISPOSITION (referenced through LDR_FACT.LDR_RECORD_KEY)
groupName	The name of the agent group or place group.	LDR_GROUP.GROUP_NAME (referenced through LDR_FACT.LDR_GROUP_KEY)
id	An identifier Genesys Info Mart generates based on the long UUID timestamp reported by CX Contact.	LDR_FACT.ID
listId	DBID of the contact list as assigned by Configuration Server.	LDR_LIST.LIST_ID (referenced through LDR_FACT.LDR_LIST_KEY)
listName	The name of the contact list.	LDR_LIST.LIST_NAME (referenced through LDR_FACT.LDR_LIST_KEY)
postalCode	The postal code of the record from the contact list.	LDR_POSTAL_CODE.POSTAL_CODE (referenced through LDR_FACT.LDR_POSTAL_CODE_KEY)
recordId	The identifier of the record from the contact list.	LDR_FACT.RECORD_ID
recordStatus	The status of the record from the contact list. This field is set to one of the following values: **Valid values:**	LDR_RECORD.RECORD_STATUS (referenced through LDR_FACT.LDR_RECORD_KEY)
Index property	**Description**	**Info Mart Database Target**

Index property	Description	Info Mart Database Target
	• No Record Status • Ready • Retrieved • Updated • Stale • Cancelled • Agent Error • Chain Updated • Missed Callback • Chain Ready	
recordType	The type of the record from the contact list. This field is set to one of the following values: **Valid values:** • No Record Status • Ready • Retrieved • Updated • Stale • Cancelled • Agent Error • Chain Updated • Missed Callback • Chain Ready	LDR_RECORD.RECORD_TYPE (referenced through LDR_FACT.LDR_RECORD_KEY)
timestamp_iso8601	The timestamp when the event regarding the suppressed contact list records was generated by CX Contact.	LDR_FACT.START_DATE_TIME_KEY
Index property	**Description**	**Info Mart Database Target**

Elasticsearch Index Fields

The following seven sections describe the seven types of Elasticsearch index fields. Each record

represents the Elasticsearch data shown in the corresponding CX Contact Analytics Reporting panel.

	Job Record
	Call List Loading Record
	Preloading Record
	Campaign Group Event Record
	Call Result Record
	Contact History Record
	SMS/EMAIL Record
	User Actions Record

Job Record (cxc-job-*)

Field	Type
id	keyword
parentid	keyword
@timestamp	date
@endtime	date
ccid	keyword
type	keyword
name	keyword
state	keyword
result	keyword
created	date
started	date
finished	date
duration	integer
error	text
errorCode	integer
trace	keyword
component	keyword

Field	Type
version	keyword
hostname	keyword
address	keyword

Call List Loading Record (cxc-didr-*)

Field	Type
id	keyword
@timestamp	date
type	keyword
jobid	keyword
jobts	date
importfile	keyword
line	integer
mappingfile	keyword
ccid	keyword
listid	integer
listTableName	keyword
listName	keyword
customTZMap	boolean
chain_id	integer
chain_n	integer
contact_info	keyword
deviceDigits	text
defaultRegion	keyword
deviceIndex	short
accepted	byte
error	keyword
e164	keyword
countryCode	keyword
areaCode	keyword
exchange	keyword
restOfNumber	keyword
maskValue	long
tzuid	integer
state_code	keyword
country_code_iso	keyword

Field	Type
mask	object

Preloading Record (cxc-contact-*)

Field	Type
id	keyword
@timestamp	date
ccid	keyword
calluuid	keyword
contact_info	keyword
contact_info_type	keyword
contact_id	keyword
chain_id	integer
chain_n	integer
callTime	date
callResult	keyword
dialingMode	keyword
optimizationGoal	integer
optimizationMethod	keyword
listName	keyword
listid	integer
campaignName	keyword
campaignGroupName	keyword
sessionuuid	keyword
campaignTemplateName	keyword
groupName	keyword
agentLoginId	keyword
disposition	keyword
successful	boolean
userData	object

Campaign Group Event Record (cxc-cgevent-*)

Field	Type
id	keyword
@timestamp	date
ccid	keyword

Field	Type
sessionuuid	keyword
action	keyword
state	keyword
dialingMode	keyword
optimizationParameter	integer
optimizationType	keyword
campaignName	keyword
campaignGroupName	keyword
campaignGroupDBID	keyword
campaignTemplateName	keyword
groupName	keyword
actualBusyFactor	float
actualHitRatio	float
actualOverdialRate	Float
actualTimeToComplete	integer
lists	object

Call Result Record (cxc-crr-*)

Field	Type
id	keyword
@timestamp	date
@endtime	date
ccid	keyword
calluuid	keyword
contact_info	keyword
contact_info_type	keyword
blockingRuleName	keyword
duration	integer
durationCall	integer
durationACW	integer
durationCPD	integer
durationQueue	integer
timeDialing	date
timeClientRinging	date
timeBadCallReleased	date
timeClientPickedUp	date

Field	Type
timeCPDFinished	date
timeQueued	date
timeAgentRinging	date
timeAgentEstablished	date
timeAMDiverted	date
timeAbandoned	date
timeAgentCallReleased	date
callTime	date
callResult	keyword
dialingMode	keyword
optimizationGoal	integer
optimizationMethod	keyword
listName	keyword
campaignName	keyword
campaignGroupName	keyword
sessionuuid	keyword
campaignTemplateName	keyword
groupName	keyword
timezoneName	keyword
timezoneNameCME	keyword
timezoneOffset	integer
agentLoginId	keyword
scheduledTime	date
recordType	keyword
recordStatus	keyword
voiceTransferDestination	keyword
countryCode	keyword
clientCountryCode	keyword
areaCode	keyword
deviceTimezone	keyword
disposition	keyword
postalCode	keyword
userData	object

Contact History Record (cxc-ldr-*)

Field	Type
id	keyword
@timestamp	date
ccid	keyword
campaignName	keyword
campaignId	integer
campaignGroupName	keyword
campaignGroupId	integer
campaignTemplateName	keyword
campaignTemplateId	integer
groupName	keyword
groupId	integer
blockingRuleName	keyword
blockingRuleId	integer
listName	keyword
listId	integer
recordId	integer
clientId	keyword
chainId	integer
chainN	integer
contact_info	keyword
contact_info_type	keyword
recordType	keyword
recordStatus	keyword
deviceCountryCode	keyword
deviceAreaCode	keyword
deviceStateCode	keyword
deviceTimezone	keyword
deviceMask	integer
postalCode	keyword
disposition	keyword
reason	keyword
customFields	object
timestamp_iso8601	date

✉ SMS/EMAIL Record (cxc-nexdr-*)

Field	Type
id	keyword
@timestamp	date
ccid	keyword
mediaType	keyword
calluuid	keyword
contact_info	keyword
clientId	keyword
chainId	integer
chainN	integer
from	keyword
subject	keyword
listName	keyword
campaignName	keyword
groupName	keyword
campaignGroupName	keyword
campaignTemplateName	keyword
sessionuuid	keyword
messageID	keyword
batchID	keyword
status	keyword
deliveryReceipt	keyword
disposition	keyword
callResult	keyword
errorCode	integer
errorMessage	keyword
timeReceivedFromOCS	date
timeSubmittedToNexus	date
timeResponseReceived	date
timeOCSNotified	date
timeConsumerResponded	date
optout	boolean
userData	object

![icon] User Actions Record (cxc-audit-*)

Field	Type
id	keyword
requestID	keyword
@timestamp	date
userName	keyword
@endtime	date
duration	integer
action	Keyword
actionDetails	keyword
objectType	keyword
objectSubtype	keyword
objectName	keyword
objectID	integer
apicall	boolean
successful	boolean
errorMessage	text
details	text
endPoint	text
changeSet	object

Elasticsearch Maintenance Recommendations

To help you better manage your indexes and snapshots and to prevent too many indexes from creating an overflow of shards, it is recommended that you set up a scheduled execution of Elasticsearch Curator:

1. Delete indexes older than 60 days according to the index name and mask.

   - cxc-job-*

   - cxc-audit-*

   - cxc-crr-*

   - cxc-didr-*

   - cxc-ldr-*

   - cxc-nexdr-*

   - cxc-cgevent-*

   - cxc-contact-*

2. Make a snapshot of each index.

    -   cxc-analytics-*

Elasticsearch Index Purge

From the version 100.0.030.0003 onwards, CX Contact creates monthly ElasticSearch indices instead of daily indices. Monthly indices follow a different file format compared to daily indices.

Here are a few examples of daily indices that include the date/day as well in the file name:

-   `cxc-didr-123-2022.07.01`
-   `cxc-didr-123-2022.07.02`
-   `cxc-didr-123-2022.07.03`

Note that the numbers 123 in the above examples denote the tenant number configured during provisioning.

Monthly indices, however, follow a different file naming convention and do not include the date. Here's an example for an index for the month of August:

-   `cxc-didr-123-2022.08`

So, when you delete a single index file, you will lose a month's data instead of a specific day's. So, it is important to reconfigure ElasticSearch curator so that data is correctly purged.

> **Important**
>
> cxc-analytics-* indexes do not have a timestamp in their name and must not be deleted. Deleting a cxc-analytics-* index will result in the loss of all CX Contact Analytics Dashboard customizations.

For example, the following configuration would purge index files whose creation date is older than 90 days:

```
filters:
filtertype: age
source: creation_date
direction: older
unit: days
unit_count: 90
```

It is recommended to change the purge duration to months as the index creation is also on a monthly basis. In order to ensure that index data from last 90 days is available, you must change the `unit` to `months` and `unit_count` to 4.

```
filters:
filtertype: age
source: creation_date
direction: older
unit: months
unit_count: 4
```