Genesys Info Mart Private Edition Guide

Upgrade, roll back, or uninstall Genesys Info Mart

12/17/2025

Contents

Learn how to upgrade, roll back, or uninstall Genesys Info Mart.

Related documentation:
-
-
-
-

RSS:
- For private edition

> **Important**
>
> The instructions on this page assume you have deployed the services in service-specific namespaces. If you are using a single namespace for all private edition services, replace the namespace element in the commands on this page with the name of your single namespace or project.

Supported upgrade strategies

Genesys Info Mart supports the following upgrade strategies:

Service	Upgrade Strategy	Notes
Genesys Info Mart	Rolling Update	

For a conceptual overview of the upgrade strategies, refer to Upgrade strategies in the Setting up Genesys Multicloud CX Private Edition guide.

In general, upgrades for the three Genesys Info Mart services are released at the same time, but this is not necessarily the case. Note that the respective release packages have different release numbers.

As long as you remain within the range of N-2 releases for each service, the Genesys Info Mart services can inter-operate with one another. However, if upgraded releases for all the Genesys Info Mart services are available, Genesys recommends that you upgrade them all during the same upgrade cycle.

Upgrading GIM might or might not require migrating the Info Mart database as well, depending on whether there are schema changes. If database migration is required, the upgraded GIM service automatically launches the migration job. GIM upgrade and database migration do not interfere with aggregation or other queries in progress. Robust backward compatibility means that an upgraded or

rolled-back GIM service can operate with either a migrated or a nonmigrated Info Mart database, even with a half-migrated database if the migration job did not complete successfully.

Timing

A regular upgrade schedule is necessary to fit within the Genesys policy of supporting N-2 releases, but a particular release might warrant an earlier upgrade (for example, because of a critical security fix).

If the service you are upgrading requires a later version of any third-party services, upgrade the third-party service(s) before you upgrade the private edition service. For the latest supported versions of third-party services, see the Software requirements page in the suite-level guide.

Scheduling considerations

Genesys recommends that you upgrade the services methodically and sequentially: Complete the upgrade for one service and verify that it upgraded successfully before proceeding to upgrade the next service. If necessary, roll back the upgrade and verify successful rollback.

Do not attempt to upgrade a service unless all the Genesys Info Mart services are operating normally.

Upgrading the services potentially affects the availability of Info Mart data. Plan a time when this eventuality is least disruptive to your operations.

If you are upgrading all the services during the same upgrade cycle, upgrade GIM first, then GSP, then GCA.

Monitoring

Monitor the upgrade process using standard Kubernetes and Helm metrics, as well as service-specific metrics that can identify failure or successful completion of the upgrade (see Observability in Genesys Info Mart).

Genesys recommends that you create custom alerts for key indicators of failure — for example, an alert that a pod is in pending state for longer than a timeout suitable for your environment. Consider including an alert for the absence of metrics, which is a situation that can occur if the Docker image is not available. Note that Genesys does not provide support for custom alerts that you create in your environment.

Preparatory steps

Ensure that your processes have been set up to enable easy rollback in case an upgrade leads to compatibility or other issues.

Each time you upgrade a service:

1. Review the release note to identify changes.

2. Ensure that the new package is available for you to deploy in your environment.

3. Ensure that your existing **-values.yaml** file is available and update it if required to implement changes.

4. Review the Reporting and Analytics Aggregates (RAA) and Genesys CX Insights (GCXI) release notes and release advisories, for information about the possible impact of Genesys Info Mart migration on aggregation, as well as workarounds or additional steps to take during Genesys Info Mart migration.

In addition, if you are upgrading the GIM service:

1. Review the "New in the Info Mart Database" and "Summary of Info Mart Schema Changes" pages in the *Genesys Info Mart Physical Data Model*, to identify whether there are schema changes that will necessitate migration of the Info Mart database as part of a GIM upgrade. If so, and if you use the Data Export capability, decide whether you want to update your export views to include new data available in the migrated database.

2. If you do want to update your export views, do so after the GIM upgrade is complete (see Update export views).

3. For reference purposes, identify and make notes of any custom changes that you made to the Info Mart database—for example, table spaces, partitions, additional indexes, views, or permissions.
   Database migration might involve creating new tables or replacing some tables with views. You will need to re-create any custom database objects or permissions that become lost or invalidated during the update process.

4. Ensure that a recent backup of the Info Mart database is available in case you need to roll back a database migration.

Rolling Update

Rolling Update: Upgrade

Execute the following command to upgrade :

```
helm upgrade --install  -f -values.yaml  -n
```

Tip: If your review of Helm chart changes (see Preparatory Step 3) identifies that the only update you need to make to your existing **-values.yaml** file is to update the image version, you can pass the image tag as an argument by using the ``--set`` flag in the command:

```
helm upgrade --install  -f -values.yaml  --set .image.tag=
```

When you upgrade GIM and GSP, increase the Helm timeout value from the default 5 minutes to 15 minutes or longer. For GIM, you need a longer timeout to accommodate possible database migration, which can take a significant amount of time if, for example, new indexes are being added. For GSP, you need to allow sufficient time for GSP to run the process to create a pre-upgrade hook, to ensure that state information is correctly preserved when the previous instance shuts down.

Examples

- GIM and GIM Monitoring:

```
   helm upgrade --install gim gim-100.0.116+2900.tgz -f gim-values.yaml -n gim --timeout 15m
    helm upgrade --install gim-monitoring gim-monitoring-100.0.116+2900.tgz -n gim
```

- GSP:

```
  helm upgrade --install gsp gsp-100.0.100+1400.tgz -f gsp-values.yaml -n gsp --timeout 15m
```

- GCA and GCA Monitoring:

```
     helm upgrade --install gca gca-100.0.100+2300.tgz -f gca-values.yaml -n gca
      helm upgrade --install gca-monitoring gca-monitoring-100.0.100+2300.tgz -n gca
```

Rolling Update: Verify the upgrade

Follow usual Kubernetes best practices to verify that the new service version is deployed. See the information about initial deployment for additional functional validation that the service has upgraded successfully.

Monitor the operations dashboard for the status of services and job execution. Genesys Info Mart does not report the Ready state for a pod until the pod is operational. The GIM service does not report the Ready state until the first post-upgrade transformation job has completed successfully.

To verify that an upgraded GIM is successfully populating the Info Mart database, use the **gim_kafka_timestamp_behind** metric to track data latency and validate that there is no growing backlog. To validate successful database migration, you can query the database directly to verify that schema changes (for example, new tables or columns) have been implemented and the new database objects are being populated.

If Genesys Info Mart is operating normally after an upgrade in which there were schema changes, but the migration job either did not run or else did not complete successfully, a transitory condition (for example, losing connection to the database or exceeding the Helm timeout for executing the job) was the likely cause of the migration job failure. Simply restart the GIM pod. The GIM service detects the out-of-date schema and restarts the migration job, which picks up from where it left off.

If a new pod does not start and/or operate successfully, Genesys recommends that you roll back to the previous version, so that Genesys Info Mart can continue operating while you investigate the reasons for failure. Rolling back the GIM service does not affect the Info Mart database.

If there are issues with a third-party dependency, such as Kafka or PostgreSQL, then rollback will not work until you resolve the third-party issue.

Contact Genesys Customer Care if you cannot resolve an upgrade issue.

Rolling Update: Rollback

Execute the following command to roll back the upgrade to the previous version:

`helm rollback`

or, to roll back to an even earlier version:

`helm rollback`

Alternatively, you can re-install the previous package:

1. Revert the image version in the `.image.tag` parameter in the **-values.yaml** file. If applicable, also revert any configuration changes you implemented for the new release.

2. Execute the following command to roll back the upgrade:

```
helm upgrade --install  -f -values.yaml
```

> **Tip:** You can also directly pass the image tag as an argument by using the `--set` flag in the command:
>
> ```
> helm upgrade --install  -f -values.yaml  --set .image.tag=
> ```

When you roll back GIM and GSP, increase the Helm timeout value from the default 5 minutes to 15 minutes or longer.

Examples

- GIM:

```
helm rollback gim --timeout 15m
```

- GSP:

```
helm rollback gsp --timeout 15m
```

- GCA:

```
helm rollback gca
```

Rolling Update: Verify the rollback

Verify the rollback in the same way that you verified the upgrade (see Rolling Update: Verify the upgrade).

Post-upgrade procedures

No routine post-upgrade procedures are required, but you might want to update your export views for Data Export if there were schema changes.

Update export views

If you use the Data Export feature, your data will continue to be exported using the export views in effect before GIM was upgraded, regardless of whether the upgrade was associated with schema changes that triggered migration of the Info Mart database.

If you want your exported data to reflect the schema changes (for example, new tables or columns), you must update your export views. Execute the following command, which forces GIM to rerun the migration job with the **make-export-views** parameter:

```
kubectl exec  -n  -- ./entrypoint.sh -job Job_MigrateGIM -make-export-views
```

You might also need to update your target schema and change your import processing to

accommodate the Info Mart schema changes. For more information, see About Data Export.

Uninstall

> ### Warning
> Uninstalling a service removes all Kubernetes resources associated with that service. Genesys recommends that you contact Genesys Customer Care before uninstalling any private edition services, particularly in a production environment, to ensure that you understand the implications and to prevent unintended consequences arising from, say, unrecognized dependencies or purged data.

Execute the following command to uninstall :

```
helm uninstall  -n
```

For example:

```
helm uninstall gsp -n gsp
```

Uninstalling the Genesys Info Mart services can be useful if, say, you have completed a beta testing program and want to clean up your environment and free up resources.

Uninstalling any of the Genesys Info Mart services does not affect the Info Mart database or any files on the S3-compatible storage location GIM uses for Data Export.

Uninstalling GIM does not affect aggregation that may be in process and does not directly affect GCXI reports. However, of course, the Info Mart database on which GCXI reporting is based will no longer be updated.

Uninstalling GSP does not remove all state-related Kubernetes resources (ConfigMaps) referring to files on the S3-compatible storage that GSP uses during processing. If you are uninstalling GSP preparatory to re-installing it, execute the following command to clear these resources:

```
kubectl delete cm --selector='app='
```

where is the service name calculated by the **gsp.fullname macro**. If you did not override the default release name, then is gsp.