



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.

## Setting up Genesys Multicloud CX Private Edition

[Setting up a CD pipeline](#)

---

## Contents

- 1 CD Pipeline for Genesys Multicloud CX private edition
- 2 Prerequisites
- 3 Procedure
- 4 Frequently asked questions
  - 4.1 What repository will Genesys use to provide Helm charts to customers?
  - 4.2 How many Helm charts are packaged for a Genesys Multicloud CX service?
  - 4.3 How do you solve dependencies between different Genesys components during deployment?
  - 4.4 Will introducing a new component or Helm parameter affect the existing ecosystem of services?

---

Provides recommendations on setting up a Continuous Deployment (CD) pipeline in a your cloud private edition infrastructure for automated deployments.

## Related documentation:

- 
- 

## RSS:

- [For private edition](#)

# CD Pipeline for Genesys Multicloud CX private edition

Genesys delivers its artifacts in the JFrog Artifactory Edge repository. You can pull the containers from JFrog and push it into your pipeline stream for automated deployments. Genesys strongly recommends automating deployments for its services via a CD pipeline.

## Prerequisites

The tools you use in your CD pipeline must execute Helm charts as part of a pipeline.

## Procedure

Here is a quick overview of the pipeline steps involved in a typical CD environment.

1. Download a container from the Genesys repository.
2. Push the downloaded container into your internal container registry or to a quarantine location for security scans.
3. Perform security scans on the container.
4. If you are confident with the scan results, promote the container to a test/pre-production environment.
5. In the test/pre-production environment, upgrade the container by referring the upgrade procedure of the specific service in its service-level documentation. You can also check out the high level upgrade procedure in this guide.
6. Test the updated environment by running the automated tests.
7. Once the test results are satisfying, promote the container to the next environment (if applicable to your organization) for further validation before moving to production.

## Important

Your organization might have different environments other than the pre-production environment in order to test a new version of a container rigorously. Therefore, promoting a container to the next environment could mean a different environment for some users and production environment for other users.

8. Upgrade the container in the production environment by referring the upgrade procedure of the specific service in its service-level documentation. If you encounter any issue with the upgrade, you can always rollback to the previous point before the upgrade by referring the upgrade procedure of the specific service in its service-level documentation. You can also check out the high level rollback procedure in this guide.

## Frequently asked questions

The following FAQs answer important considerations when you are planning your CI/CI pipeline.

### What repository will Genesys use to provide Helm charts to customers?

Genesys will provide continuous delivery updates in the JFrog Artifactory Edge repository used for the initial deployment, as described in [Downloading your Genesys Multicloud CX containers](#).

### How many Helm charts are packaged for a Genesys Multicloud CX service?

Genesys typically packages one Helm chart per service. However, for specific services like Genesys Web Services (GWS), you can use the same Helm chart and deploy different services by varying the values in the Helm chart.

### How do you solve dependencies between different Genesys components during deployment?

During initial deployment, we enforce a specific deployment order to be followed when you deploy Genesys Multicloud CX services. This will resolve the requirements on dependencies between different services. Once your initial deployment is up and running, you can upgrade individual services at different times.

We recommend you create a platform level CD pipeline to perform initial deployment in the required order.

### Will introducing a new component or Helm parameter affect the existing ecosystem of services?

No. You can deploy the new service by following its instructions provided the core components like GAuth, GWS, Tenant service, etc. are already deployed.

---

## Important

For additional information on pipelines and examples, refer to the Private Edition page in our public repository.