



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 Knowledge Center Deployment Guide

Knowledge Center 8.5.1

3/14/2022

Table of Contents

Genesys Knowledge Center Deployment Guide	3
New in this Release	5
What is Genesys Knowledge Center?	7
GEO Location	11
Knowledge Center Components	13
High-Level Architecture	19
Prerequisites	22
Installing the Knowledge Center Cluster Application	24
Installing Knowledge Center Server	28
Installing the Knowledge Center CMS	45
Installing and Using the Administrator Plugin	57
Installing the Pulse Plugin	70
Installing the Workspace Desktop Edition Plugin	76
Importing into the Knowledge Center Server	89
Security	94
SSL Configuration for Knowledge Center Servers	95
Transport Layer Security (TLS)	98
Authentication	109
UTF8	110
Sample UI	111
Configuring CMS Cluster	118
Load-Balancing Configuration	122
Configuration Options	125
Sizing	153

Genesys Knowledge Center Deployment Guide

What's New

Find out what's new in this release of Genesys Knowledge Center Deployment.

[New in this release](#)

Getting Started

Find information to help plan your Genesys Knowledge Center Deployment.

[What is Genesys Knowledge Center](#)
[Genesys Knowledge Center Components](#)
[High-level architecture](#)
[Prerequisites](#)

Installing Knowledge Center

Find procedures to install and configure Genesys Knowledge Center Cluster and Server.

[Installing the Cluster Application](#)
[Installing Genesys Knowledge Center Server](#)
[Importing Data Into the Knowledge Base](#)

Installing Knowledge Center CMS

Find procedures to install and configure Genesys Knowledge Center CMS.

[Installing the Genesys Knowledge Center CMS](#)

Installing Plugins

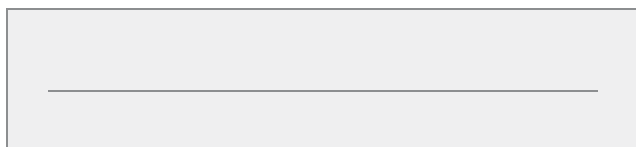
Find procedures to install and configure the Genesys Knowledge Center plug-ins.

[Installing and Using the Administrator Plugin](#)

Deploying in Production

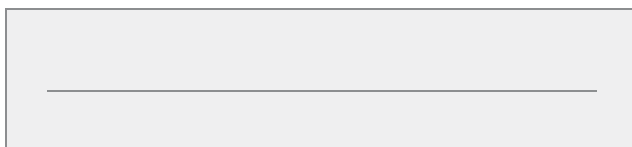
Find procedures to deploy and configure a cluster of Genesys Knowledge Center servers.

[Load-Balancing Configuration](#)



Installing the Pulse Plugin

Installing the Workspace Desktop Edition
Plugin



Security

UTF8

Configuration Options

New in this Release

This is the 8.5.1 release of Genesys Knowledge Server. Here are the latest and greatest features:

Knowledge Center Plugin for Workspace Desktop Edition

- Workspace Desktop Plugin localization for French, German, Portuguese, Spanish
- Enables agents to:
 - work with knowledge bases in multiple languages
 - view attachments within found answers
 - post comments to documents that don't match their questions, allowing documents to be improved
- Added browsing capability, allowing navigation through the content of knowledge bases
- Added support for formatted content in the documents
- Multiple minor improvements to interfaces around the agent experience for the Workspace Desktop integrations

Improved Language Support

- Content Management System (CMS) support for authoring of content in any language that is UTF-8 compliant
- NLP search available for content authored in French, German, Italian, Portuguese, and Spanish (keyword-based search is already supported for any UTF-8 compliant language.)

Security Improvements

- Mutual TLS (Transport Layer Security) for the Knowledge Center Server and the CMS.

Pulse Reporting Improvements

- Expanded dashboards in Pulse enable deeper dives into knowledge events analysis
- GEO location tracking and reporting of questions/searches based on IP address
- Reporting API: access to data around events such as search, content viewed, content feedback and others
- Keyword clouds
- Role-Based Access Control: When you add qualifying skills to a knowledge base, only agents with those skills may access that knowledge base.
- Ability to secure Elasticsearch API from data modification requests

Integrations

- Configurable web integration with:

- Genesys Email Forms
- Genesys Web Engagement
- Genesys Web Callback

Search Tuning Enhancements

- System can use customer implicit feedback (such as navigation through result provided) to improve search precision over time
- Confidence score: Question and answer matching boosted when a customer opens an answer and keeps answer open for a long period of time

Knowledge Center CMS

- Rich text editing capabilities available for content authoring in CMS
- Ability to setup regular, automatic synchronizations of changed and approved content from CMS to Knowledge Center Server

Other Improvements

- Indexer:
 - imports documents with all allowed fields (including custom fields and attachments) from XML files
 - applies XSLT transformation to provided XML files to align structure
- Sample UI supports:
 - attachment viewing
 - rich text document viewing
 - comments for negative feedback

What is Genesys Knowledge Center?

Genesys Knowledge Center allows you to make the best use of your enterprise knowledge by capturing, storing, and distributing it wherever it is needed. Let's take a closer look at the various capabilities of Knowledge Center and some corresponding use cases.

- Knowledge-assisted Channels
- Proactive Knowledge
- Knowledge Web Search

Knowledge-assisted Channels

With Knowledge Center, you can:

- Knowledge-enable channels by providing the right answers to customers in-channel to deflect interactions, leading to cost reduction and better customer service.
 - Knowledge-assisted Email form: Find applicable support articles based on email ticket submission and web form.
- Empower agents with context-appropriate knowledge in a unified desktop for faster resolution when agent-assisted service is needed.

Use Case: Knowledge-assisted Email

Basic Flow	Outcome 1	Outcome 2
<ol style="list-style-type: none">1. Tracy clicks on an email web form to find out if GDemoTelecom has service in an area that she is moving to.2. Tracy types <i>"Do you have service in Belmont, CA?"</i> in the subject line.3. Tracy clicks out of the subject line to type the content in the message body.4. An FAQ search is invoked.	<p>Tracy is provided with the coverage map for Belmont, CA as a suggested answer.</p> <p>She provides feedback and closes the window.</p>	<p>Tracy ignores the FAQ search and types content in the message body since she has more questions.</p> <p>An email request is logged and placed in queue.</p>

Note: This use case requires customization of Web Form with Knowledge Search API.

Use Case: Knowledge-assisted Social or SMS

Basic Flow	Outcome 1	Outcome 2
<ol style="list-style-type: none">1. @tibwizz sends a Tweet (or SMS) <i>"looks like I will miss my connecting flight from LAX to SFO"</i> to @blueskyairlines.2. Interaction is created and queued.3. Orchestration script invokes Knowledge API to find answers on what to do when you miss connections.	<p>Answer found.</p> <p>@Blueskyairlines auto-responds to @tibwizz <i>"Click here to schedule a call with our travel consultant to rebook"</i>.</p>	<p>Answer not found.</p> <p>Queue the message for agent.</p>

Note: This use case requires customization of Orchestration logic.

Proactive Knowledge

- Combine Knowledge with Proactive Engagement to proactively provide suggested articles at the right moment.
- Provide knowledge-based assistance for agents if the customer asks for a human-assisted channel escalation.
- Reduce effort, reduce friction and channel escalation.

Use Case: Proactive Knowledge

Basic Flow	Outcome 1	Outcome 2	Outcome 3
<ol style="list-style-type: none">1. Jurgen browses www.Gbank.com to research <i>College Savings Plan</i>.2. He navigates to the page.3. Web Engagement Rules can trigger knowledge article lookup to provide <i>knowledge nudges</i>.	<p>Suggested Pages/Info</p> <p>Within the suggested articles section of the page, a few links are populated:</p> <ul style="list-style-type: none">• Starting a college savings plan• Transferring an existing college savings plan• College Savings Plan Calculator	<p>Jurgen ignores the suggestions.</p> <p>No action taken.</p>	<p>Jurgen looks at suggestions, but still continues to browse.</p> <p>Proactively offer customers the ability to escalate to assisted service.</p>

Note: This use case requires customization of Rules and Web Page logic.

Knowledge Web Search

Enable dynamic FAQ and channel deflection using natural language search and present knowledge articles to customers via the web.

Use Case: Contact Center Escalation

The following list of outcomes from examples on this page demonstrates how Knowledge Center allows customers to serve themselves if they want to, while providing them with easy ways to contact an agent if they cannot find what they are looking for:

- Outcome 3 in the Web Search and Proactive Knowledge examples
- Outcome 2 in the Knowledge-assisted Email example
- Outcome 3 in the Knowledge Assisted Chat example

Use Case: Web Search

Basic Flow	Outcome 1	Outcome 2	Outcome 3
<ol style="list-style-type: none">1. John recently booked an Alaskan vacation for his family on Blue Sky Airlines.2. John would like to know if he can gate check his baby's stroller and car seat.3. John goes on www.blueskyairlines.com and in the search box types <i>"can I gate check my infant car seat and stroller?"</i>	<p>One Question. One Answer.</p> <p>Knowledge Center finds the right answer in the FAQs and provides the answer to John.</p>	<p>Top 3 Answers.</p> <p>Knowledge Center also provides two other articles that contain information about gate checking guidelines.</p>	<p>John is not satisfied with the answers and says answer was not helpful.</p> <p>John is offered a choice of chat, email, or callback based on agent availability or hours of operation.</p> <p>Agent receiving John's request is presented with all the relevant information about John, his reservations, and the answers viewed by John so that he/she can quickly help John.</p>

Note: This use case requires customization of Rules and web page logic.

Use Case: Fast access to content with auto-complete

Basic Flow	Outcome 1	Outcome 2
<ol style="list-style-type: none">1. John goes online to the Blue Sky Airlines website.2. He navigates the website and finds the page for <i>Traveling with an infant</i>.3. After reviewing the page, John is not clear if he can gate check his stroller.4. John notices a Search Bar at the top of the page and types "can I gate check"5. Genesys Knowledge Center Auto-complete functionality provides suggestions like "can I gate check my infant car seat?".	<p>John finds the answers to the suggested questions helpful.</p> <p>He provides feedback and closes the window.</p>	<p>John has more questions.</p> <p>Create a chat interaction and place John in queue.</p>

Use Case: Browsing though document categories

Basic Flow	Outcome
<ol style="list-style-type: none">1. As John reads the knowledge article about gate checking his infant's car seat, he also notices a category link called <i>"Traveling with Infants"</i>.2. John clicks on the link and now has access to four other articles:<ul style="list-style-type: none">• Travel tips for parents traveling with infants• Baggage allowance for infants• Online check-in for parents traveling with kids	<p>John now has all the information he needs.</p> <p>He answers "Yes" to the feedback question from the original article, which now ranks the article higher for subsequent searches.</p> <p>Note: Feedback is not available for browsed articles, since all feedback is directly related to a search query.</p>

GEO Location

Important

Collecting information about a customer's location may be subject to regulations or restrictions within your country, please check with your national legislation to ensure you are not in violation. This feature can be turned off if needed.

The Geo-location is based on the client IP address and this client IP address and its relevant geo-location coordinates are then stored in the History index.

The Administrator is able to configure the behavior of the geo-location for the cluster (cluster/reporting/geo) as:

- off - both IP and longitude and latitude are empty for historical records
- IP - only IP address is stored
- country - IP and country longitude and latitude of country are stored
- city (default) - IP and city longitude and latitude are stored

This stored data is used in the Kibana to visualize:

- a geo-map with requests heat indicators
- the top 10 countries



Activity Heatmap

Path to the GeoIP database

Geo-location functionality requires a special database to translate an IP address to the geographical location of the customer. When Genesys Knowledge Center Server is installed it provides the database stored in **<installation directory>\linguatools\geoip folder**. The folder storing the database can be changed in the **gks.yml** file:

```
...  
path.geoip : <IP folder>/GeoIP/GeoLiteCity.dat  
...
```

To update the database download the most recent version from <https://www.maxmind.com> and overwrite the file in the database folder. The Knowledge Center server needs to be restarted to work with the new database.

Knowledge Center Components

Before you start **working with Genesys Knowledge Center**, you might find it helpful to learn about its components:

- **Knowledge Center Server**—Combines indexing and natural language-based search capabilities to provide effective knowledge article retrieval from one or more knowledge bases.
- **Knowledge Center CMS**—Provides customers who do not have an existing Content Management System (CMS) with the ability to create and update their knowledge bases and push them to the Genesys Knowledge Center Server for indexing and search. This component also allows customers to import and edit knowledge articles from a file.
- **Knowledge Center Plugin for Administrator**—Enables system administrators to use Genesys Administrator to configure their knowledge clusters.
- **Knowledge Center Plugin for Pulse**—Allows contact center managers to view Genesys Knowledge Center reporting at near real-time from the Pulse user interface.
- **Knowledge Center Plugin for Workspace Desktop Edition**—Provides agents with access to knowledge events (searches, article views and feedback) related to the current customer and also allows them to search the knowledge base right from their desktop.
- **Knowledge Center Data Import Tool**—Use this tool to import XML-based QNA data into a Knowledge Center index.
- **Knowledge Center REST API**—Can be used for both client and management functions.
- **Genesys Web Engagement Integration**—Knowledge Center can be used with GWE to provide proactive engagement capabilities.

Knowledge Center Server

The Genesys Knowledge Center Server combines indexing and search capabilities that allow for effective FAQ retrieval over one or more knowledge bases. It is web-based, and can run under the **Jetty** HTTP Server.

At its core Knowledge Center Server consists of two key parts:

- The **Elasticsearch** search and analytics engine
- Several Elasticsearch plugins

Elasticsearch is a search server based on **Lucene**. It provides a distributed, multi-tenant-capable full-text search engine with a RESTful web interface and schema-free JSON documents. Elasticsearch is distributed, which means that indices can be divided into shards and each shard can have zero or more replicas. Each node hosts one or more shards, and acts as a coordinator to delegate operations to the correct shards.

Other Features of the Knowledge Center Server

- Knowledge Center Server exposes a **REST API** that can be used for both client and management functions.
- Knowledge Center Server is a cluster application, meaning that several nodes or servers can be grouped within a single cluster.
- Knowledge Center Server requires two application objects in Genesys Administrator:
 - One to describe the server itself (type = *Genesys Knowledge Center Server*)
 - Another for storing high-level options and knowledge base configurations, and for integrating the Knowledge Center server with other applications (type = *Application Cluster*)
- You can use third-party load-balancers above the cluster to organize your servers into a single pool, thereby providing a single point of entry for your users.
- Knowledge Center Server uses **Genesys Roles** to restrict access, and to authorize and authenticate users.
- The Knowledge Center installation package includes a launcher that can launch both Jetty and all of the applications deployed on Jetty as a standalone Genesys application. To accomplish this goal, the launcher communicates with the Genesys Config Server to fetch the required options.

Knowledge Center CMS

The Knowledge Center Content Management System (CMS) serves several purposes:

- Creates, activates, and deactivates knowledge bases
- Creates, updates, and deletes questions and answers in a knowledge base
- Assigns categories to this content
- Imports historical information from the Knowledge Center Server

The CMS primarily interacts with the Knowledge Center Server when creating or updating index data.

Plugin for Administrator

This plugin lets you manage the structure of the knowledge bases that are controlled by the Knowledge Center Server Cluster application object in Genesys Administrator.

After you install this plugin, you will have access to a separate page in Administrator that displays a user interface for creating new knowledge bases and for editing the descriptions, options, languages, and custom fields in existing knowledge bases.

Plugin for Pulse

The Knowledge Center Plugin for Pulse displays Knowledge Center Server statistics, such as KPIs, user activity, trending topics, like and dislike trends, types of activities, and more.

Important

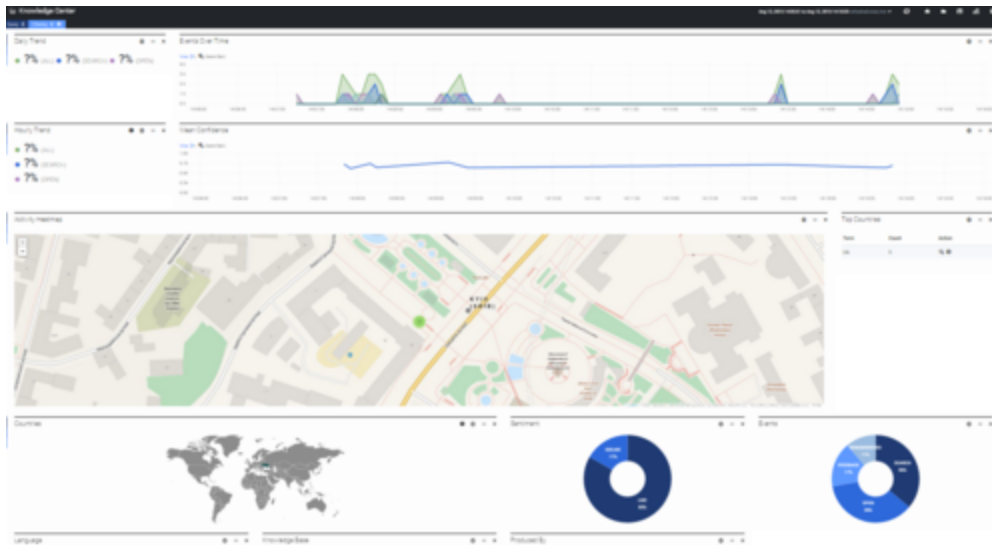
Pulse plugin is an integral part of Genesys Knowledge Center Server and does not require any additional installation steps.

Here is a sample display of key performance indicators:



Key Performance Indicators

This image shows a sample dashboard containing analytic reports:



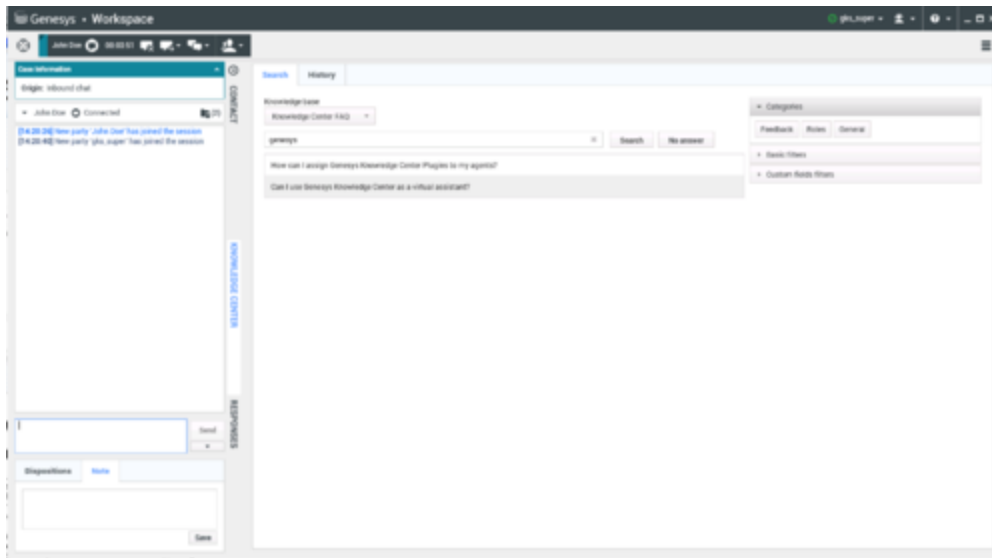
Analytics Report

Plugin for WDE

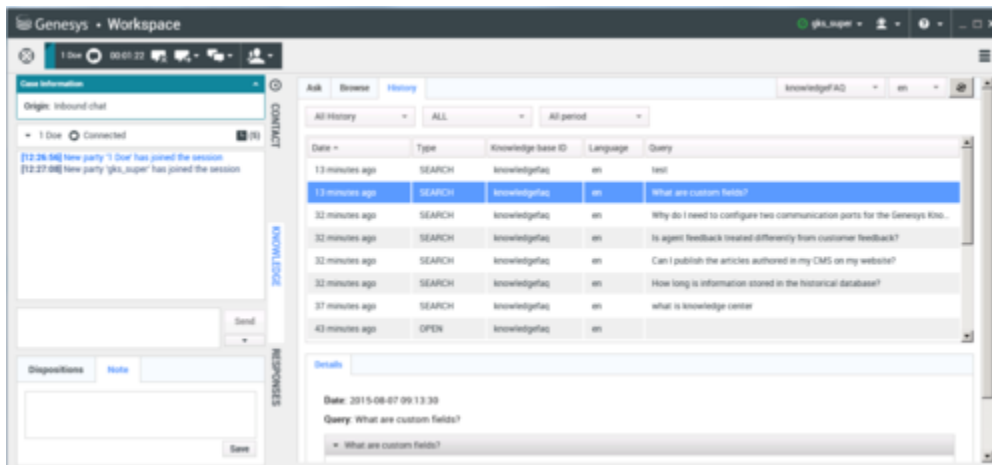
Your agents can use the Knowledge Center Plugin for Workspace Desktop Edition (WDE) to access Knowledge Center data from their WDE worksession.

For example, if a customer escalates a question using a chat widget and the resulting interaction is routed to an agent, Knowledge Center can pre-populate a search based on the data that is attached to the chat interaction. When the interaction reaches the agent, he or she will see the customer's search history, so the customer's needs can be met more quickly. In cases where the customer doesn't authorize automatic search-based access, the agent will also be able to search the customer's session history if the customer allows this during their chat.

The following images show a QNA search and customer history, respectively.



QNA Search



Customer History

Data Import Tool

You can use the data import tool to import QNA data from an XML file into a Knowledge Center index . The data in your XML file must be stored in a specific format, as shown in the following simple example:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<documents kbsId="gkc" lang="en">
  <document>
    <id>gkc_1</id>
    <question>What is Genesys Knowledge Center Server?</question>
    <answer>Genesys Knowledge Center Server combines indexing and search capabilities that
```

```
        allow for effective FAQ retrieval over one or more knowledge bases.</answer>
    <categories>
        <category>
            <id>1</id>
            <name>Common article</name>
        </category>
    </categories>
</document>
</documents>
```

Knowledge Center REST API

Knowledge Center **REST API** exposes three sets of functionality:

- The **Knowledge API** can be used by Knowledge Center Server clients who are interested in retrieving FAQ-related information from a knowledge base, including things like the structure of the knowledge base and its feedback data
- The **Management API** allows service components—such as content management systems, the Knowledge Center Administrator plugin, and data importers—to create, populate, and manage knowledge basess
- The **Reporting API** provides reporting engines—such as Easy Pulse or third-party products—with data on the various knowledge-related activities carried out by agents and customers

Genesys Web Engagement Integration

While it isn't exactly a component, we thought this would be a good place to mention that you can integrate Knowledge Center with **Genesys Web Engagement**. GWE helps you monitor, identify, and proactively engage web visitors in conversations that match your business objectives. And Knowledge Center can be used with GWE to provide proactive engagement capabilities.

For more information, see how to **integrate Knowledge Center with Genesys Web Engagement**.

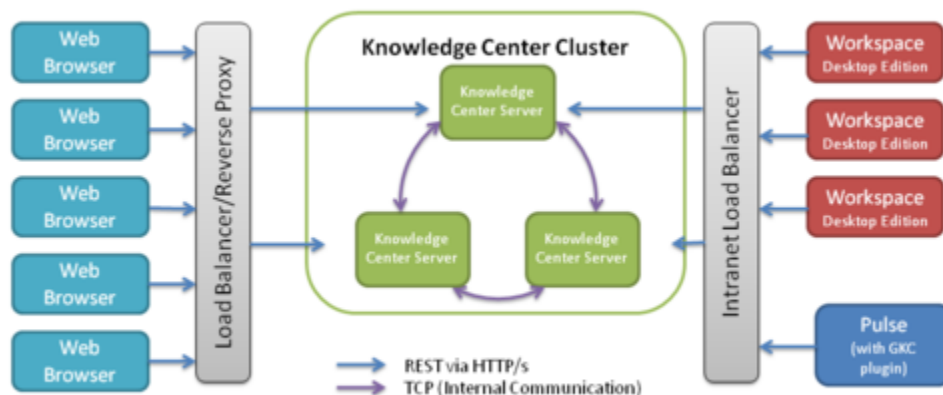
High-Level Architecture

Genesys Knowledge Center consist of several components:

- Genesys Knowledge Center Server
- Genesys Knowledge Center CMS
- Genesys Knowledge Center plugins for the following Genesys products:
 - Workspace Desktop Edition
 - Administrator
 - Pulse

Genesys Knowledge Center Server

Knowledge Center Server is the heart of the Genesys Knowledge Center solution. For purposes of load balancing and reliability, you can logically group your Knowledge Center Servers within a Knowledge Center Cluster. Each server in the cluster owns the same data and can be used to execute any desired queries against this data. These servers must be accessed by means of a properly configured load balancer that distributes the load among the server instances.



Knowledge Center High Level Architecture

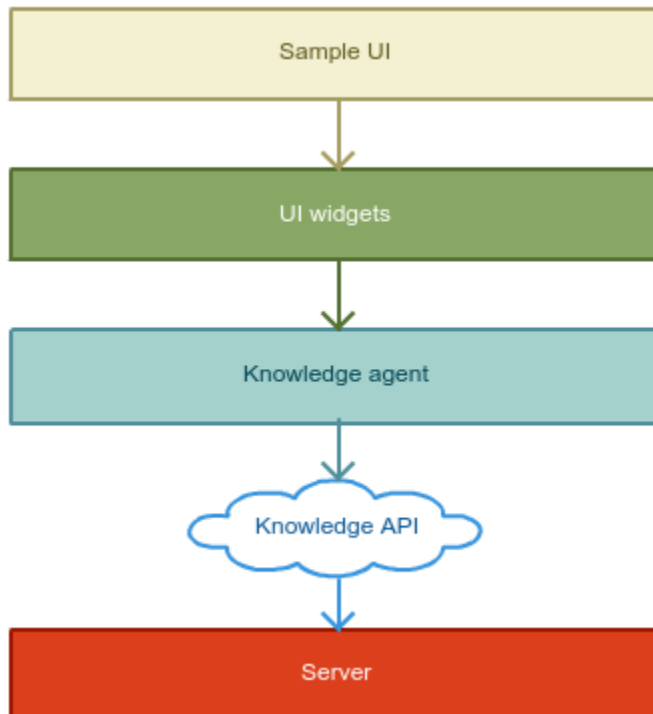
Genesys Knowledge Center Server provides several levels of integration, allowing you to access your knowledge wherever you need it—and in the way that best suits your needs. This includes a set of **RESTful APIs** that enables you to index data, query the server to find answers, and check usage information.

The Sample UI

The **Sample UI** is a JavaScript/HTML sample application that you can use as an example of how to integrate Knowledge Center into your corporate site. It runs in the visitors' browser and allows them to find answers to their questions in your corporate knowledge base.

The Sample UI offers all of the available levels of integration, allowing you to choose the one that best suits a particular need, whether it is:

- **The Knowledge API**—The RESTful web service that provides access to the Knowledge Center Server functionality
- **The Knowledge Agent**—A low-level JavaScript mapper that covers the Knowledge API and encapsulates Knowledge session management
- **The UI Widgets**—Basic and atomic UI elements covering different aspects of working with knowledge
- **The Sample UI**—An integrated sample application that implements fully functional access to the knowledge stored in Knowledge Center Server



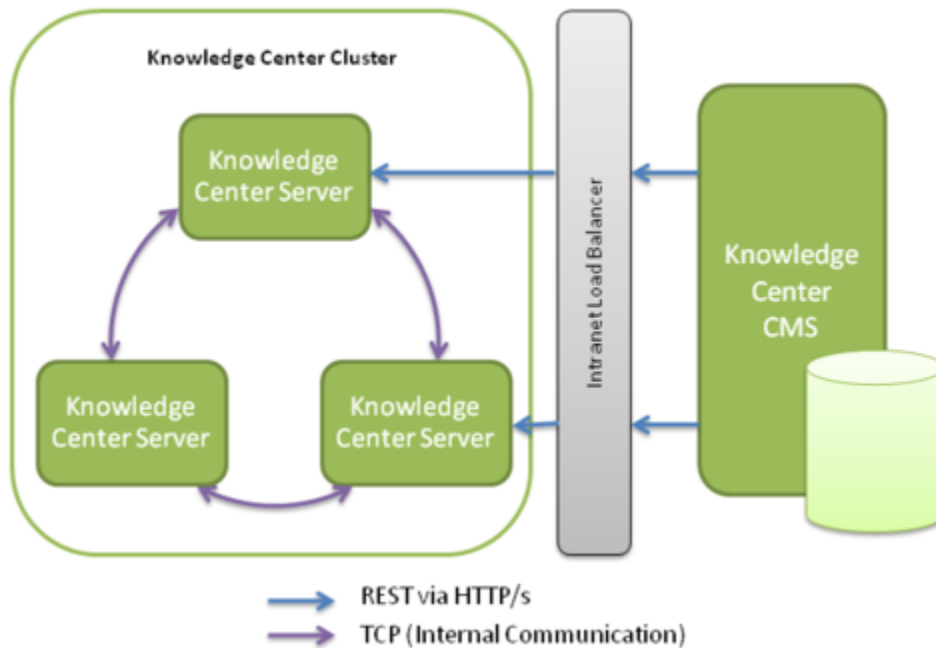
Knowledge Center Sample UI

Genesys Knowledge Center CMS

The Genesys Knowledge Center CMS is an optional component that can be used to store company knowledge and allow role-based access for authoring and improvement. The CMS is seamlessly integrated with Knowledge Center Server using its public **REST APIs** and allows you to:

- Index authored information into the Knowledge Center Server to expose it for use by agent and customers
- Retrieve usage information from the Knowledge Center Server and use it to better understand customer needs and to improve your knowledge base

For more information, consult the [Knowledge Center User's Guide](#).



Knowledge Center CMS Architecture

Genesys Knowledge Center Plugins

Genesys Knowledge Center comes with three plugins that allow it to be easily integrated into the Genesys environment:

- **The Plugin for Workspace Desktop Edition**—enriching standard agent workplace with the knowledge functionality and history of customer interaction with the knowledge
- **The Plugin for Pulse**—exposing into the Pulse capability to analyze the way how customer and agents interacts with the knowledge
- **The Plugin for Administrator**—simplifies the way you create new knowledge bases via simple step-by-step graphical interface

You can also [integrate Knowledge Center with Genesys Web Engagement](#). This allows you to take actions based on the way your knowledge is used by the customer and agents.

Prerequisites

OS Requirements

Knowledge Center Server

- OS Red Hat Enterprise Linux AS 7 (Intel EM64T)
- OS Windows Server 2008 (Intel EM64T)
- OS Windows Server 2012

Knowledge CMS

- OS Red Hat Enterprise Linux AS 7 (Intel EM64T)
- OS Windows Server 2008 (Intel EM64T)
- OS Windows Server 2012 (Intel EM64T)

Genesys Knowledge Center Plugin for Workspace Desktop Edition

- OS Windows Vista (Intel 32-bit)
- OS Windows Server 2008 (Intel 32-bit, Intel EM64T)
- OS Windows 7 (Intel 32-bit, Intel EM64T)
- OS Windows Server 2012 (Intel EM64T)
- OS Windows 8 (Intel EM64T)

Genesys Knowledge Center Plugin for Administrator

- OS Red Hat Enterprise Linux AS 5 (Intel EM64T)
- OS Windows Server 2008 (Intel EM64T)
- OS Windows Server 2012 (Intel EM64T)

Genesys Knowledge Center Plugin for Pulse

- OS Red Hat Enterprise Linux AS 5 (Intel EM64T)
- OS Windows Server 2008 (Intel 32-bit, Intel EM64T)
- OS Windows Server 2012 (Intel EM64T)

Web Browsers

- Google Chrome 34+
- Mozilla Firefox 54+
- Microsoft Internet Explorer 9, 10, 11
- Microsoft Edge
- Apple Safari 10+

Java Requirements

- Java 8 x64 SE Bundle

Important

Knowledge Center requires Oracle JDK to be used.

Genesys Environment

- Workspace Desktop Edition 8.5
- Pulse 8.5.1 or higher
- Genesys Framework 8.1–8.5
- Configuration Server (8.1.300.21 / 8.5.100.02)
- Genesys Administrator Extension (GAX) 8.5.210.10 or higher

Important

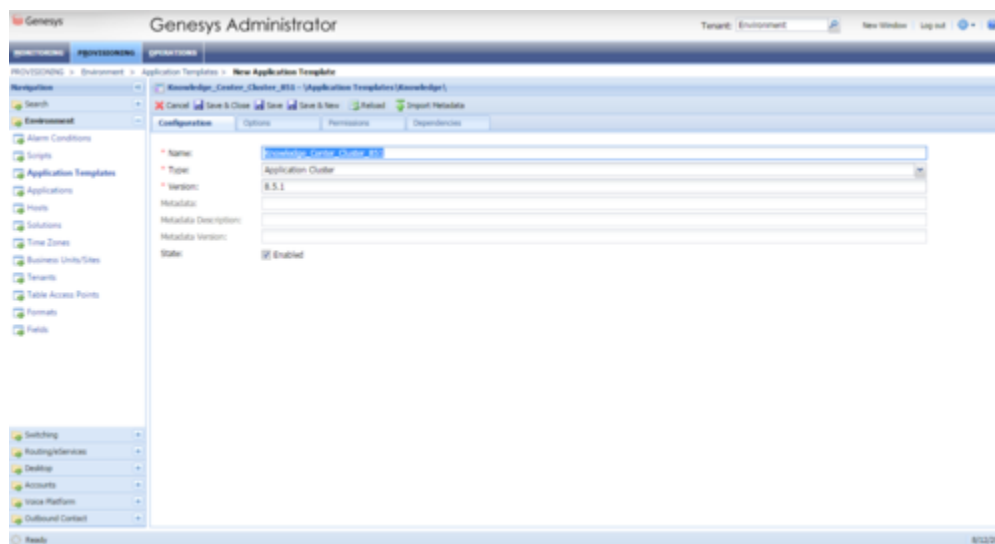
GAX is needed for Knowledge Center 3.5.302 and earlier only.

Installing the Knowledge Center Cluster Application

Carry out the procedures below, in order, to install and configure the Knowledge Center Cluster Application.

Import the Knowledge Center Cluster Application Template

1. Open Genesys Administrator and navigate to **Provisioning > Environment > Application Templates**.
2. In the **Tasks** panel, click **Upload Template**.
3. In the **Click 'Add' and choose application template (APD) file to import** window, click **Add**.
4. Browse to the *Knowledge_Center_Cluster_851.apd* file available in the templates directory of your installation CD. The **New Application Template** panel opens.



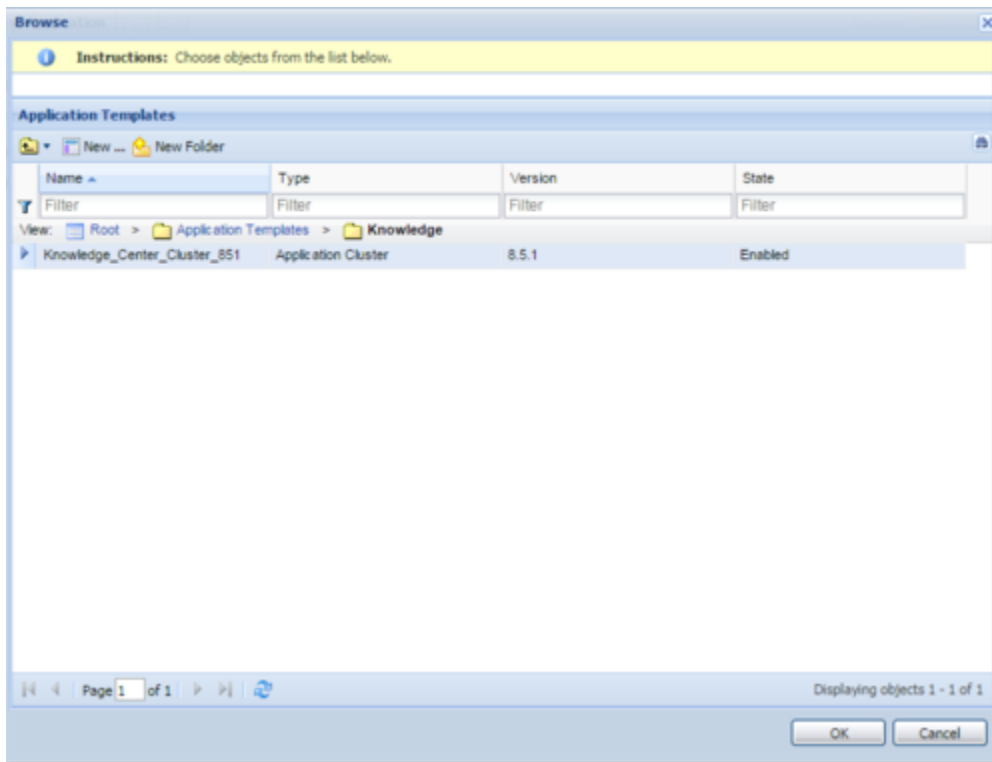
New Application Template Panel

5. Click **Save and Close**.

Create Cluster Applications

1. Open Genesys Administrator and navigate to **Provisioning > Environment > Applications**.
2. In the **Tasks** panel, click **Create New Application**.

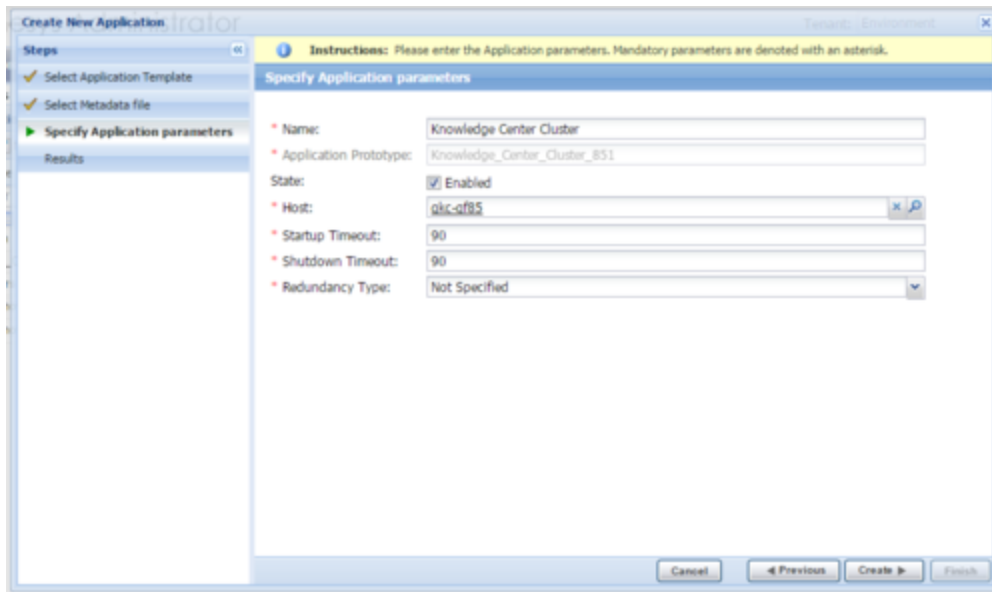
3. In the **Select Application Template** panel, click **Browse for Template** and select the Genesys Knowledge Center Cluster application template that you imported earlier. Click **OK**.



Selecting Knowledge Center Cluster Application Template

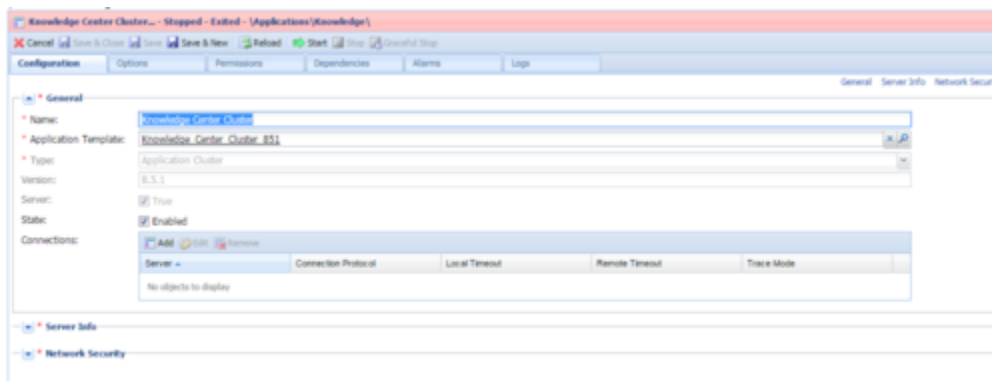
4. The template is added to the **Select Application Template** panel. Click **Next**.
5. In the **Select Metadata file** panel, click **Browse** and select the *Knowledge_Center_Cluster_851.xml* file. Click **Open**.
6. The metadata file is added to the **Select Metadata** panel. Click **Next**.
7. In **Specify Application parameters**:
 1. Enter a name for your application. For instance, *Knowledge Center Cluster*.
 2. Enable the **State**.
 3. Select the **Host** on which the Knowledge Center Cluster load-balancer will reside.
 4. Click **Create**.

Installing the Knowledge Center Cluster Application



Specifying Knowledge Center Cluster Application Parameters

8. The **Results** panel opens.
9. Enable **Opens the Application details form after clicking 'Finish'** and click **Finish**. The Knowledge Center Cluster application form opens and you can start configuring the Cluster application.

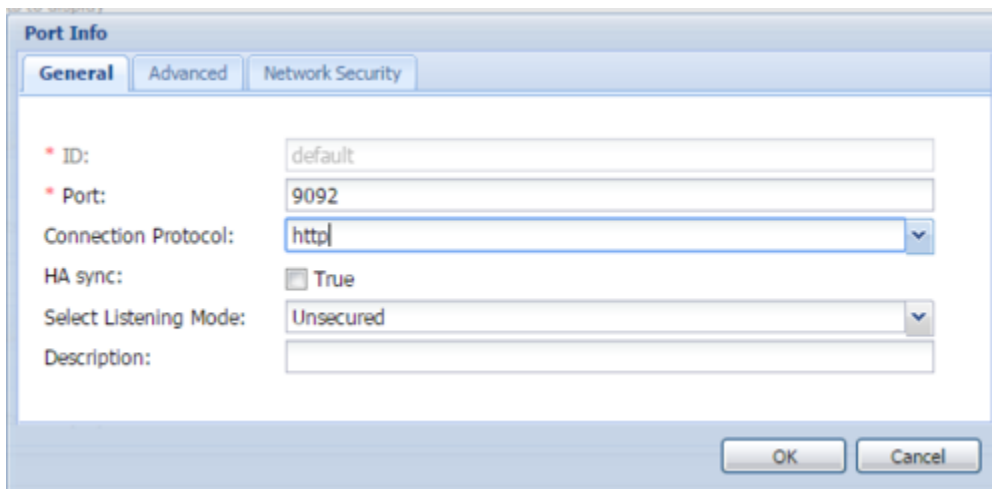


Configuring the Knowledge Center Cluster Application

Configure the Cluster Application

1. If your Knowledge Center Cluster application form is not open in Genesys Administrator, navigate to **Provisioning > Environment > Applications**. Select the application defined for the Knowledge Center Cluster and click **Edit...**
2. Expand the **Server Info** pane.
3. If your **Host** is not defined, click the lookup icon to browse to the host on which the Knowledge Center Cluster load-balancer will reside.

4. In the **Listening Ports** section, create the default port by clicking **Add**. The **Port Info** dialog opens.
 1. Enter the port number for the Knowledge Center Cluster load-balancer, for instance, *9092*.
 2. Choose *http* for the **Connection Protocol**.
 3. If you will be using a secure connection to the cluster, choose *Secured* for the **Listening Mode**.
 4. Click **OK**. The HTTP port with the default identifier appears in the list of **Listening ports**.



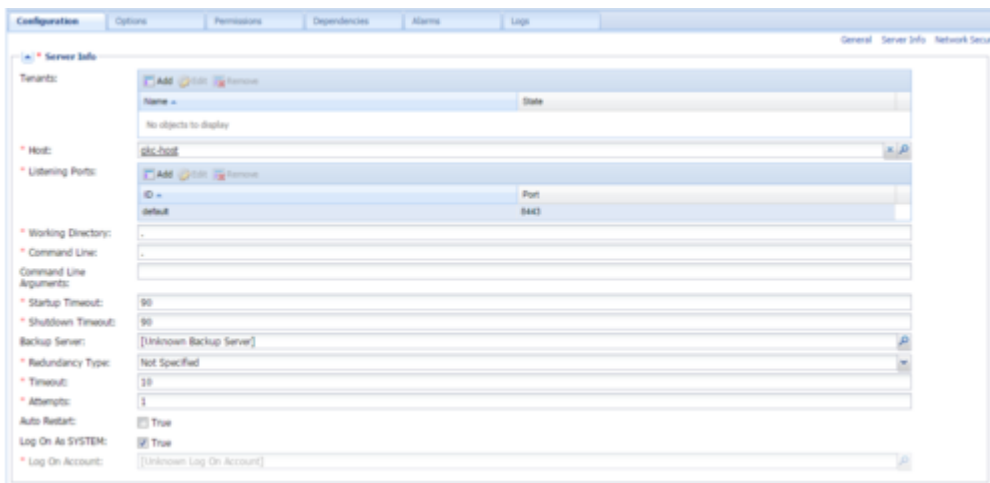
The **Port Info** dialog box is shown with the **General** tab selected. It contains the following fields and options:

- ID:** default
- Port:** 9092
- Connection Protocol:** http (selected from a dropdown menu)
- HA sync:** ☐ True
- Select Listening Mode:** Unsecured (selected from a dropdown menu)
- Description:** (empty text field)

At the bottom right are **OK** and **Cancel** buttons.

Knowledge Center Cluster Port Information

5. In the **Tenants** section, add a working tenant by clicking **Add**. Browse and choose the appropriate tenant in the pop-up dialog. Click **Ok**.
6. Ensure the **Working Directory** and **Command Line** fields contain "." (period).



The **Configuration** dialog box is shown with the **Options** tab selected. It contains the following fields and options:

- Tenants:** A table with columns **Name** and **State**. It shows "No objects to display".
- Host:** gkc-host
- Listening Ports:** A table with columns **ID** and **Port**. It shows a row with ID "default" and Port "9092".
- Working Directory:** .
- Command Line:** .
- Command Line Arguments:** (empty text field)
- Startup Timeout:** 90
- Shutdown Timeout:** 90
- Backup Server:** [Unknown Backup Server]
- Redundancy Type:** Not Specified
- Timeout:** 10
- Attempts:** 1
- Auto Restart:** ☐ True
- Log On As SYSTEM:** ☒ True
- Log On Account:** [Unknown Log On Account]

At the bottom right are **OK** and **Cancel** buttons.

Knowledge Center Cluster Server Information

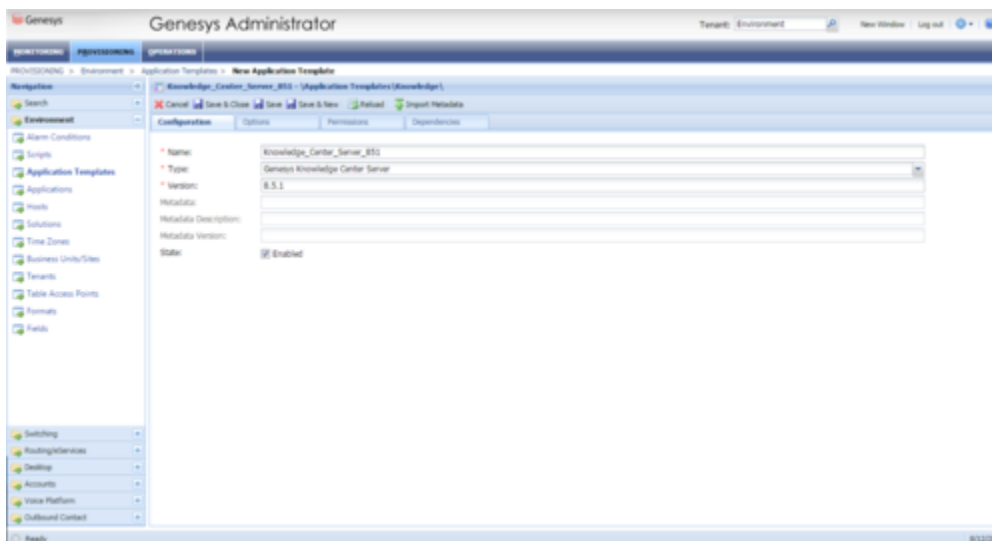
7. Click **Save**.
8. The **Confirmation** dialog for changing the application's port opens. Click **Yes**.

Installing Knowledge Center Server

Import the Knowledge Center Server Application Template

Start

1. Open Genesys Administrator and navigate to **Provisioning > Environment > Application Templates**.
2. In the **Tasks** panel, click **Upload Template**.
3. In the **Click 'Add' and choose application template (APD) file to import** window, click **Add**.
4. Browse to the *Knowledge_Center_Server_851.apd* file available in the *templates* directory of your installation CD. The **New Application Template** panel opens.



The Knowledge Center Server Application Template

5. Click **Save and Close**.

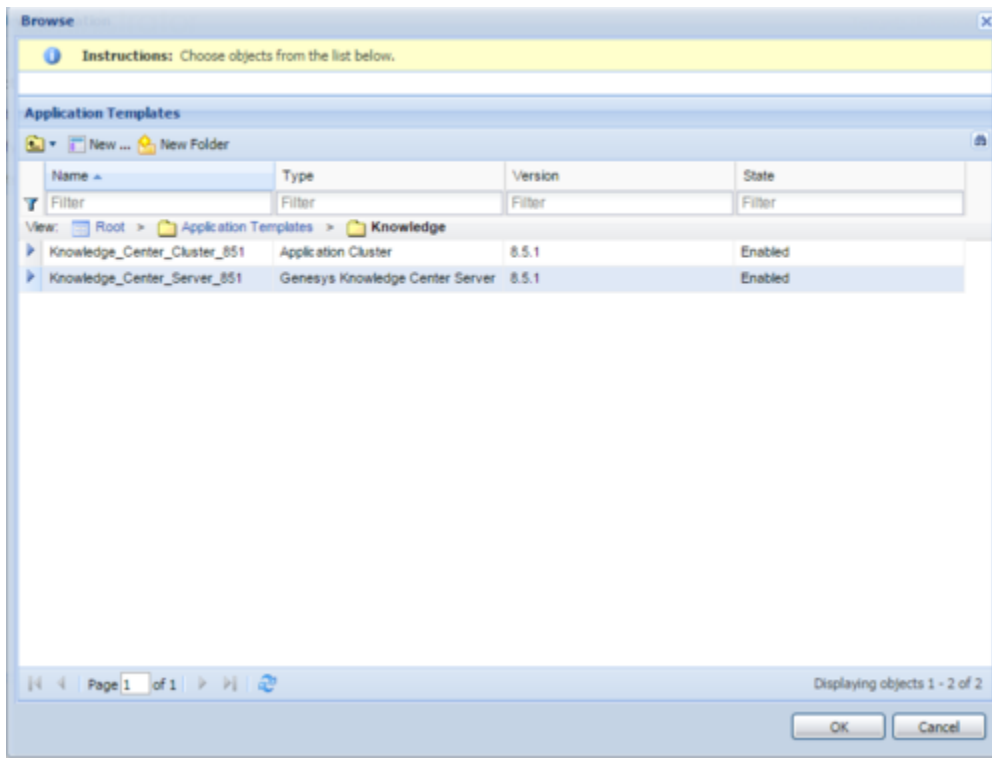
End

Create Server applications

Start

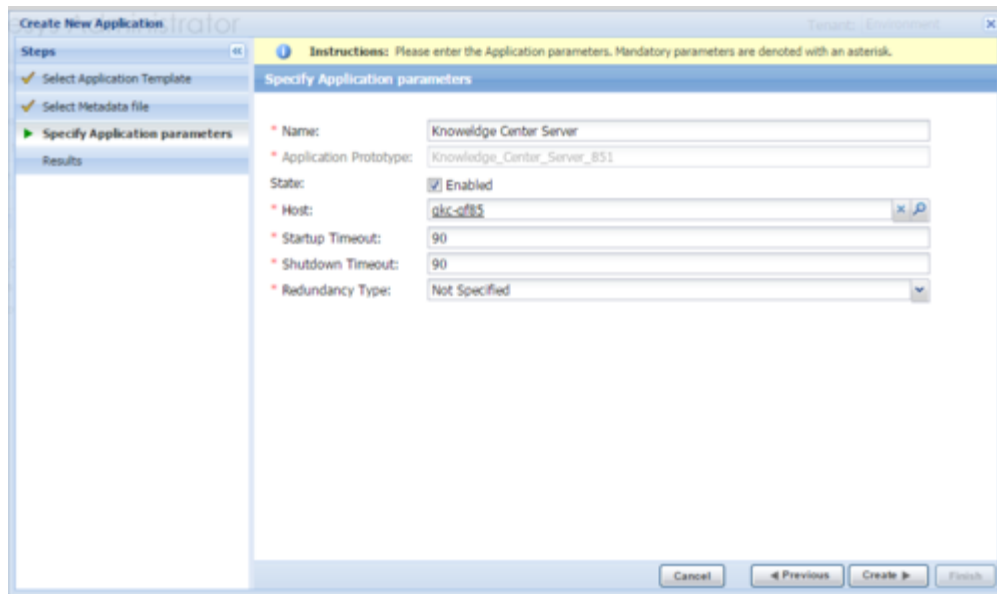
1. Open Genesys Administrator and navigate to **Provisioning > Environment > Applications**.
2. In the **Tasks** panel, click **Create New Application**.

3. In the **Select Application Template** panel, click **Browse for Template** and select the Genesys Knowledge Center Server application template that you imported earlier. Click **OK**.



Selecting the Knowledge Center Server Template

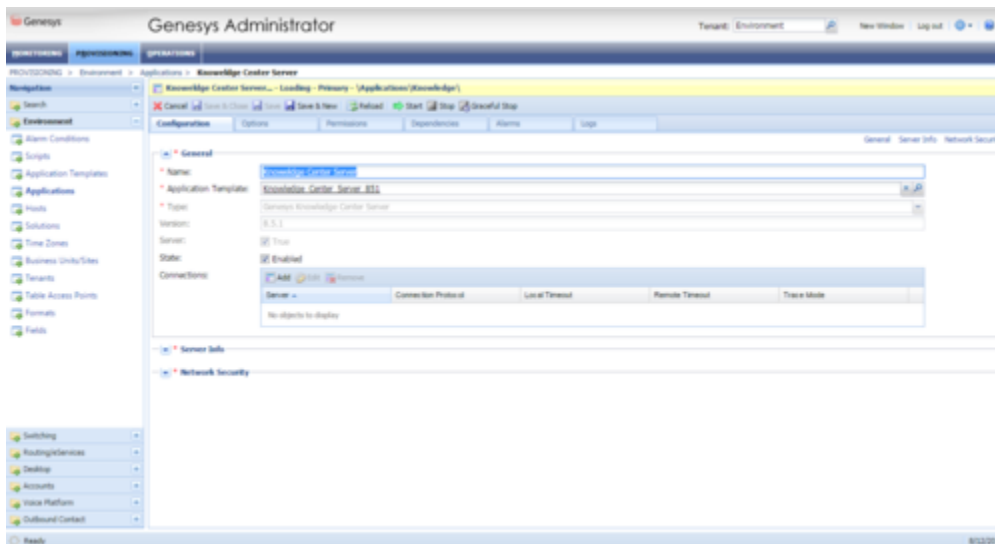
4. The template is added to the **Select Application Template** panel. Click **Next**.
5. In the **Select Metadata** file panel, click **Browse** and select the *Knowledge_Center_Server_851.xml* file. Click **Open**.
6. The metadata file is added to the **Select Metadata** file panel. Click **Next**.
7. In **Specify Application parameters**:
 1. Enter a name for your application. For instance, *Knowledge Center Server*.
 2. Enable the **State**.
 3. Select the Host on which the Knowledge Center Server will reside.
 4. Click Create.



Creating the Knowledge Center Server Application

5. The **Results** panel opens.
6. Enable **Opens the Application details** form after clicking **Finish** and click **Finish**.

The Knowledge Center Server application form opens and you can start configuring the Knowledge Center Server application.



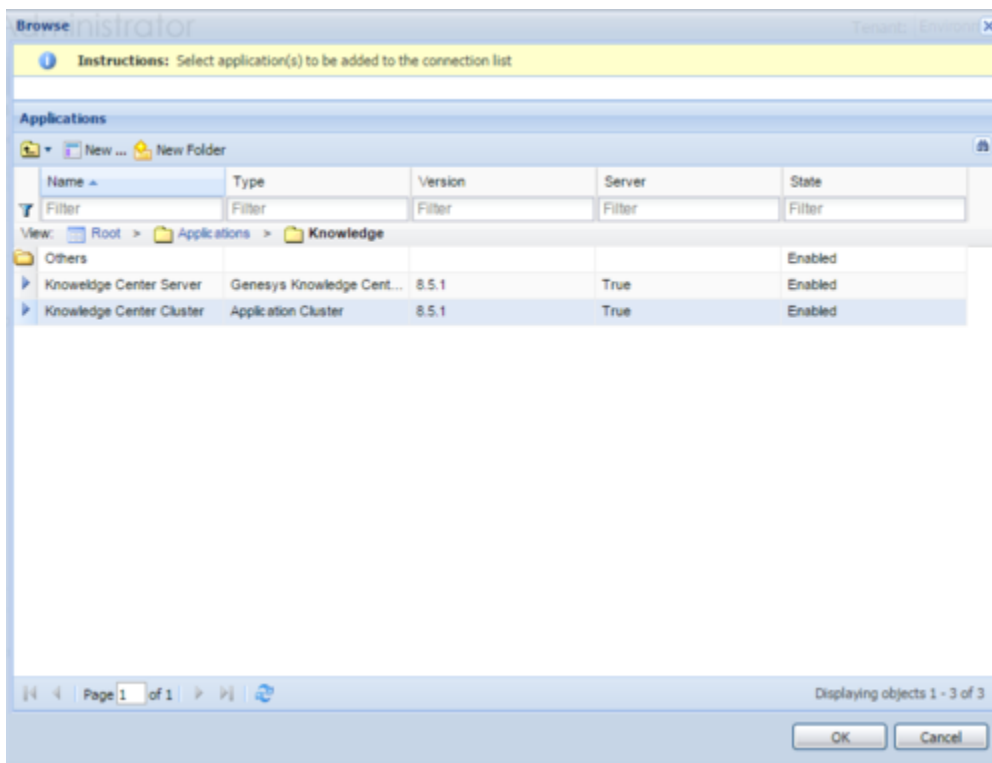
Knowledge Center Server Application Details

End

Configuring the Knowledge Center Server Application

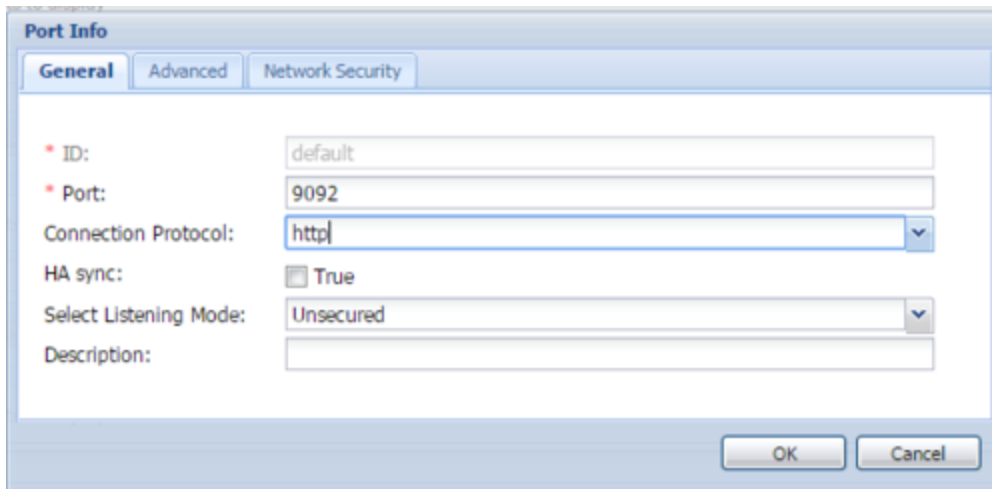
Start

1. If your Knowledge Center Server application form is not open in Genesys Administrator, navigate to **Provisioning > Environment > Applications**. Select the application defined for the Knowledge Center Server and click **Edit...**.
2. In the **Connections** section of the **Configuration** tab, click **Add**. The **Browse for applications** panel opens. Select the Knowledge Center Cluster application, then click **OK**.



Selecting the Knowledge Center Cluster Application

1. Expand the **Server Info** pane.
2. If your **Host** is not defined, click the lookup icon to browse to the hostname of your application.
3. In the **Listening Ports** section, create the default port by clicking **Add**. The **Port Info** dialog opens.
 1. Enter the **Port**. For instance, *9092*. This should be the port number for the Knowledge Center Server instance.
 2. Click **OK**. The port with the default identifier appears in the list of **Listening ports**.



The screenshot shows the 'Port Info' dialog box with the 'General' tab selected. The fields are as follows:

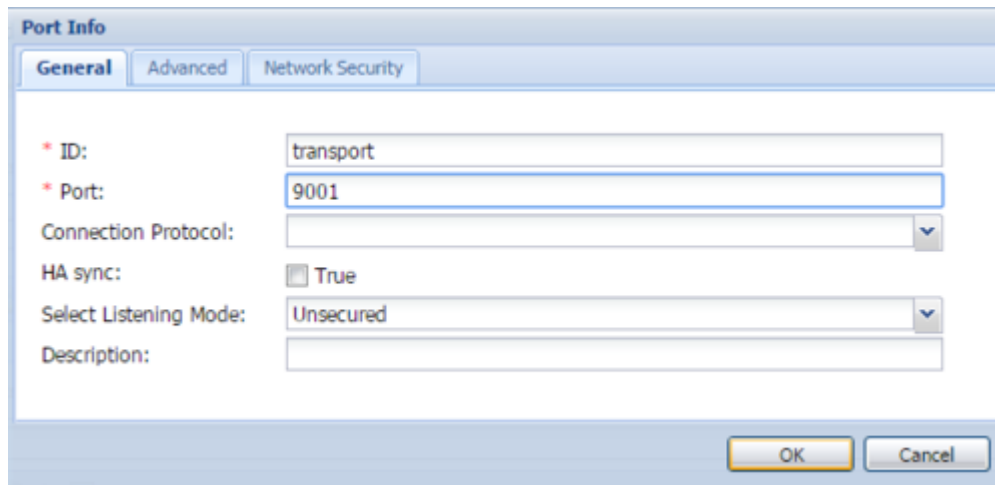
Field	Value
ID	default
Port	9092
Connection Protocol	http
HA sync	<input checked="" type="checkbox"/> True
Select Listening Mode	Unsecured
Description	

Buttons: OK, Cancel

Knowledge Center Server Port Information

6. Optionally, you can explicitly add Transport port for ElasticSearch engine. If you do not define transport port, port 9300 will be used. To specify the stop port, click the **Add** button. The Port Info dialog opens.

1. Enter *transport* for the **ID** field.
2. Enter the **Port**. For instance, 9001.
3. Click **OK**.



The screenshot shows the 'Port Info' dialog box with the 'General' tab selected. The fields are as follows:

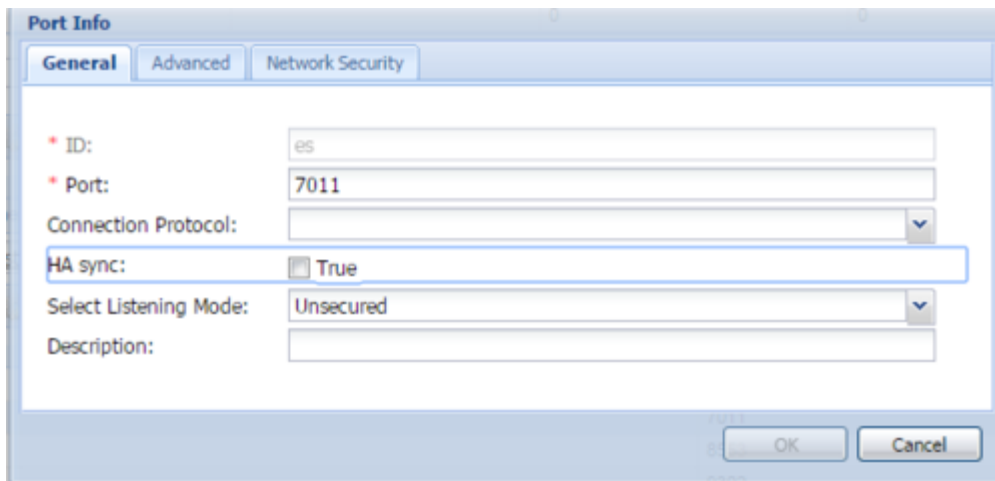
Field	Value
ID	transport
Port	9001
Connection Protocol	
HA sync	<input checked="" type="checkbox"/> True
Select Listening Mode	Unsecured
Description	

Buttons: OK, Cancel

Knowledge Center Server Transport Port Information

7. Optionally, you can explicitly add a port for access to ElasticSearch engine. If you do not define this port, port 9200 will be used. To specify the stop port, click the **Add** button. The **Port Info** dialog opens.

1. Enter *es* for the ID field.
2. Enter the **Port**. For instance, 7011
3. Click **OK**.



Port Info

General Advanced Network Security

* ID: es

* Port: 7011

Connection Protocol: [dropdown]

HA sync: ☒ True

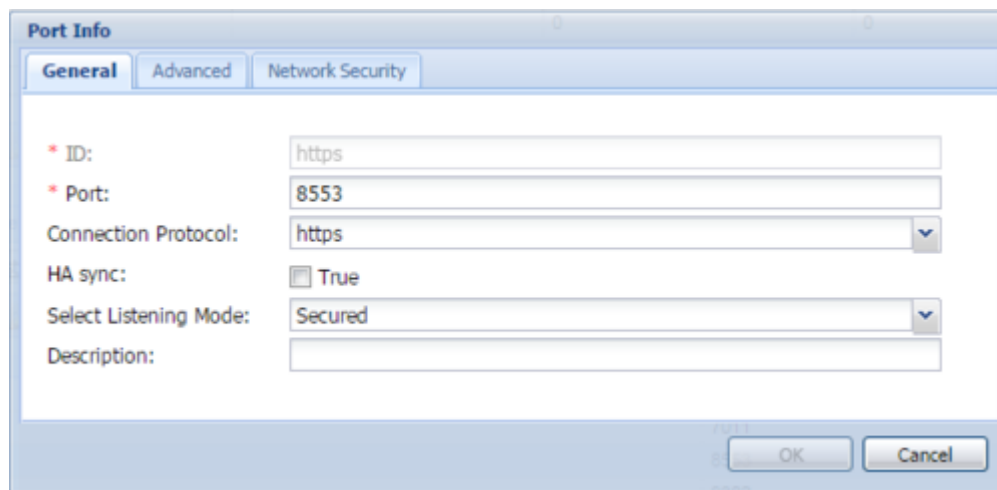
Select Listening Mode: Unsecured [dropdown]

Description: [text box]

OK Cancel

GKS es Port Information

8. Optionally, you can add a secure listening port for authenticated users, secured connections, and secure chat. Click **Add**. The **Port Info** dialog opens.
 1. Enter *https* for the ID field
 2. Enter the **port** . For instance, 8553
 3. Enter *https* for the **Connection Protocol**.
 4. Choose **Secured** for the **Listening Mode**.
 5. Click **OK**.



Port Info

General Advanced Network Security

* ID: https

* Port: 8553

Connection Protocol: https [dropdown]

HA sync: ☒ True

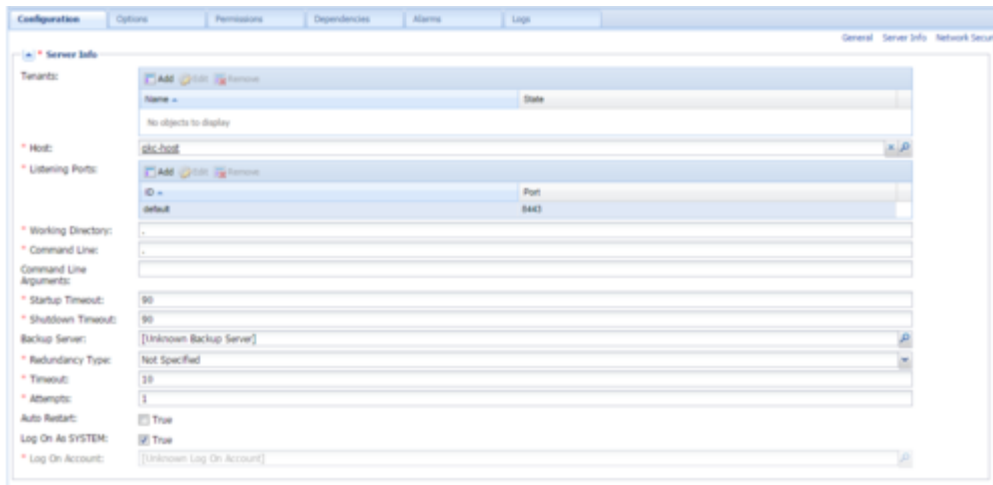
Select Listening Mode: Secured [dropdown]

Description: [text box]

OK Cancel

GKS https Port Information

6. In the **Tenants** section, add a working tenant by clicking **Add**. Browse and choose the appropriate tenant in the pop-up dialog. Click Ok.
7. Ensure the **Working Directory** and **Command Line** fields contain "." (period).



Knowledge Center Server Application Information

8. If you are using Access Groups to assign privileges to agents:
 - Uncheck **Log On As System**
 - In **Log On Account** specify the user account that has the ability to view access groups (for example, user from the Super Administrators access group).
9. Click **Save**.
10. The Confirmation dialog for changing the application's port opens. Click **Yes**.
11. (Optional) Select the **Options** tab. In the **[log]** section, the **all** option is set to *stdout* by default. Enter a filename if you wish to enable logging to a file. For example, you can enter *stdout, C:\Logs\Knowledge\Knowledge_server* to force the system to write logs both to the console and to a file.

Log (5 Rows)			
logall	log	all	stdout, C:\Logs\Knowledge\Knowledge_server
logerror	log	error	20
logsegment	log	segment	10000
logstandard	log	standard	stdout
logtrace	log	trace	stdout
logverbose	log	verbose	all

Knowledge Center Server Application Logging Options

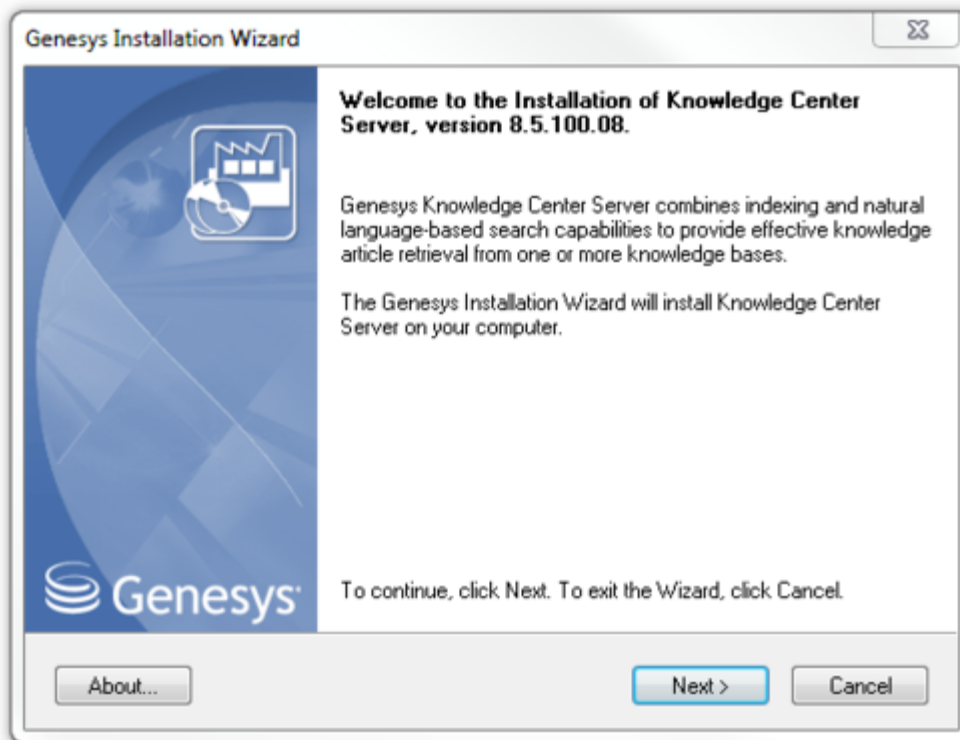
End

Installing Knowledge Center Server

Windows Installation Procedure

Start

1. In your installation package, locate and double-click the *setup.exe* file. The Install Shield opens the welcome screen.



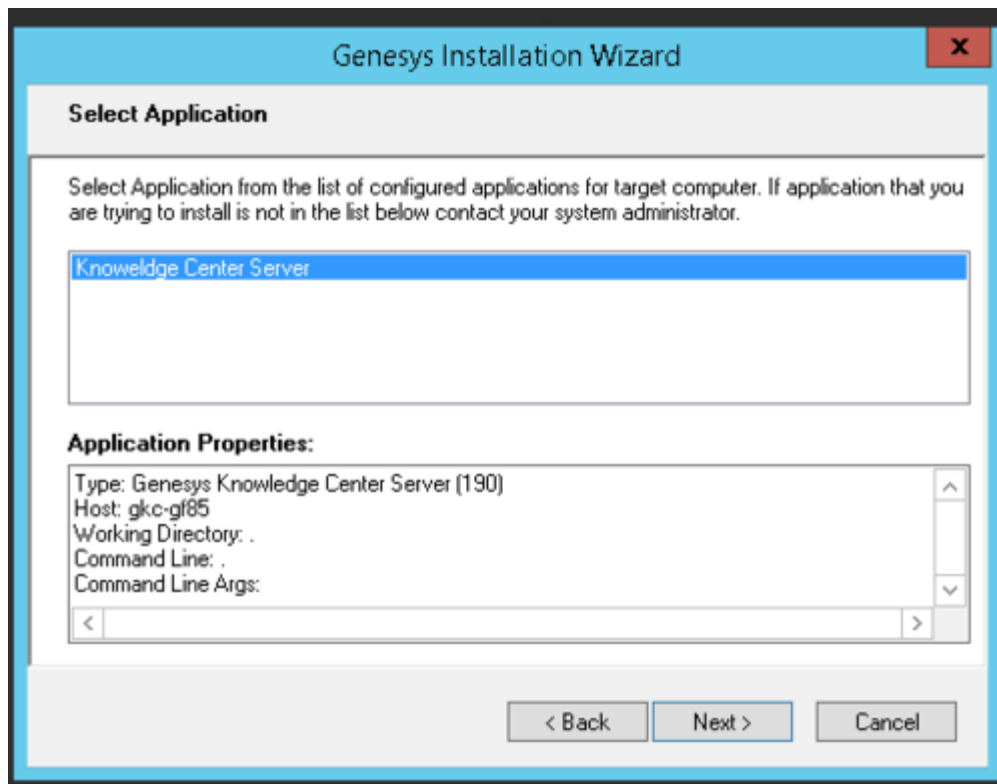
Knowledge Center Server Installation Window

2. Click **Next**. The **Connection Parameters to the Configuration Server** screen appears.

The screenshot shows a Windows-style dialog box titled "Genesys Installation Wizard" with a close button (X) in the top right corner. The main title is "Connection Parameters to the Configuration Server". Below this, a message states: "The parameters in the Host and User fields are required to establish a connection to Configuration Server." There are two main sections: "Host" and "User". The "Host" section includes the instruction "Specify the host name and port number for the machine on which Configuration Server is running." and has two input fields: "Host name:" with the value "localhost" and "Port:" with the value "2020". The "User" section includes the instruction "Specify your Configuration Server user name and password." and has two input fields: "User name:" with the value "default" and "Password:" with a masked password represented by ten black dots. At the bottom of the dialog are three buttons: "< Back", "Next >", and "Cancel".

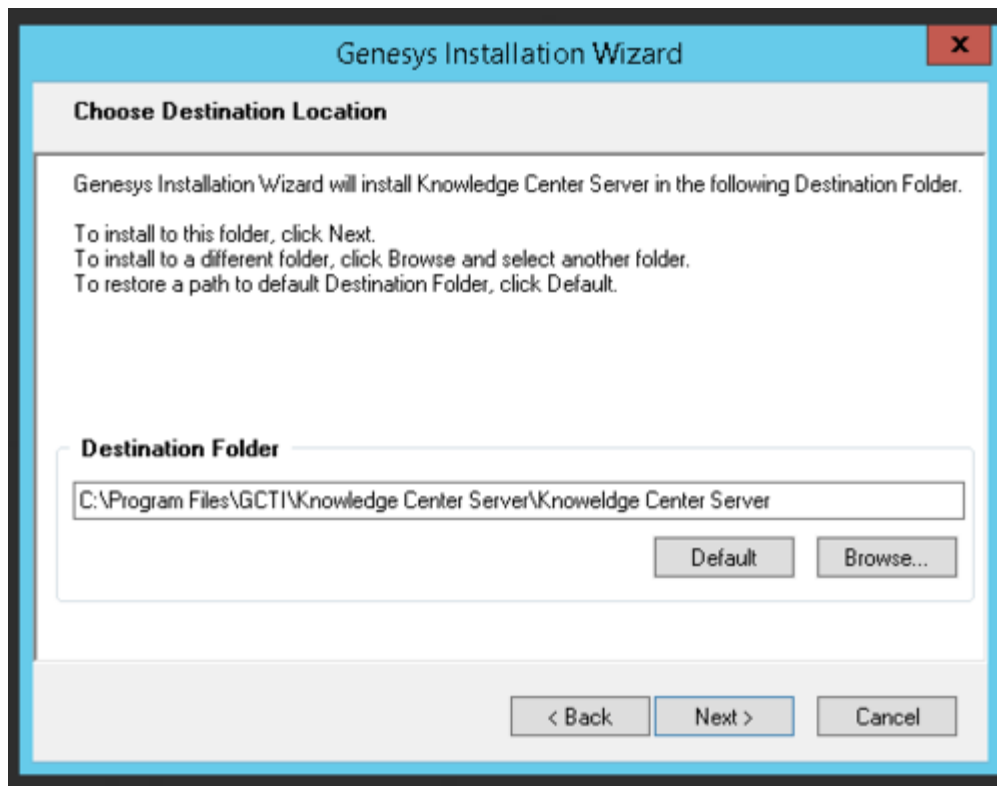
Knowledge Center Server Connection Parameters

3. Under **Host**, specify the host name and port number where Configuration Server is running. (This is the main listening port entered in the **Server Info** tab for Configuration Server.)
4. Under **User**, enter the user name and password for logging into Configuration Server.
5. Click **Next**. The **Select Application** screen appears.



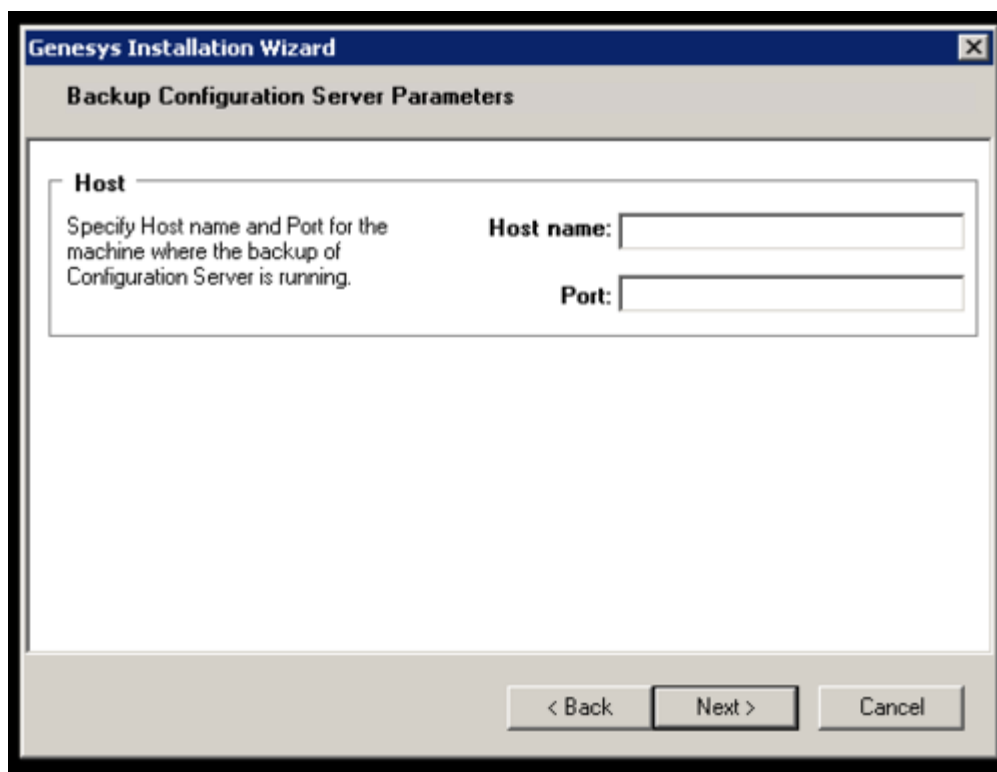
Selecting the Knowledge Center Server Application

6. Select the Knowledge Center Server application that you are installing. The **Application Properties** area shows the **Type**, **Host**, **Working Directory**, **Command Line executable**, and **Command Line Arguments** information previously entered in the **Server Info** and **Start Info** tabs of the selected Application object.
7. Click **Next**. The **Choose Destination Location** screen appears.



Choosing the Knowledge Center Server Installation Destination

8. Under **Destination Folder**, keep the default value or browse to the desired installation location.
9. Click **Next**. The **Backup Configuration Server Parameters** screen appears.



Genesys Installation Wizard

Backup Configuration Server Parameters

Host

Specify Host name and Port for the machine where the backup of Configuration Server is running.

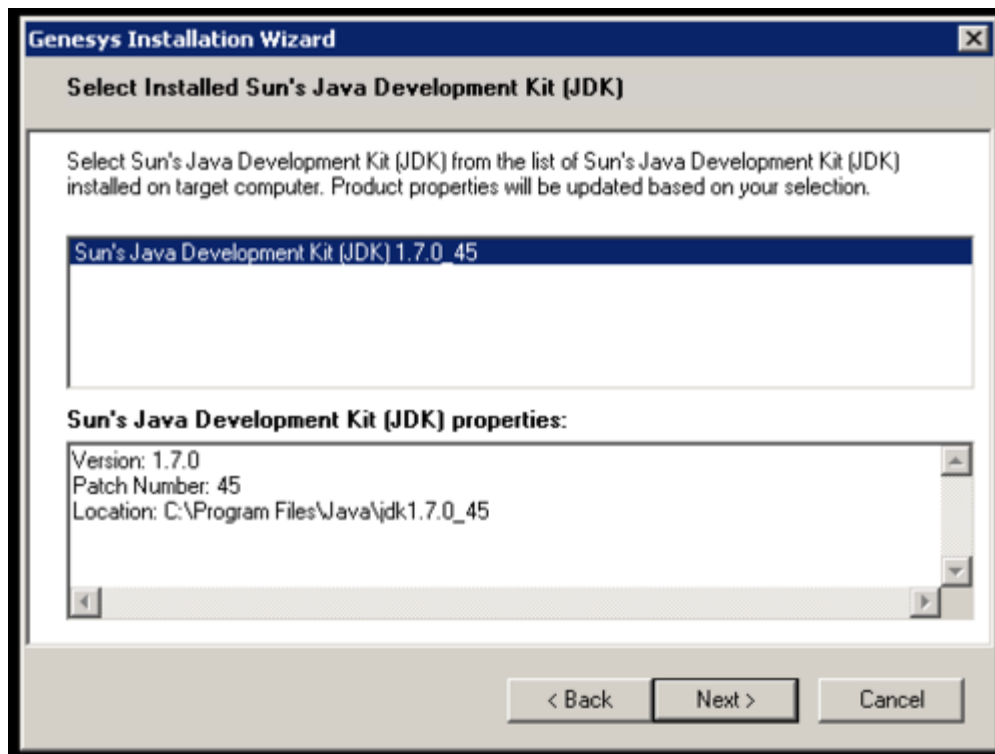
Host name:

Port:

< Back Next > Cancel

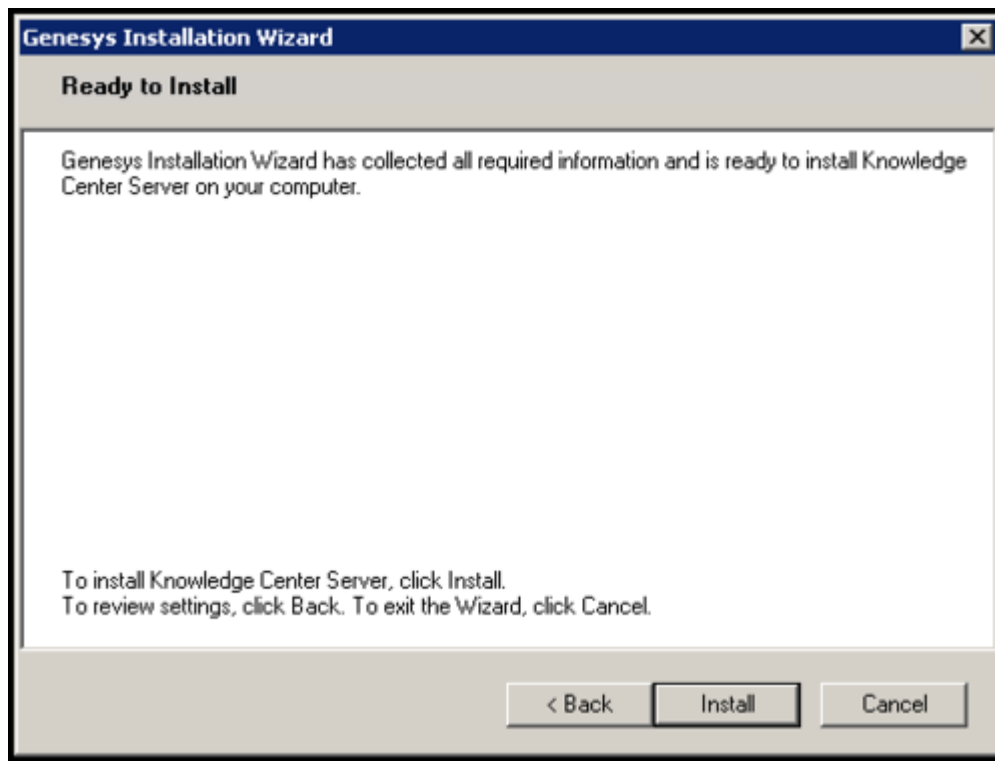
Knowledge Center Backup Config Server Parameters

10. If you have a backup Configuration Server, enter the **Host name** and **Port**.
11. Click **Next**. Choose the appropriate version of the Java JDK.



Selecting the Knowledge Center Server Java Version

12. Click **Next**. The **Ready to Install** screen appears.



Knowledge Center Server is Ready to Install

13. Click **Install**. The Genesys Installation Wizard indicates it is performing the requested operation for Backend Server. When through, the **Installation Complete** screen appears.
14. Click **Finish** to complete your installation.
15. Inspect the directory tree of your system to make sure that the files have been installed in the location that you intended.

Important

The Windows service will not be automatically configured during installation. To configure the Windows service, start *server.bat* with the following parameters: **server.bat install**. To run the server as service, comment out the (REM) APP_TYPE property in *senenv.bat* before installing the service.

End

Linux Installation Procedure

Important

Knowledge Center Server must have boost libraries installed on the Linux host for a successful installation. You must install boost.x86_64 library with dependencies (boost-filesystem.x86_64, boost-graph.x86_64, boost-program-options.x86_64, boost-regex.x86_64, boost-system.x86_64, boost-thread.x86_64, boost-wave.x86_64). For example using yum packet manager: `yum install boost`

Start

1. Open a terminal in the Genesys Knowledge Center Server CD/DVD or the Genesys Knowledge Center Server installation package and run the *install.sh* file. The Genesys installation starts.
2. Enter the hostname of the host on which you are going to install.
3. Enter the connection information required to log in to the Configuration Server:
 1. **Hostname**—For instance, *demosrv.genesyslab.com*
 2. **Listening port**—For instance, *2020*
 3. **User name**—For instance, *demo*
 4. **Password**
4. If you have a backup Configuration Server, enter the **Host name** and **Port**.
5. If the connection settings are successful, a list of keys and Genesys Knowledge Center Server applications is displayed.
6. Enter the key for the Genesys Knowledge Center Server application that you created previously on

Configuration Server.

7. Enter the full path to your installation directory and confirm that it is correct.

If the installation is successful, the console displays the following message:

Installation of Genesys Knowledge Center Server, version 8.5.x has completed successfully.

End

Understanding the Knowledge Center Server Configuration Files

Knowledge Center Server includes an embedded Jetty server and Lingua Tools in its installation folder. Product installation pre-configures all of the links between these resources, but there are cases in which they need to be changed. This section describes how to work with the configuration files stored in the Knowledge Center Server.

Jetty Configuration

After you complete these steps, Knowledge Center Server will be available as a web service on the following URLs:

- `http://host:jetty.port/gks-server`—GKC Server
- `http://host:jetty.port/gks-sample-ui`—Sample UI sandbox

ElasticSearch Engine Configuration

1. Go to the `./server` folder and open the `gks.yml` configuration file.
2. Configure the following settings:
 1. `index.number_of_shards`: #—Number of ElasticSearch shards
 2. `path.data`: [PATH]—Path to the folder that contains index data for this node (default: `/gks/data`)
 3. `path.freeling`: [PATH]—Path to Freeling data folder
 4. `path.plugins`: [PATH] – path to Pulse Plugins
 5. `path.geoip`: [PATH]\GeoLite2-City.mmdb—Path to geo database used for calculations of IP location

Language Resources Configuration

- GEOIP Database
 - Database for Geo-IP Location
 - The path to `/linguatools/geoip/GeoLite2-City.mmdb` can be changed in the `gks.yml` file: `path.geoip`.
- Freeling tokenizer:

- In Windows
 - The path to `/linguatools/freeling/data/` can be changed in the `gks.yml` file: `path.freeling`.
 - The following path will be added to the Windows PATH variable during installation: *Path to installation directory/linguatools/freeling/bin*.
- In Linux
 - The path to `/linguatools/freeling/data/` can be changed in the `gks.yml` file: `path.freeling`.
 - `setenv.sh` exports the following environment variables:
 - `FREELINGSHARE`—Path to *Path to installation directory/linguatools/freeling/data*
 - `LD_LIBRARY_PATH`—Path to *Path to installation directory/linguatools/freeling/bin*

Tip

Access to a knowledge base may be limited by an agent's assigned skills (see [Installing and Using the Administrator Plugin](#)). Please add the appropriate skills so your agent may see the required knowledge bases (see [Bulk Assignment of Skills to Agents](#) for more information).

Provide Knowledge Center Access to Agents

Genesys Knowledge Center supports the following privileges to restrict agent access:

- **Knowledge.ADMINISTER**—Configure knowledge bases in the Knowledge Center Cluster application
- **Knowledge.AUTHOR**—Create, populate, and manage knowledge bases
- **Knowledge.REPORTING**—Extract data on the activities carried out by agents and customers while using the knowledge service

To configure the appropriate privileges for an Agent:

Start

1. Go to **Provisioning > Accounts > Roles**.
2. In the taskbar, click **New** to create a new object.
3. Set the name of the role in the **General** section.

The screenshot shows the 'Configuration' window with the 'General' tab selected. The 'Name' field is 'knowledge_manager', 'Description' is 'Role for manage Knowledge Center', 'Tenant' is 'Environment', and 'State' is 'Enabled'. The 'Members' section is collapsed.

Knowledge Center Server Access Roles

4. Go to the **Role Privileges** tab and select the set of roles for Genesys Knowledge Center.
5. Open the list of privileges for Knowledge Center Server.
6. Set the appropriate privileges to **Allowed**.

The screenshot shows the 'Configuration' window with the 'Role Privileges' tab selected. The 'Add/Remove Products' section shows 'Genesys Knowledge Center 8.5.000.00' selected. The 'Privileges' section shows a list of privileges with the 'Value' column set to 'Allowed'.

Name	Value
Genesys Knowledge Center Server Privileges (3 items)	
Allows agent to change data in a knowledge base	Allowed
Allows agent to manage knowledge bases	Allowed
Allows agent to use reporting capabilities	Allowed

Setting Knowledge Center Server Access Privileges

7. Go back to the **Configuration** tab.
8. In the **Members** section, add the appropriate Agent by clicking the **Add** button.

The screenshot shows the 'Configuration' window with the 'Members' tab selected. The 'Users' section shows a table with columns: User Name, Agent, Last Name, First Name, Employee ID, and State. The 'Access Groups' section shows a table with columns: Name, Type, and State.

User Name	Agent	Last Name	First Name	Employee ID	State
default	False	default	default	0	Enabled

Name	Type	State
Administrators	Administrators	Enabled

Knowledge Center Server Members Section

9. Save and Close.

End

Installing the Knowledge Center CMS

Important

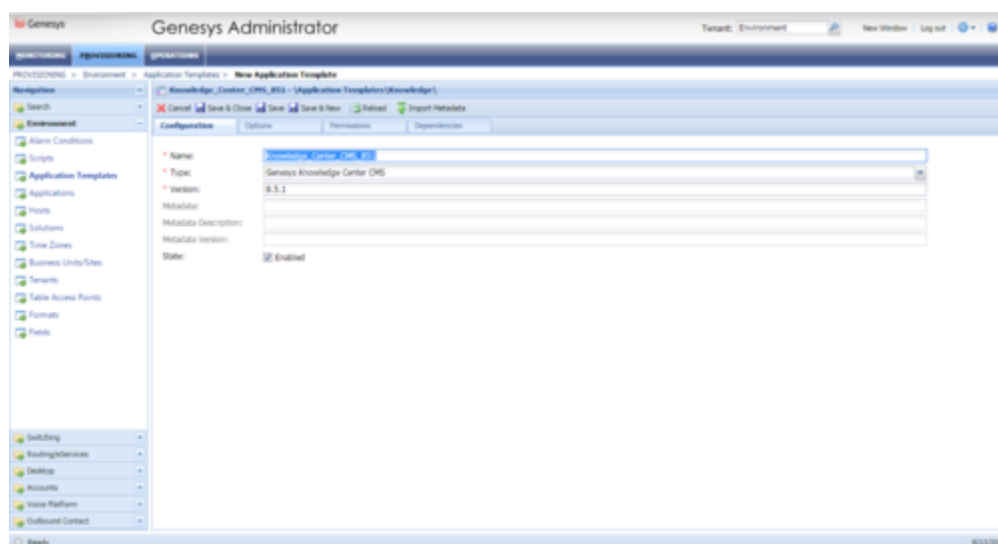
You need to configure the CMS options that are stored in the "cms:general" section of the Application Cluster object. (See [Configuration Options](#))
If you would like to use several instances of the CMS, you need to configure the CMS Cluster. (See [Configuring CMS Cluster](#))

Install the CMS

Import the CMS Application Template

Start

1. Open Genesys Administrator and navigate to **Provisioning > Environment > Application Templates**.
2. In the **Tasks** panel, click **Upload Template**.
3. In the **Click 'Add' and choose application template (APD) file to import** window, click **Add**.
4. Browse to the *Knowledge_Center_CMS_851.apd* file available in the templates directory of your installation CD. The **New Application Template** panel opens.



The Knowledge Center CMS Application Template

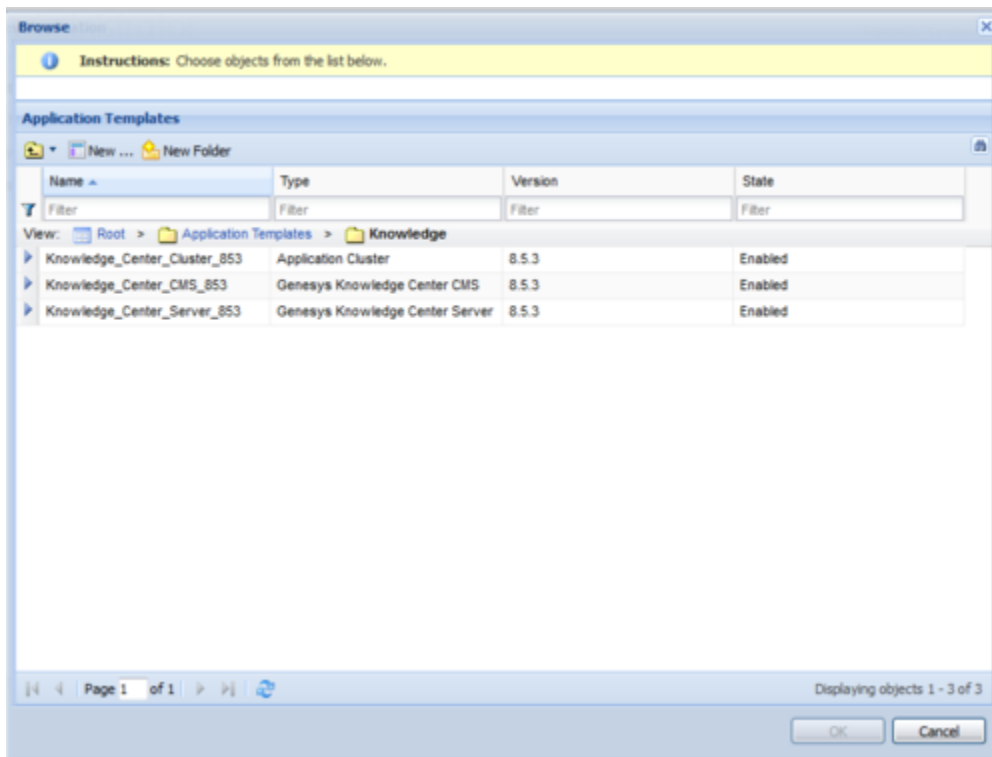
5. Click **Save and Close**.

End

Create CMS Applications

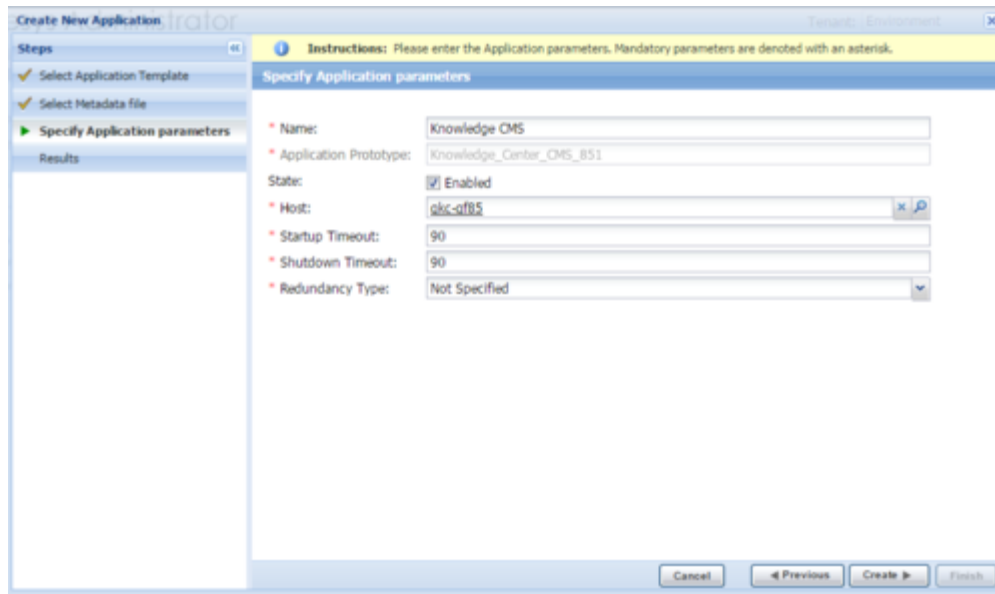
Start

1. Open Genesys Administrator and navigate to **Provisioning > Environment > Applications**.
2. In the **Tasks** panel, click **Create New Application**.
3. In the **Select Application Template** panel, click **Browse for Template** and select the Genesys Knowledge Center Server application template that you imported earlier. Click **OK**.



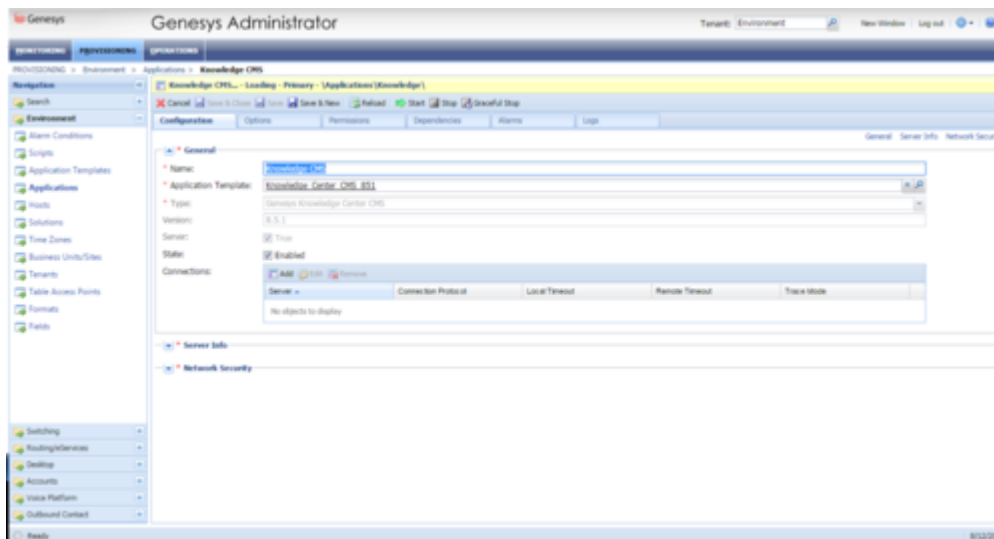
Selecting the Knowledge Center CMS Template

4. The template is added to the **Select Application Template** panel. Click **Next**.
5. In the **Select Metadata** file panel, click **Browse** and select the *Knowledge_Center_CMS_851.xml* file. Click **Open**.
6. The metadata file is added to the **Select Metadata** file panel. Click **Next**.
7. In **Specify the appropriate application parameters**:
 1. Enter a name for your application. For instance, *Knowledge Center CMS*.
 2. Enable the **State**.
 3. Select the Host on which the CMS will reside.
 4. Click **Create**.



Creating the Knowledge Center CMS Application

8. The **Results** panel opens.
9. Enable **Opens the Application details form after clicking 'Finish'** and click **Finish**. The Knowledge Center CMS application form opens and you can start configuring the CMS application.



Configuring the Knowledge Center CMS

End

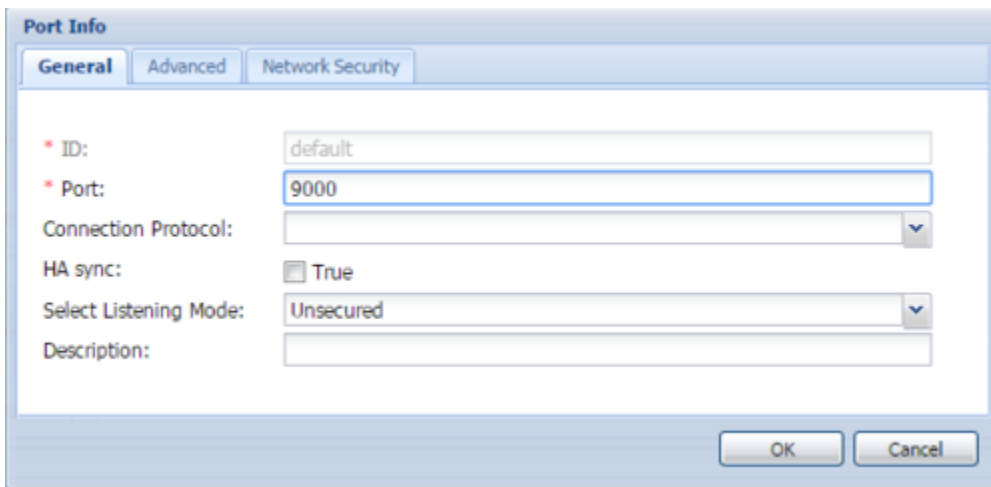
Configure the CMS Application

Start

1. If your Knowledge Center CMS application form is not open in Genesys Administrator, navigate to
-

Provisioning > Environment > Applications. Select the application defined for the Knowledge Center CMS and click **Edit...**

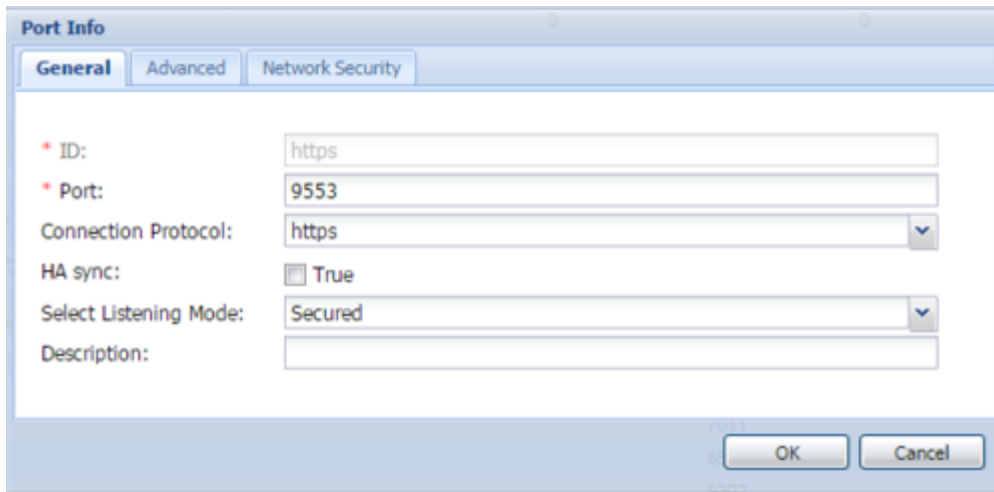
2. In the **Connections** section of the **Configuration** tab, click **Add**. The **Browse for applications** panel opens.
3. Select the Knowledge Center Cluster application, then click **OK**.
4. Expand the **Server Info** pane.
5. If your Host is not defined, click the lookup icon to browse to the hostname of your application.
6. In the **Listening Ports** section, create the default port by clicking **Add**. The **Port Info** dialog opens.
 1. Enter the **Port**. For instance, 9000.
 2. Click **OK**. The port with the default identifier appears in the list of **Listening ports**.



The screenshot shows the 'Port Info' dialog box with the 'General' tab active. The 'ID' field contains 'default'. The 'Port' field contains '9000'. The 'Connection Protocol' is a dropdown menu. The 'HA sync' checkbox is checked. The 'Select Listening Mode' is a dropdown menu showing 'Unsecured'. The 'Description' field is empty. The 'OK' and 'Cancel' buttons are at the bottom right.

Knowledge Center CMS Port Information

7. Optionally, you can add a secure listening port for authenticated users, secured connections, and secure chat. Click **Add**. The **Port Info** dialog opens.
 1. Enter *https* for the **ID** field.
 2. Enter the port . For instance, 9553.
 3. Enter *https* for the **Connection Protocol**.
 4. Choose **Secured** for the **Listening Mode**.
 5. Click **OK**.



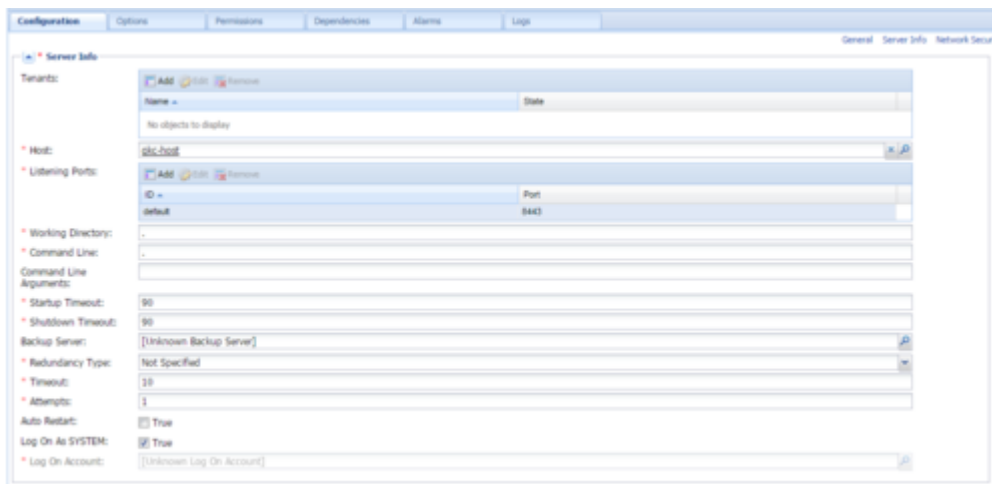
The 'Port Info' dialog box is shown with the 'General' tab selected. The fields are as follows:

Field	Value
ID	https
Port	9553
Connection Protocol	https
HA sync	<input checked="" type="checkbox"/> True
Select Listening Mode	Secured
Description	

Buttons: OK, Cancel

GKS CMS https port

8. Ensure the **Working Directory** and **Command Line** fields contain "." (period).



The 'Configuration' window is shown with the 'General' tab selected. The fields are as follows:

Field	Value
Tenants	None
Host	gks-host
Listening Ports	default (9443)
Working Directory	.
Command Line	.
Startup Timeout	90
Shutdown Timeout	90
Backup Server	[Unknown Backup Server]
Redundancy Type	Not Specified
Timeout	10
Attempts	1
Auto Restart	<input checked="" type="checkbox"/> True
Log On As SYSTEM	<input checked="" type="checkbox"/> True
Log On Account	[Unknown Log On Account]

Knowledge Center CMS Information

9. In the **Tenants** section, add a working tenant by clicking **Add**. Browse and choose the appropriate tenant in the pop-up dialog. Click **OK**.
10. Uncheck **Log On As SYSTEM**.
11. In **Log On Account** specify the user account that:
 - has the ability to view access groups (this is required if you use access groups to set privileges for your agents)
 - has **Knowledge.ADMINISTER** privileges and belongs to a Super Administrators access group (required for exporting configuration definitions)
 - has **Knowledge.AUTHOR** privilege (required for **scheduled synchronization**)
- Click **Save**.

- The **Confirmation** dialog for changing the application's port opens. Click **Yes**.

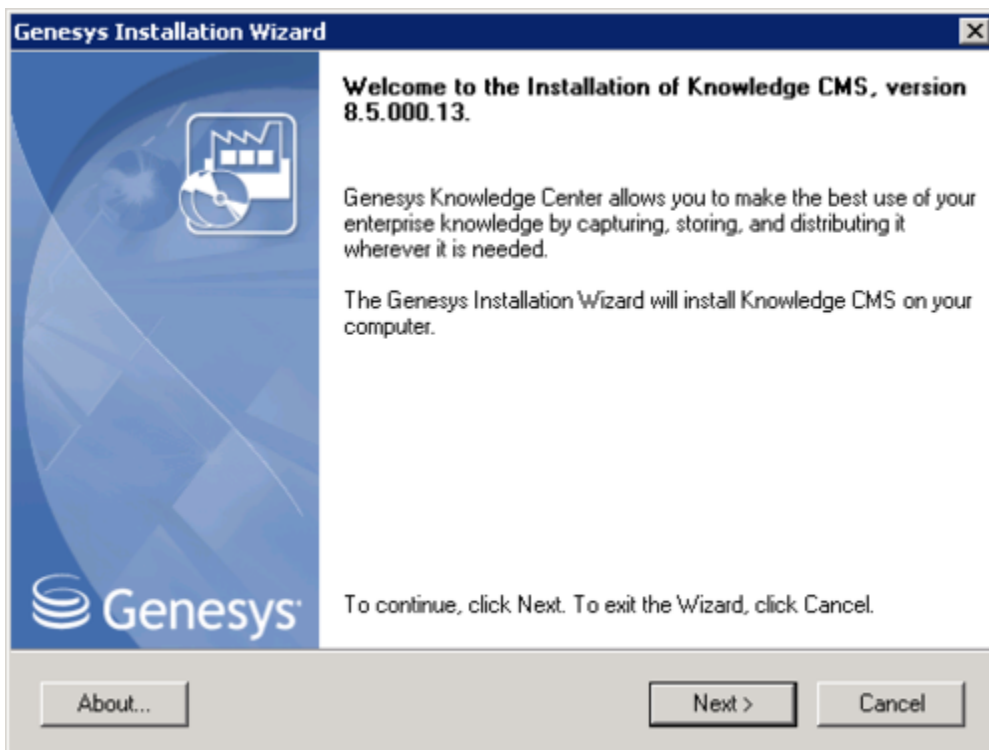
End

Installing the CMS

Windows Installation Procedure

Start

1. In your installation package, locate and double-click the *setup.exe* file. The Install Shield opens the welcome screen.



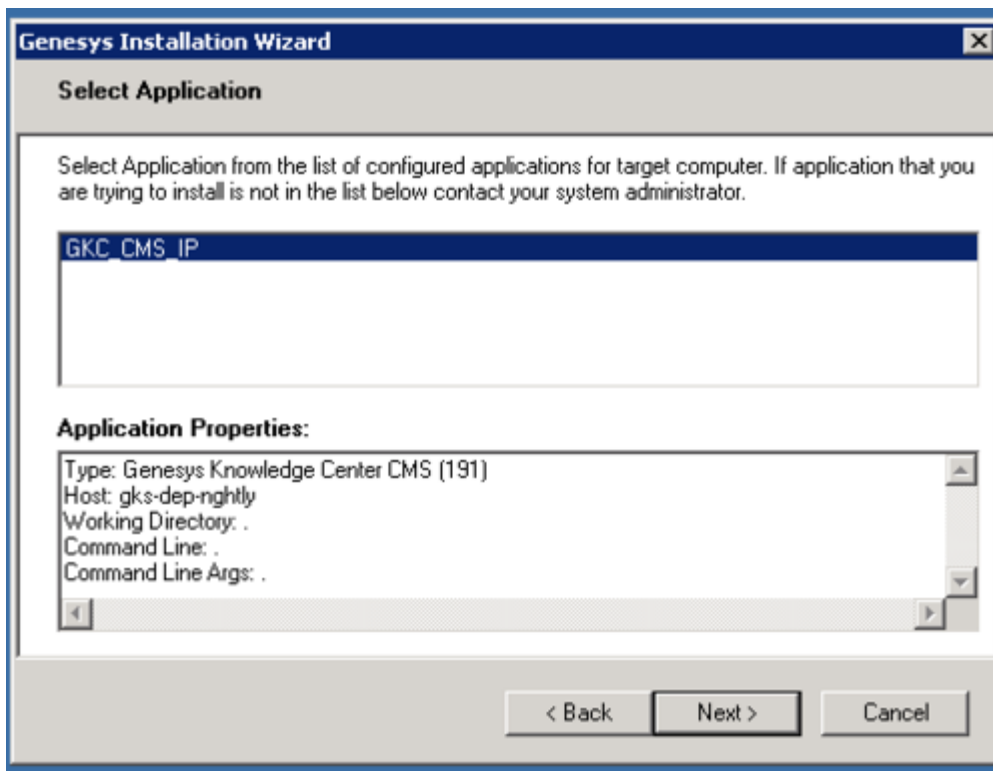
Knowledge Center CMS installation Window

2. Click **Next**. The **Connection Parameters to the Configuration Server** screen appears.

The screenshot shows a Windows-style dialog box titled "Genesys Installation Wizard" with a close button (X) in the top right corner. The main title bar is blue. Below the title bar, the dialog has a header section with the title "Connection Parameters to the Configuration Server". The main content area has a light gray background and contains the following text: "The parameters in the Host and User fields are required to establish a connection to Configuration Server." Below this, there are two sections: "Host" and "User". The "Host" section has a label "Specify the host name and port number for the machine on which Configuration Server is running." and two input fields: "Host name:" with the value "localhost" and "Port:" with the value "2020". The "User" section has a label "Specify your Configuration Server user name and password." and two input fields: "User name:" with the value "default" and "Password:" with a masked password represented by ten black dots. At the bottom of the dialog, there are three buttons: "< Back", "Next >", and "Cancel".

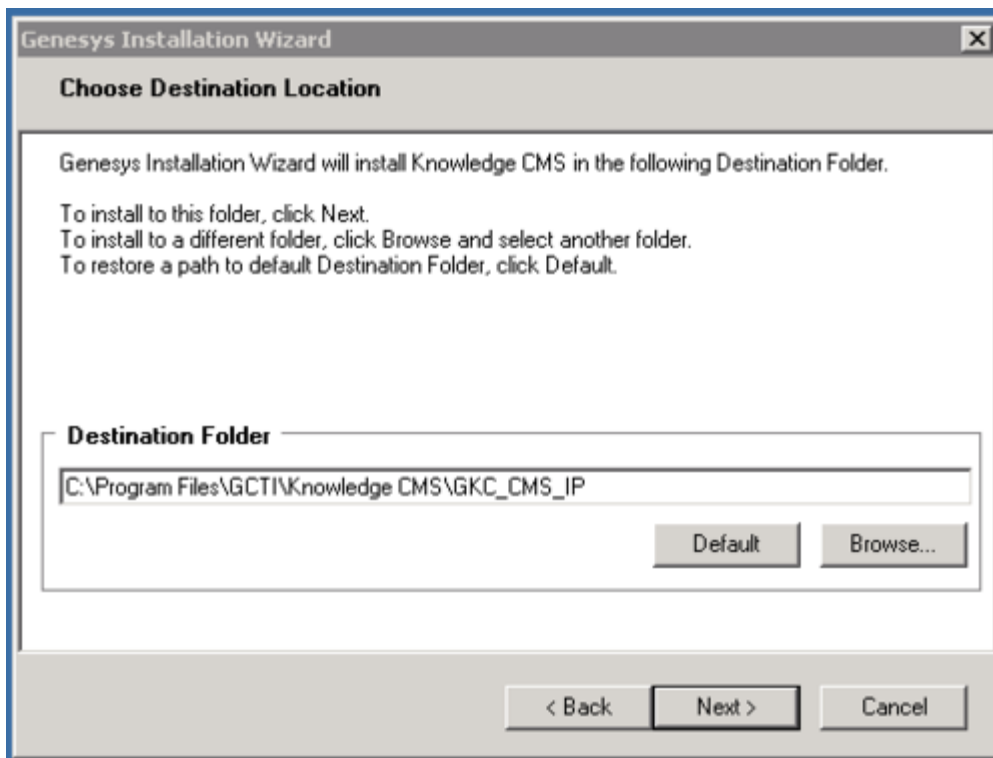
Knowledge Center CMS Connection Parameters

3. Under **Host**, specify the host name and port number where Configuration Server is running. (This is the main listening port entered in the **Server Info** tab for Configuration Server.)
4. Under **User**, enter the user name and password for logging in to Configuration Server.
5. Click **Next**. The **Select Application** screen appears.
6. Select the Knowledge Center CMS that you are installing. The **Application Properties** area shows the **Type**, **Host**, **Working Directory**, **Command Line executable**, and **Command Line Arguments** information previously entered in the **Server Info** and **Start Info** tabs of the selected application object.



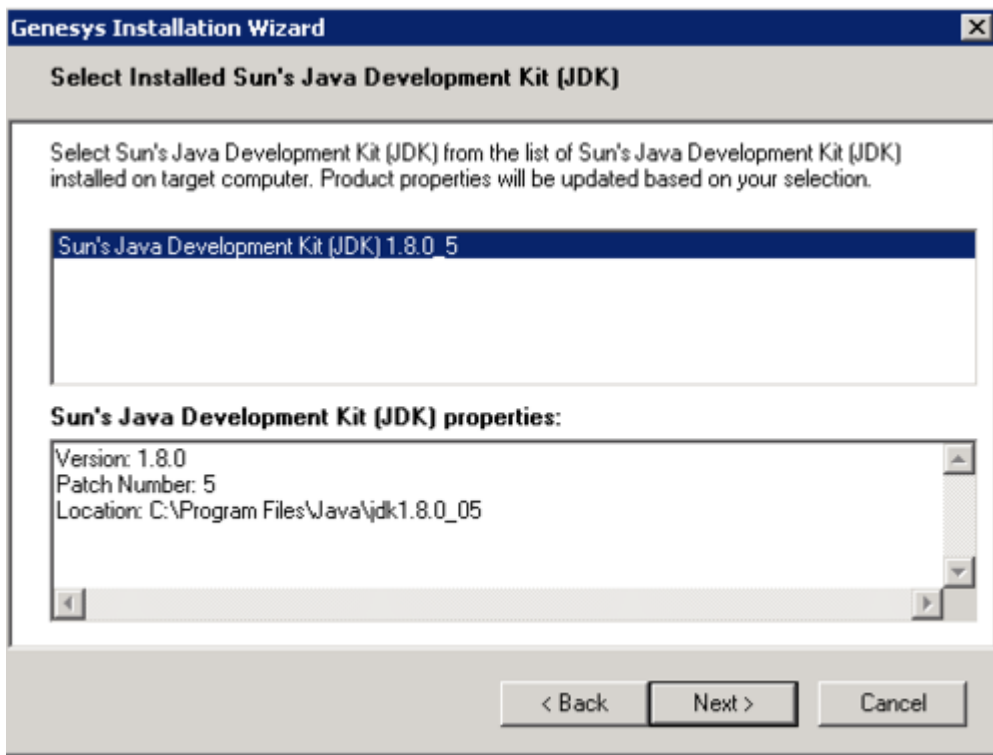
Selecting the Knowledge Center CMS Application

7. Click **Next**. The **Choose Destination Location** screen appears.
8. Under **Destination Folder**, keep the default value or browse for the desired installation location.



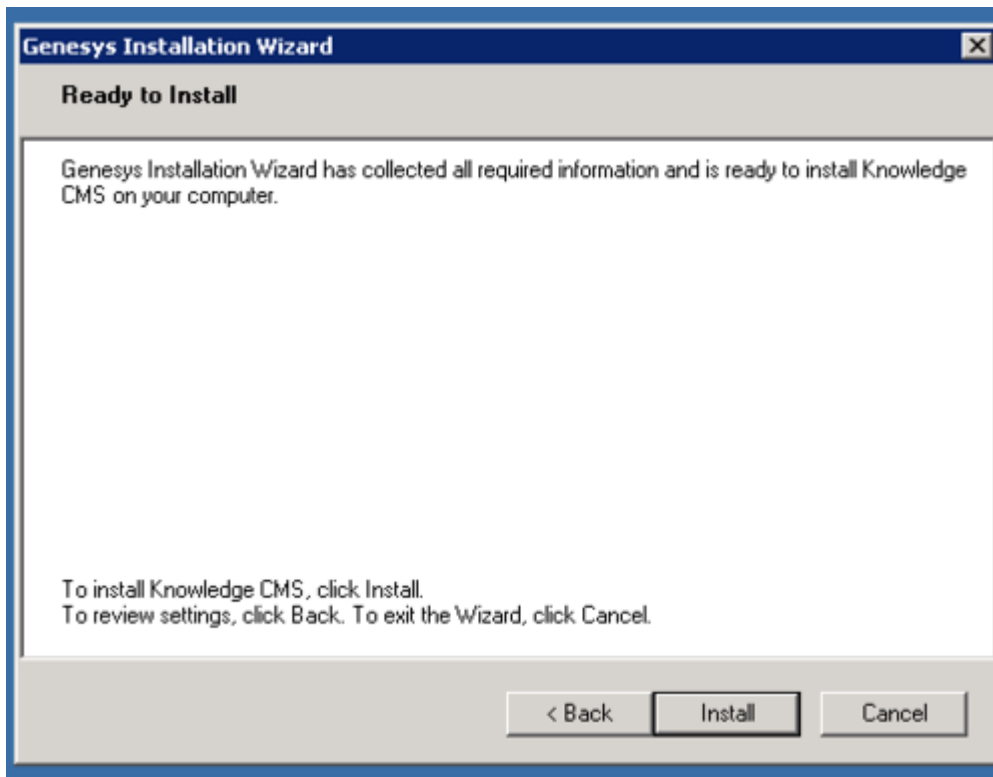
Choosing the Knowledge Center CMS Installation Destination

9. Click **Next**. Choose the appropriate version of the Java JDK.



Selecting the Knowledge Center CMS Java Version

10. Click **Next**. The **Ready to Install** screen appears.



Knowledge Center Knowledge Center is Ready to Install

11. Click **Install**. The Genesys Installation Wizard indicates it is performing the requested operation for the Genesys Knowledge Center CMS. When through, the **Installation Complete** screen appears.
12. Click **Finish** to complete your installation.
13. Inspect the directory tree of your system to make sure that the files have been installed in the location that you intended.

Important

The Windows service will not be automatically configured during installation of the CMS. To configure the Windows service, start *server.bat* with the following parameters: **server.bat install**. To run the server as service, comment out the (REM) APP_TYPE property in *senenv.bat* before installing the service.

End

Linux Installation Procedure

Start

1. Open a terminal in the CMS installation package, and run the *install.sh* file. The Genesys installation starts.
2. Enter the hostname of the host on which you are going to install.
3. Enter the connection information required to log in to the Configuration Server:
 1. **Hostname**—For instance, *demosrv.genesyslab.com*
 2. **Listening port**—For instance, *2020*
 3. **User name**—For instance, *demo*
 4. **Password**
4. If you have a backup Configuration Server, enter the Host name and Port.
5. If the connection settings are successful, a list of keys and Knowledge Center CMS applications is displayed.
6. Enter the key for the Knowledge Center CMS application that you created previously in Configuration Server.
7. Enter the full path to your installation directory and confirm that it is correct.
8. If the installation is successful, the console displays the following message:
Installation of Genesys Knowledge CMS, version 8.5.x has completed successfully.

End

Configuring the CMS

The Knowledge Center Server includes an embedded Jetty server. After installation, you can carry out your initial configuration by creating a *work* directory for temporary Jetty files inside the *./server* folder.

Configure Required CMS Access Options

Genesys Knowledge Center supports the following privileges to restrict agent access:

- **Knowledge.CMS.Document.Author**—create, edit, or delete documents
- **Knowledge.CMS.Category.Author**—create, edit, or delete categories
- **Knowledge.CMS.Approver**—approve documents and categories, and export data
- **Knowledge.CMS.Administrator**—create, edit, or delete knowledge bases

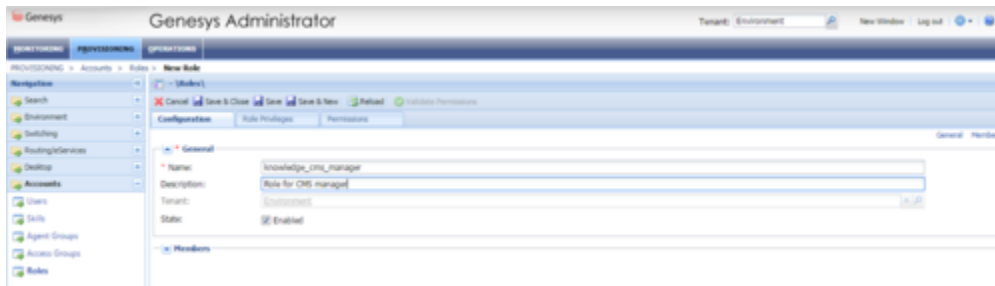
Important

Only agents who have both **Knowledge.CMS.Document.Author** and **Knowledge.CMS.Category.Author** privileges can successfully import data from XML files.

To configure the appropriate privileges for an agent:

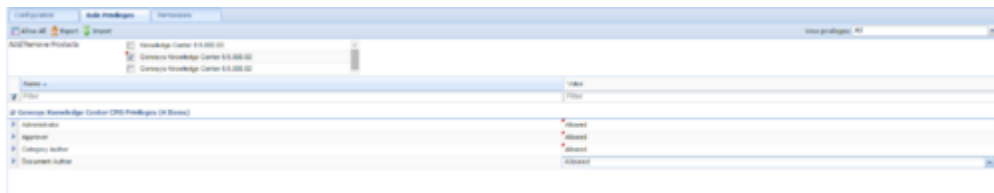
Start

1. Go to **Provisioning > Accounts > Roles**.
2. In the taskbar, click **New** to create a new object.
3. Set the name of the role in the **General** section.



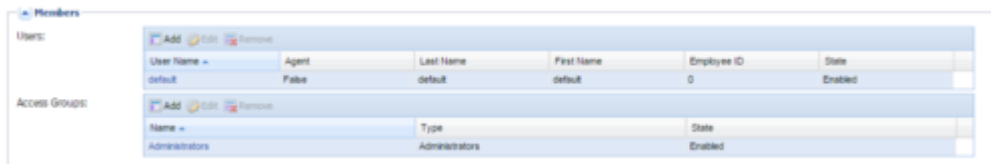
Knowledge Center CMS Access Roles

4. Go to the **Role Privileges** tab and select the set of roles for Genesys Knowledge Center.
5. Open the Genesys Knowledge Center CMS privileges list.
6. Set the appropriate privileges to **Allowed**.



Setting Knowledge Center CMS Access Privileges

7. Go back to the **Configuration** tab.
8. In the **Members Section**, add the appropriate Agent by clicking the **Add** button.



Knowledge Center CMS Members Section

9. In the **Tenants** section, add a working tenant by clicking **Add**. Browse and choose the appropriate tenant in the pop-up dialog. Click **OK**.
10. Save and Close.

End

Installing and Using the Administrator Plugin

Installing the Knowledge Center Plugin for Administrator

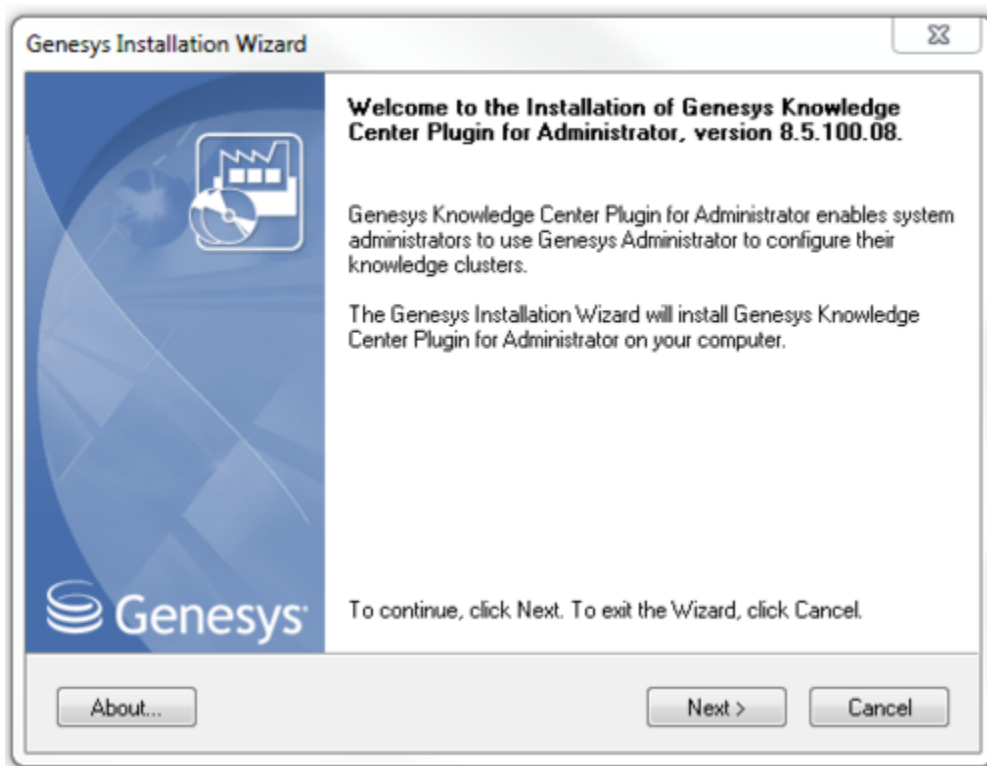
Prerequisites

- Genesys Administrator must have been installed, but should be stopped before installing the plugin
- If the Administrator Plugin was previously installed on the current host, manually remove the previous version from the `/plug-ins` folder in the Genesys Administrator installation directory

Windows Installation Procedure

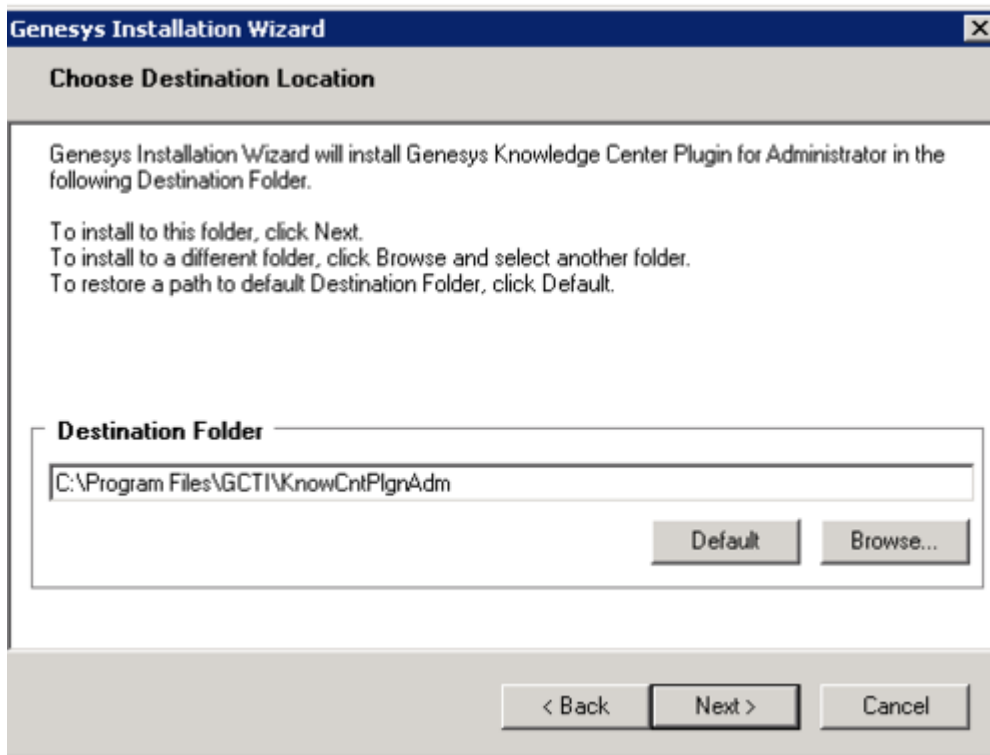
Start

1. In your installation package, locate and double-click the **setup.exe** file. Install Shield opens its welcome screen.



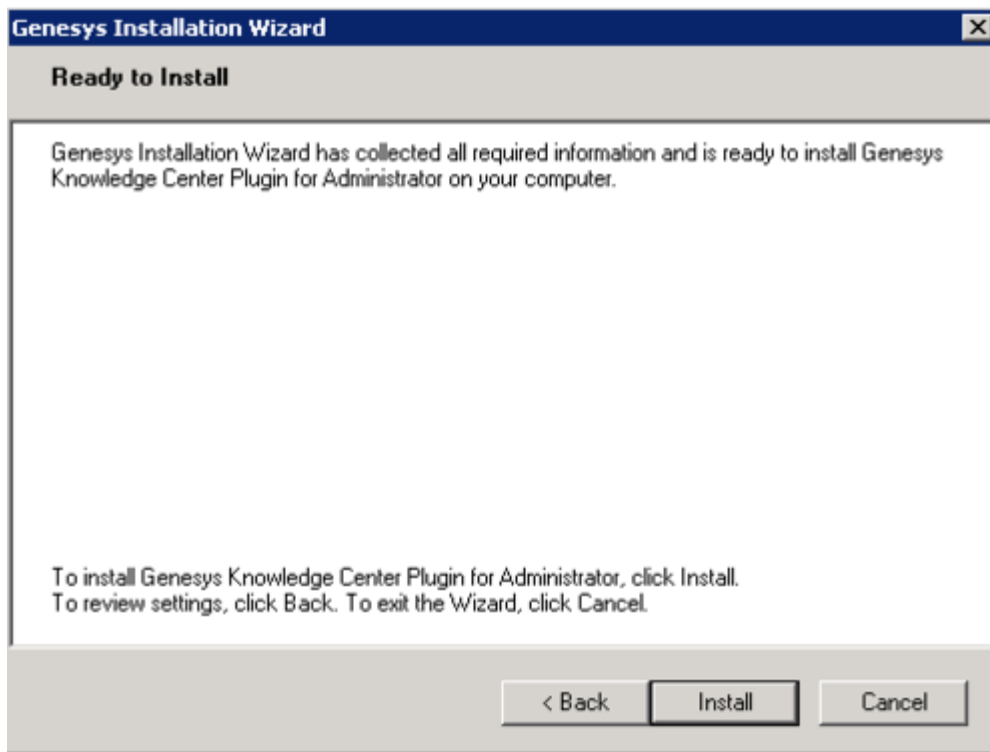
Knowledge Center Administrator Plugin Install Shield Window

2. Click **Next**. The **Choose Destination Location** screen appears.



Knowledge Center Administrator Plugin Destination Window

3. Under **Destination Folder**, keep the default value or browse to the desired installation location. Click **Next**.
4. Click **Install**. The Genesys Installation Wizard indicates it is performing the requested operation for the Backend Server. When it has finished, the **Installation Complete** screen appears.



Knowledge Center Administrator Plugin Installation Complete

5. Click **Finish** to complete your installation.
6. Inspect the directory tree of your system to make sure that the files have been installed in the location that you intended.
7. *gax-plugin-knowledge.jar* should be added as a Genesys Administrator plugin.
8. Restart Genesys Administrator.

End

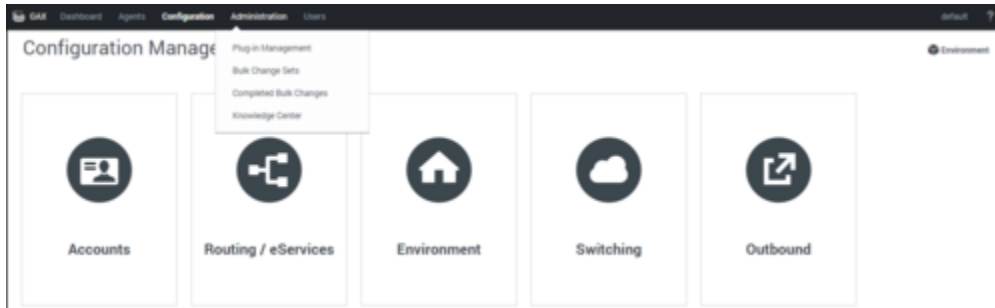
Linux Installation Procedure

Start

1. Open a terminal in the Genesys Knowledge Center Plugin for Administrator IP, and run the *install.sh* file. The Genesys Installation starts.
2. Enter full path to the GAX installation directory.
3. Enter full path to your installation directory for the plugin and confirm it.
4. If the installation is successful, the console displays the following message: Installation of Genesys Knowledge Center Plugin for Administrator, version 8.5.x has completed successfully.
5. *gax-plugin-knowledge.jar* should be added as a Genesys Administrator plugin.
6. Restart Genesys Administrator.

End

A **Knowledge Center** item should appear under the Administration menu.



Knowledge Center in Administrator Menu

Providing access to Knowledge Center Plugin for Administrator

Important

Users must have the next privilege in order to use the Administrator plugin.

- Knowledge. ADMINISTER — Enables access to the Knowledge Center Plugin for Administrator tab in Genesys Administrator
- To save a created configuration the user should at least belong to the Administrators Access Group

To configure the appropriate role for an agent:

Start

1. Go to **Provisioning > Environment > Application Templates**.
2. In the **Tasks** panel, click **Upload Template**.
3. In the **Click 'Add' and choose application template (APD) file to import** window, click **Add**.
4. Choose the application template (APD) file from the import window and click **Add**.
5. Browse to the Knowledge_Center_GAX_Plugin_851.apd file available in the templates directory of your installation CD. The **New Application Template** panel opens.
6. Click **Import Metadata**.
7. Click **Add** and select the Knowledge_Center_GAX_Plugin_851.xml file.
8. Click **Open**.
9. Information from the metadata file will be added to the template and the appropriate privilege will be added into the framework.

10. Save and Close.
11. Go to **Provisioning > Accounts > Roles**.
12. In the taskbar click **New** to create a new object.
13. Set the name of the role in the **General** section.
14. Go to the **Role Privileges** tab, and select the set of roles for Genesys Knowledge Center.
15. Open the **Genesys Administrator privileges** list and select the Genesys Knowledge Center Plug-In Privileges section.
16. Set the appropriate privileges as allowed.
17. Go back to the **Configuration** tab.
18. Add the appropriate **Agent** to the **Members** section by clicking the **Add** button.
19. Save and Close.

End

Managing Knowledge Bases

In order to use Knowledge Center Server you need to create at least one knowledge base in the Knowledge Center Cluster application, using the Knowledge Center Plugin for Administrator. This section describes the structure and specific options you need in order to create an index for this knowledge base in Knowledge Center Server.

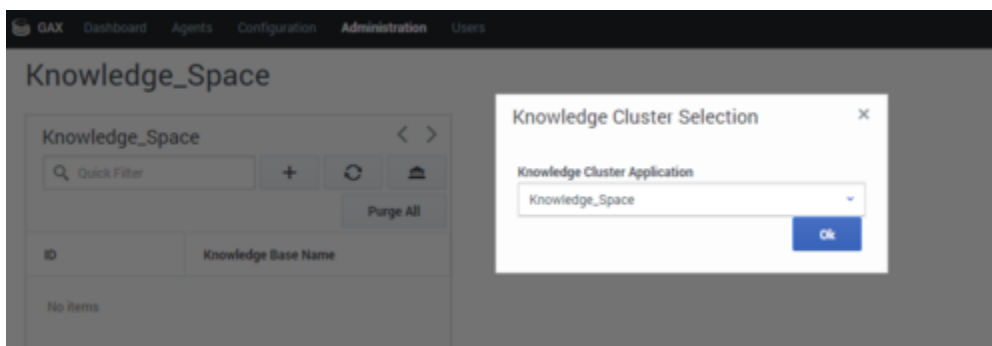
Selecting the Knowledge Center Cluster Application

Start

1. Log in to Genesys Administrator and navigate to the **Administration > Knowledge Center** menu item.



2. Using the  button, open the menu for **Select Knowledge Cluster**. Select the appropriate cluster from the drop-down and click the **Ok** button. A list of the knowledge bases that have been defined for this cluster will be displayed.



Selecting a Knowledge Cluster

End

Creating a Knowledge Base

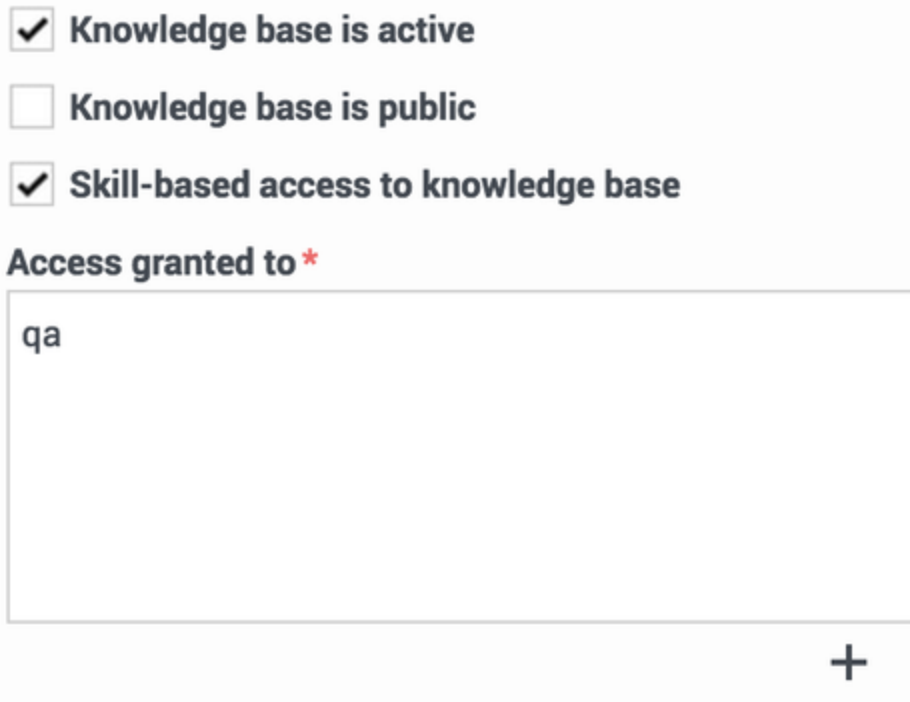
Start

1. Click the + button. A panel with the main knowledge base parameters will be displayed. Fill in the following fields:
 - **ID**—The ID should only contain numbers, lower-case Latin letters, and underscores, with a maximum length of 50 characters. The limitation to lower-case letters is because ElasticSearch is case-insensitive and will therefore render all names as lower-case.
 - **Name**—Maximum length is characters
 - **Description**—optional
 - Select the default knowledge base language.
 - Make the knowledge base public or private. (If the knowledge base is made public, it will be visible to all users, whether or not they are authorized.)

Important

For private knowledge bases you can specify whether the knowledge base should be available to all of your agents or only to the agents that have one of the specified skills. In the case where you have specified several skills for the knowledge base, the agent needs to have at least one of them to access the knowledge base. Skill level does not influence ability to access the knowledge base.

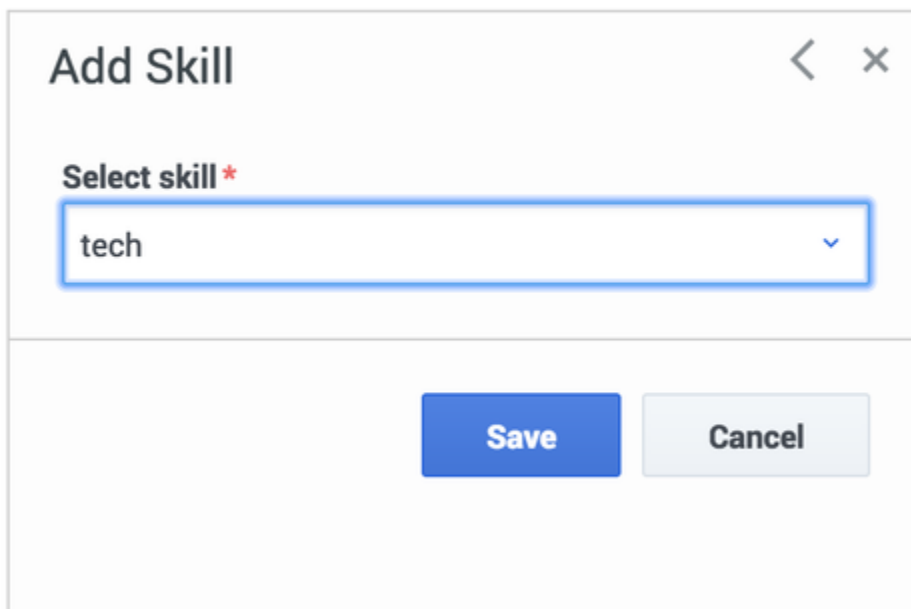
- Make the knowledge base active or inactive. If you un-check **Knowledge Base is active** neither your customers nor your agents will be able to search for information in that knowledge base. Authors, agents with the privilege **Knowledge.Author** and administrators with the privilege **Knowledge.Admin**, can still use the base to prepare content stored in it.
- If you mark the knowledge base as public it will be accessible to both your agents and customers. A knowledge base that is not marked public will be treated as a private one and will only be accessible by your agents.



A screenshot of a configuration interface for a knowledge base. It features three checkboxes: 'Knowledge base is active' (checked), 'Knowledge base is public' (unchecked), and 'Skill-based access to knowledge base' (checked). Below these is a section titled 'Access granted to *' with a text input field containing 'qa'. A plus sign icon is located at the bottom right of the input field.

Specifying Access To Your Knowledge Base

- You can make your private knowledge bases (when **Knowledge base is public** is not check-marked) available to only a subset of your agents by selecting **Skill-based access to knowledge base**. If you choose to make the knowledge base accessible to an agent with specific skills, you will need to select the skills that will grant an agent access to the knowledge base.



A screenshot of a dialog box titled 'Add Skill'. It has a back arrow and a close 'X' button in the top right corner. The main content area has a label 'Select skill *' above a dropdown menu that currently shows 'tech' with a downward arrow. At the bottom, there are two buttons: 'Save' (blue) and 'Cancel' (light gray).

Making a Knowledge Base Accessible by Skill

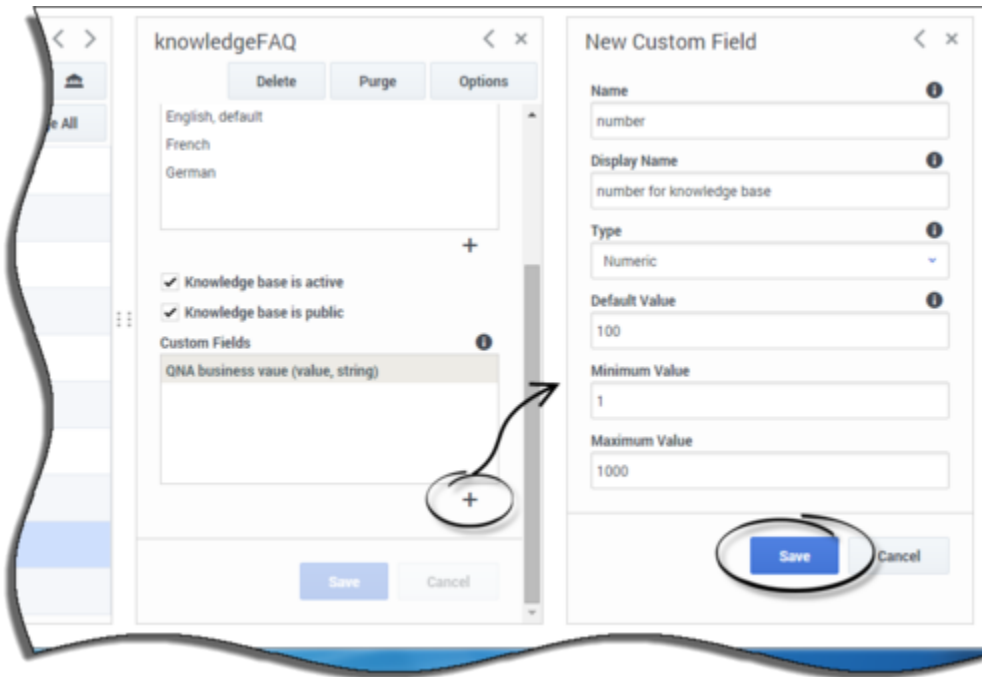
2. Click **Save**. The knowledge base will be created.

End

Creating Custom Fields

Start

1. Click the + sign under the **Custom Fields** section. The **New Custom Field** panel will be displayed.



Creating a Custom Field

2. To define a custom field, fill in the following information:
 - **Name**—Should consist only of numbers, Latin letters and underscores, with a maximum length of 50 characters.
 - **Display name**
 - Select the type of field
 - For **String** fields define:
 - Default value (optional)
 - If the field can be left empty, set the check box to **Allow empty**
 - For **Numeric** fields define:
 - Default value (optional)
 - Minimum value (optional)

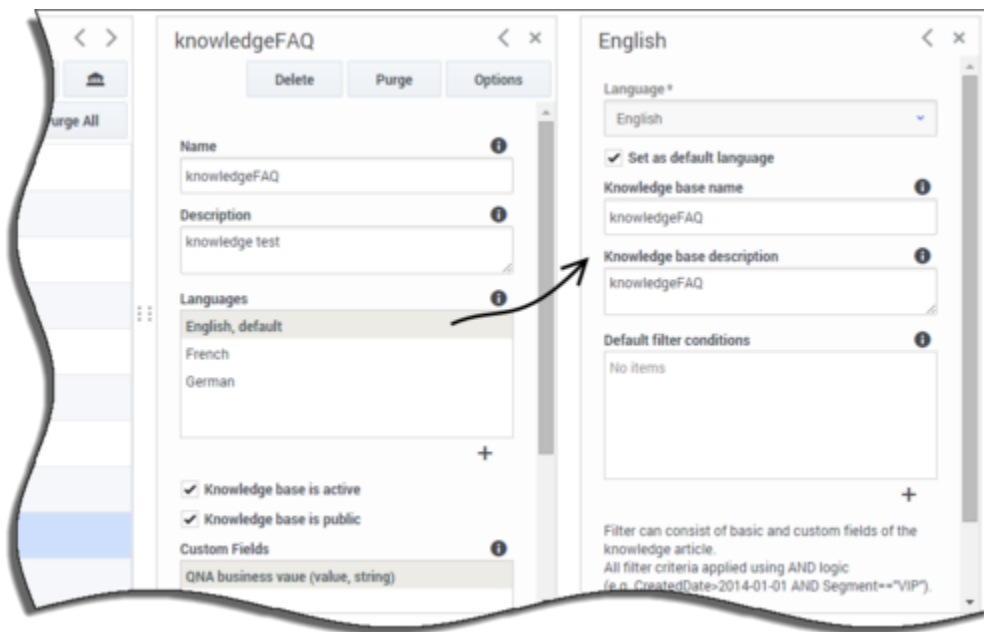
- Maximum value (optional)
- For **DateTime** fields define:
 - Default value (optional; format should be yyyy-MM-dd HH:mm:ss)
- Click **Save** to save your changes.

End

Adding Language-specific Information

Start

1. Click the **English, default** row in the **Languages** section. A panel with language-specific settings will be displayed.



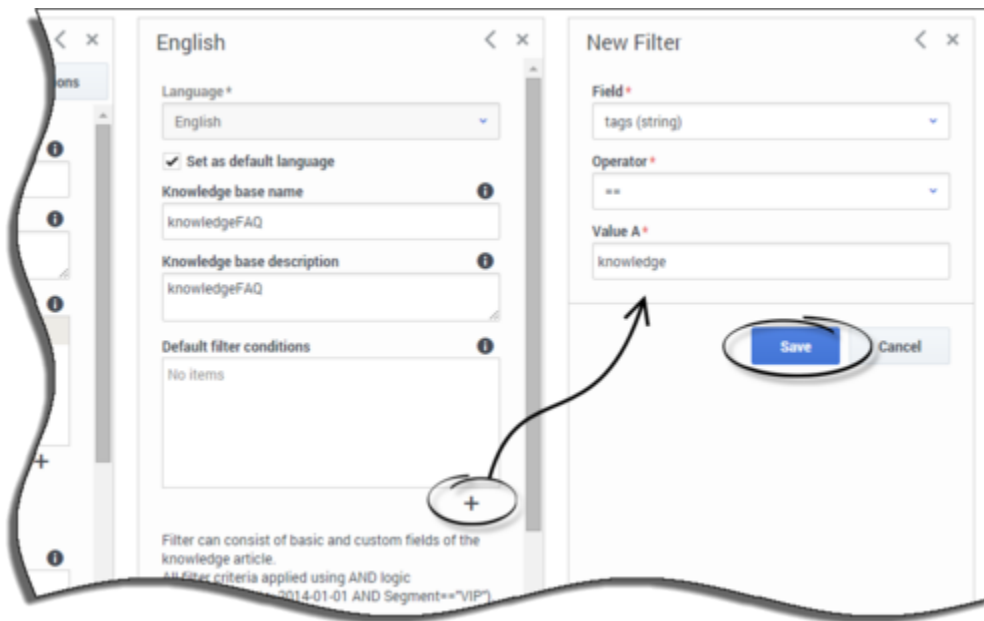
Adding Language-Specific Information

2. You can define the following parameters in this section:
 - A localized knowledge base name
 - A localized knowledge base description
 - Whether or not the selected language is the default
 - Default filter conditions
- To create a default filter condition click on the + under the **Default filter conditions** section and fill in the appropriate mandatory fields:
 - Select the appropriate field (custom or basic)

- Select a filter operator
- Fill in the values for the filter criteria

Important

All filter criteria are applied using AND logic. For example,
CreatedDate>2014-01-01 00:00:00 AND Segment=="VIP".



Default Filter

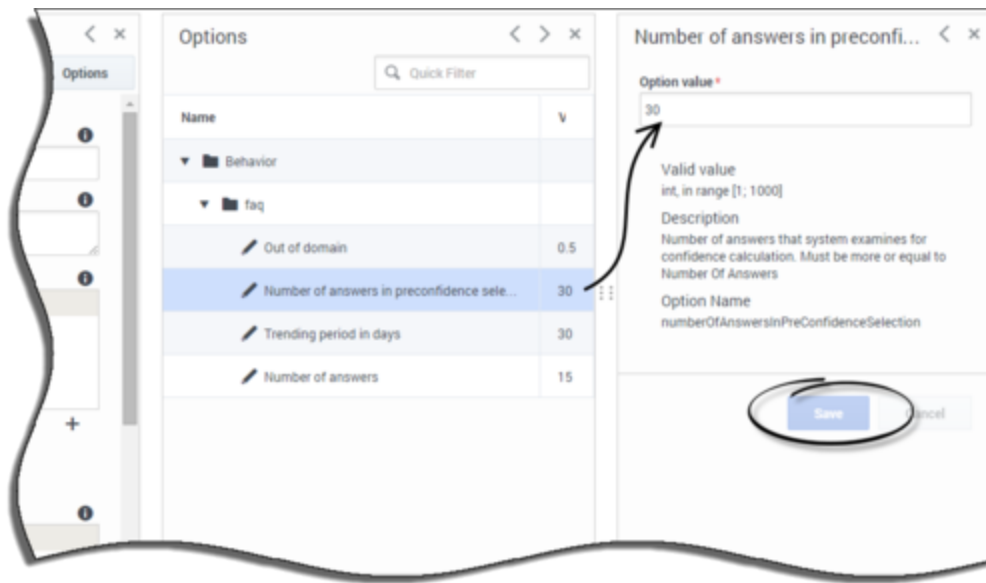
4. Click the **Save** button

End

Editing Knowledge Base Options

Start

1. To edit the options for a particular knowledge base, click the **Options** button and then click the appropriate option to edit its value. The options are initialized with their default values.



Editing Knowledge Base Options

2. Enter the new option value and click the **Save** button.

End

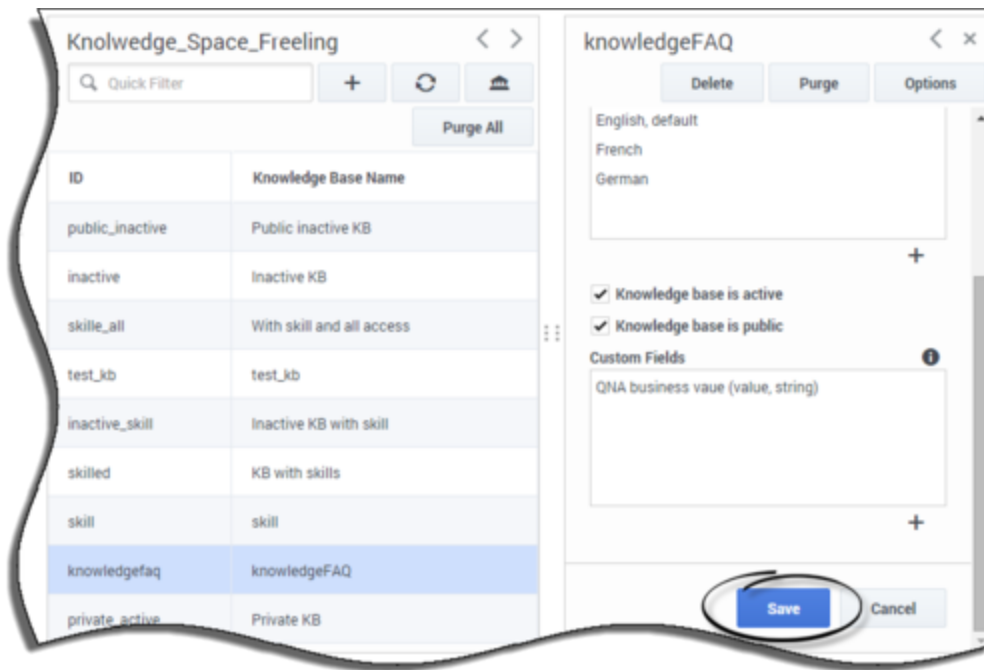
Important

It is not recommended to set the out-of-domain value higher than 0.75 as it represents an exact match of the question with no feedback accumulated for the query. The optional setting is 0.5 (default value).

Editing a Knowledge Base Definition

Start

1. Select a knowledge base from the list.



Editing Knowledge Base Definition

2. Edit the knowledge base definition and click the **Save** button.

End

Deleting a Knowledge Base Definition

Start

1. Select a knowledge base from the list.
2. Press the **Delete** button and confirm the action.

End

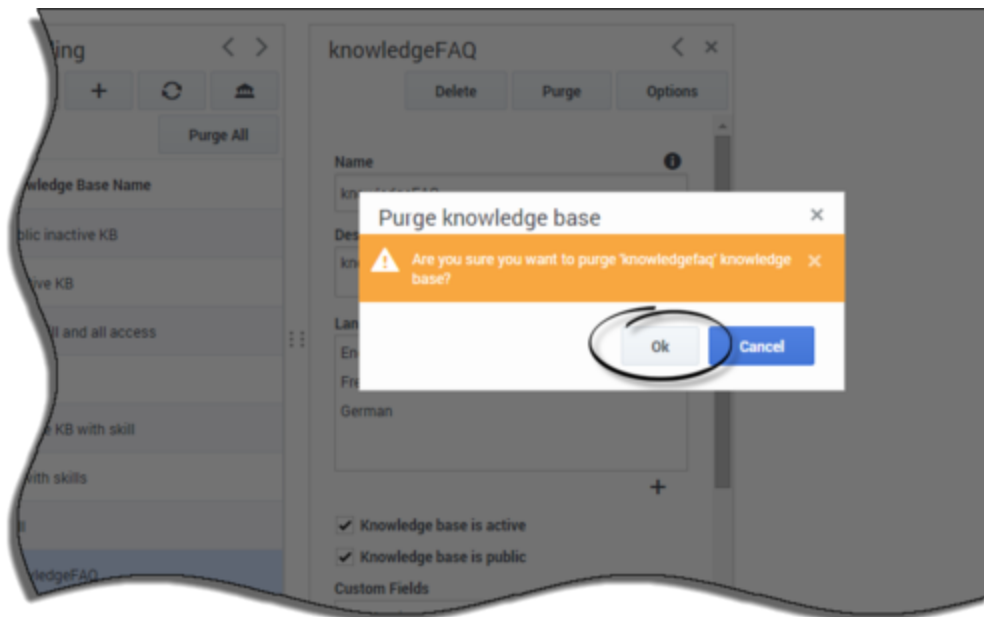
Purging Knowledge Bases

Prerequisites

- The Administrator user must have **Knowledge.ADMINISTER** privileges
- You must create and select a Knowledge Center Cluster application

Start

1. To purge a particular knowledge base, select it from the list, press the **Purge** button, and confirm the action.



Purging a Knowledge Base

2. To purge all knowledge bases, use the **Purge All** button.

End

Installing the Pulse Plugin

The Genesys Knowledge Center Plugin for Pulse provides access to Knowledge Center Server statistics such as KPI, user activity, trending topics, like and dislike trends, and activity types.

Install Genesys Knowledge Center Plugin for Pulse

Components required for Pulse plugin come pre-integrated into every deployment of Genesys Knowledge Center Server. So you do not need any additional steps to install them, please proceed directly to the configuration.

Important

World map widget requires access to the internet to generate a map image. Public map services are used for this purpose. If you would like to limit access to the internet you can remove this widget from the dashboard or use your own generation service (in this case, a compatible tile service provider must be used).

Warning

Since July 11, 2016 Map Quest had changed their terms of the service for tile service they provide. It resulted in an error message displayed instead of the map in the Map widget of dashboard.

To restore the map functionality you need to make the following changes in **<server installation folder>/server/plugins/gkc-dashboard/_site/app/panels/bettermap/module.js** :

Old code:

```
L.tileLayer('http://otile1.mqcdn.com/tiles/1.0.0/map/{z}/{x}/{y}.jpg', {
  attribution: '"Data, imagery and map information provided by MapQuest, '+
  'OpenStreetMap <http://www.openstreetmap.org/copyright> and contributors,
  ODbL',
```

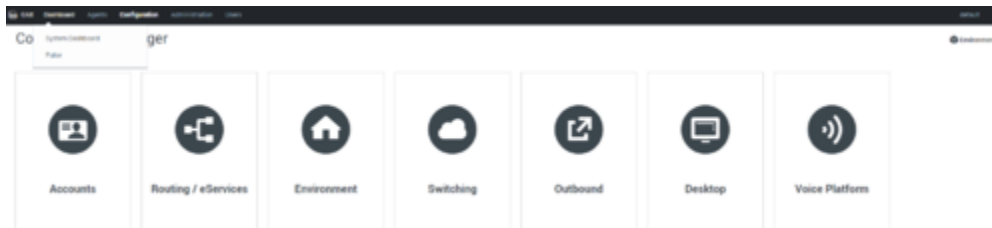
New code:

```
L.tileLayer('http://a.tile.openstreetmap.org/{z}/{x}/{y}.png', {
  attribution: 'Map data © <a href="http://openstreetmap.org">OpenStreetMap</a>
  contributors, ' +
  '<a href="http://creativecommons.org/licenses/by-sa/2.0/">CC-BY-SA</a>',
```

Configure Genesys Knowledge Center Plugin for Pulse

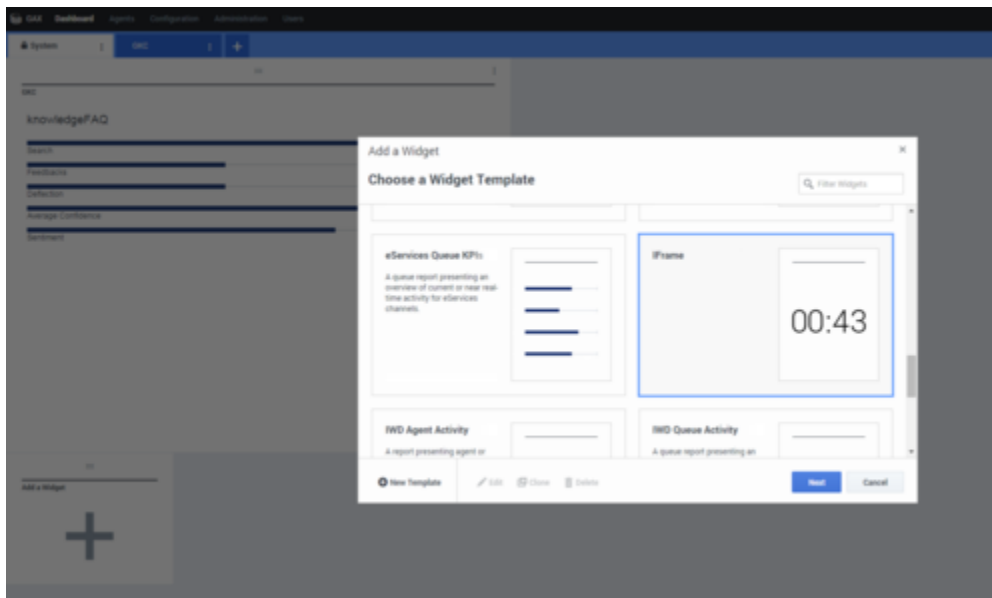
Start

1. Log into Genesys Administrator.
2. Go to **Dashboard > Pulse**.



Selecting the Pulse Dashboard options in Genesys Administrator

3. Click **Add a Widget**.
4. Select the **IFrame** widget type.



Adding a Pulse iFrame widget

5. Set the name of the widget.

Add a Widget

Display Options

Widget Title *
IFRAME

Size 1 X 2

☐ Allow resize

Widget refresh rate
60 seconds

Dashboard Widget URL
http://example/

Widget Preview

IFRAME

Complete URL by hitting Enter or moving to another field.

Previous Finish Cancel

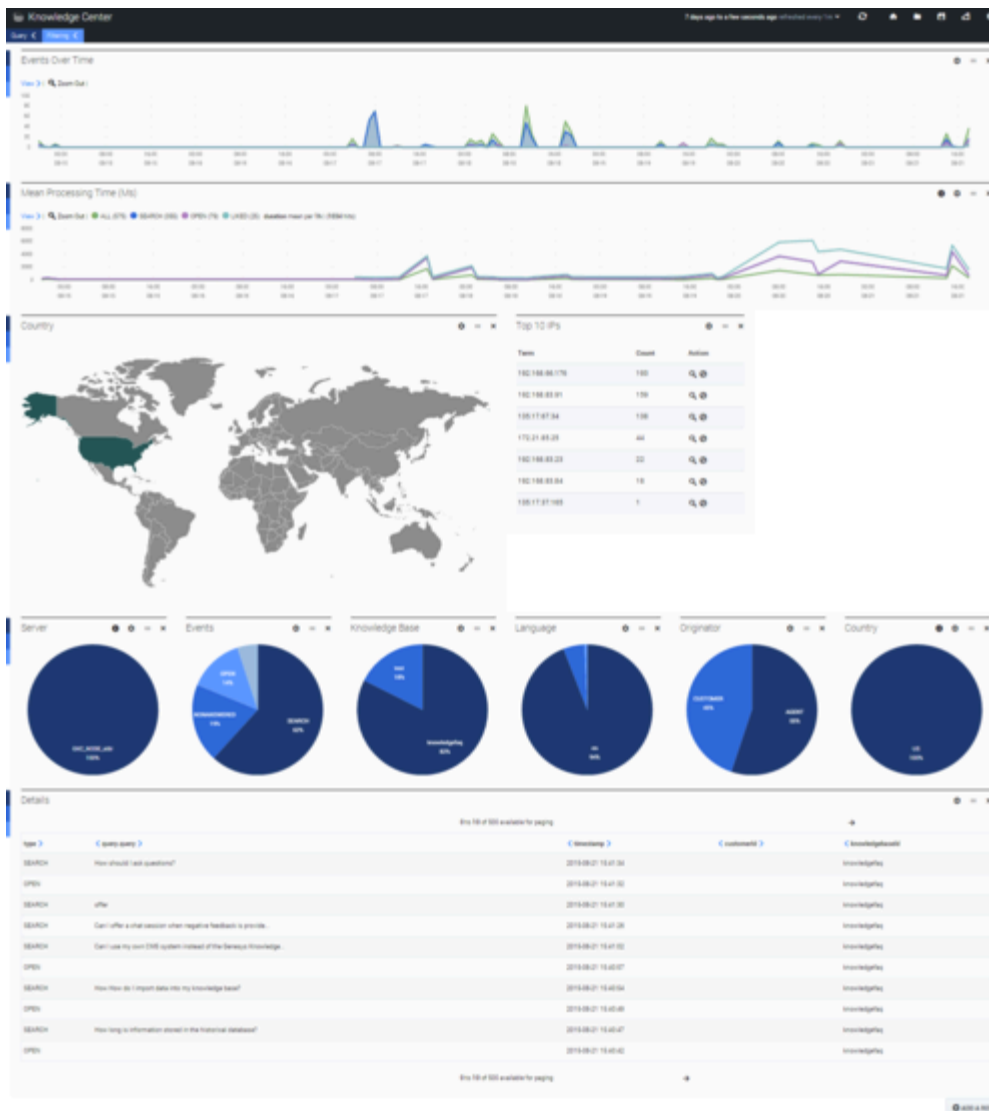
Setting the Pulse widget options

6. Set the widget URL to: `http://<host>:<es_port>/_plugin/gkc-kpi/kb/<knowledge_base_id>/lang/<en>` (see [Knowledge Center Pulse Plugin Configuration Options](#) for more information about parameters).
7. Set the Maximized widget URL. You can set it to the Default Dashboard (`http://<host>:<es_port>/_plugin/gkc-dashboard/#/dashboard/file/default.json`) or the Performance Dashboard (`http://<host>:<es_port>/_plugin/gkc-dashboard/#/dashboard/file/performance.json`).
8. Click **Finish**.

Installing the Pulse Plugin



Pulse Dashboard Widget



Pulse Performance Dashboard Widget

You have successfully added a widget for accessing Knowledge Center statistics.

End

Knowledge Center Pulse Plugin Configuration Options

You can customize the KPI widget by adding the bolded parameters to the URL:

`http://<host>:<es_port>/_plugin/gkc-kpi/
kb/<knowledge_base_id>/lang/<en>?timeframe=<timeframe>`

- `/kb/<knowledge_base_id>`—Set the appropriate knowledge base name
- `/lang/<en>`—Choose Knowledge Base's Language representation
- `timeframe=<timeframe>`—Select KPI's timeframe, for example *now-1M*

Important

Timeframe expression must start with an “anchor” date - **now** and follow by a math expression starting from - **and / (rounding)**. The units supported are **y** (year), **M** (month), **w** (week), **d** (day), **h** (hour), **m** (minute), and **s** (second). For example, *now-1h*, *now-1h-1m*, *now-1h/d*.

Installing the Workspace Desktop Edition Plugin

Installing the Plugin for Workspace Desktop Edition

Agents can use the Knowledge Center Plugin for Workspace Desktop Edition (WDE) to access knowledge-related information right from their desktop. For example, if a customer asks a question using a chat widget and the corresponding interaction is routed to an agent, Knowledge Center can execute a pre-populated search based on data attached to the new interaction, as well as displaying the customer's search history and providing the agent with full access to the knowledge base access. And if the customer has not authorized during their search, the agent can link their session history to that customer's ID to access their full history while working with the interaction. To use this plugin complete the procedures below, in order.

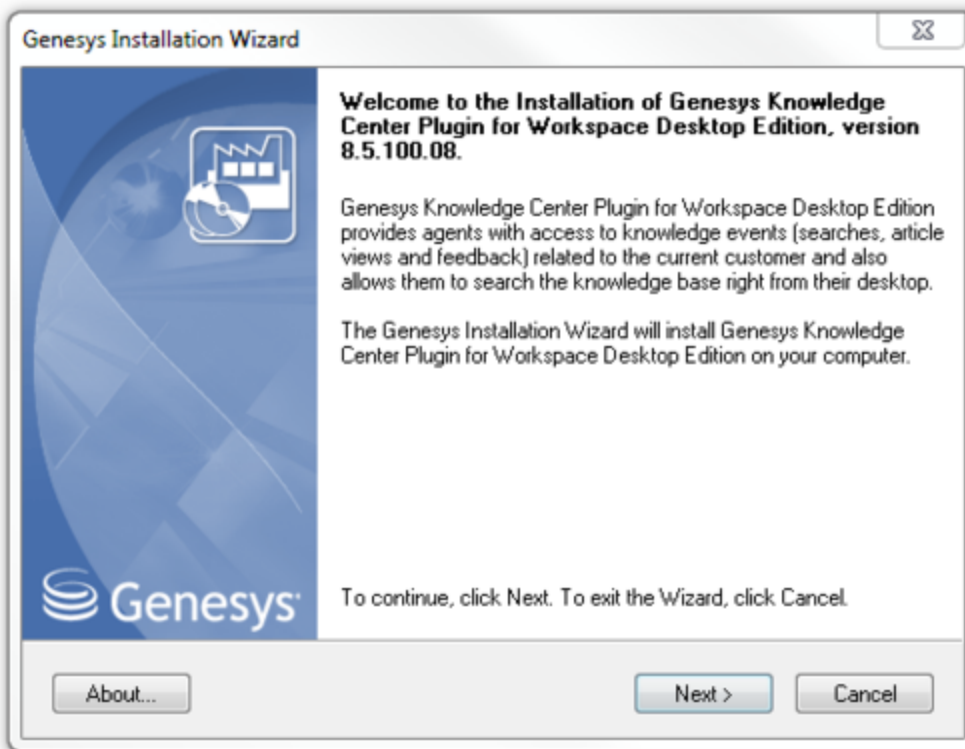
Installing the Plugin for Workspace Desktop Edition

Prerequisites

Workspace Desktop Edition must be installed and configured to work with voice or media interactions.

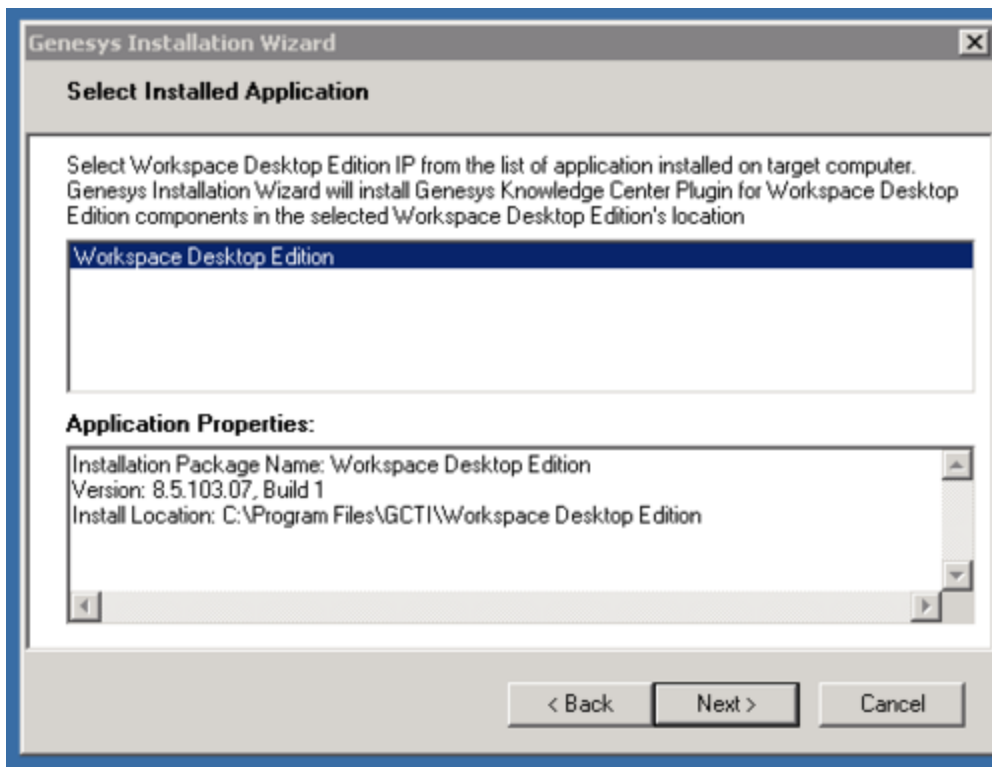
Start

1. In your installation package, locate and double-click the **setup.exe** file. The Install Shield opens the welcome screen.



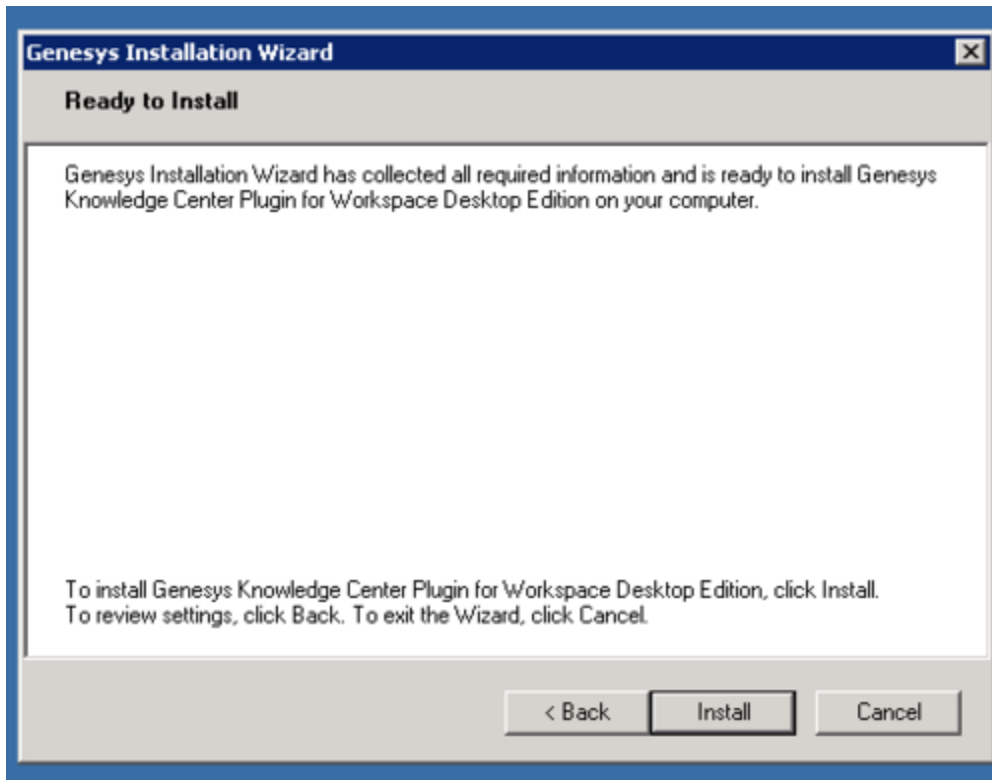
Knowledge Center WDE Plugin—Install Shield Screen

2. Click **Next**. The **Select Installed Application** screen appears.
3. Select the installed Workspace Desktop Edition Application for which you want to install the plugin. The **Application Properties** area shows the **Type**, **Host**, **Working Directory**, **Command Line executable**, and **Command Line Arguments** information previously entered in the Server Info and Start Info tabs of the selected Application object.



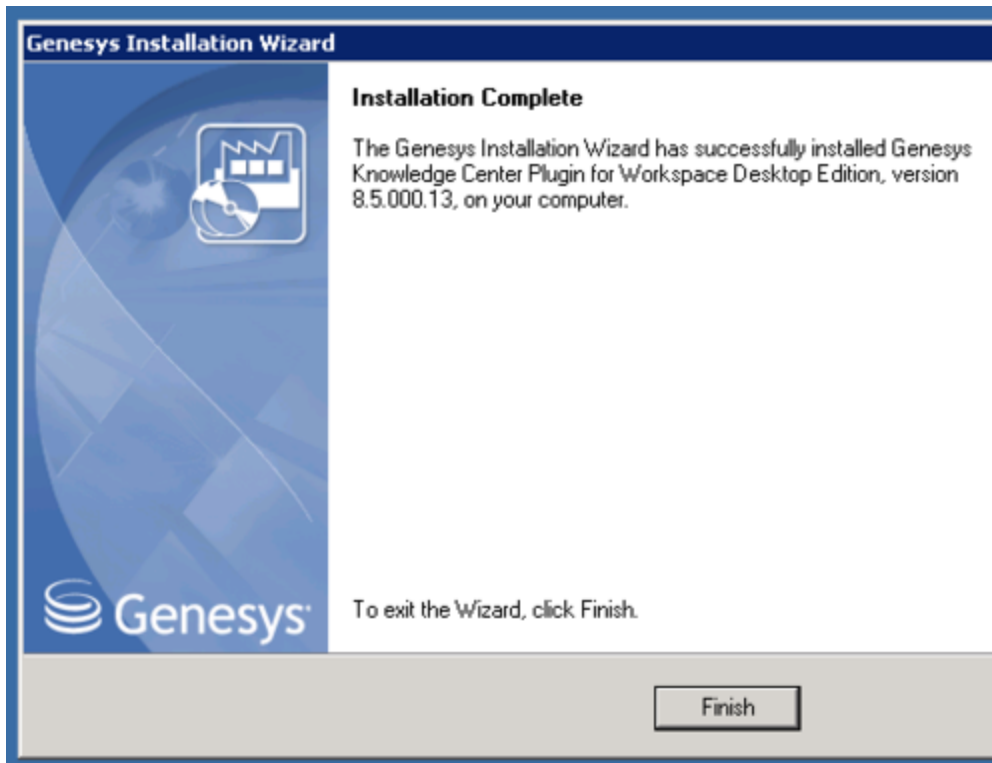
Select Installed Workspace Desktop Edition Application

4. Click **Next**. The **Ready to Install** screen appears.



Knowledge Center WDE Plugin—Ready to Install

5. Click **Install**. The Genesys Installation Wizard indicates it is performing the requested operation for Backend Server. When through, the **Installation Complete** screen appears.



Knowledge Center WDE Plugin—Installation Complete

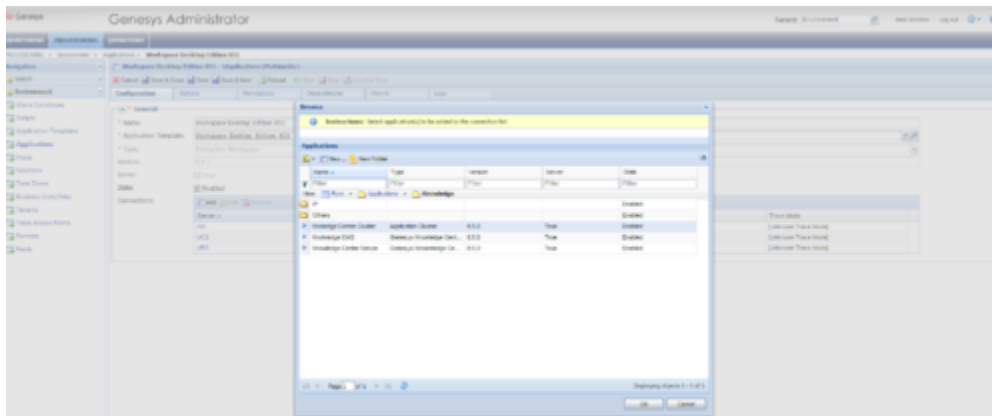
6. Click **Finish** to complete your installation.
7. Inspect the directory tree of your system to make sure that the following files have been installed in the location that you intended:
 - *GWEInstallationFolder\Genesyslab.Desktop.Modules.Knowledge.dll*
 - *GWEInstallationFolder\Genesyslab.Desktop.Modules.Knowledge.module-config*
 - *GWEInstallationFolder\Genesyslab.Desktop.Modules.Knowledge.pdb*
 - *GWEInstallationFolder\Newtonsoft.Json.dll*
 - *GWEInstallationFolder\RestSharp.dll*
 - *GWEInstallationFolder\System.Net.Http.Formatting.dll*
 - *GWEInstallationFolder\Language\Genesyslab.Desktop.Modules.Knowledge.en-US.xml*

End

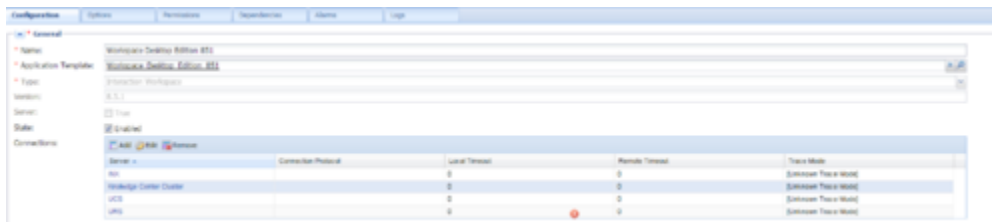
Configuring the WDE Application to work with the WDE Plugin

Add the Knowledge Center Cluster to Your WDE Connections

1. If your Workspace Desktop Edition application form is not open in Genesys Administrator, navigate to **Provisioning > Environment > Applications**. Select the application defined for the Workspace Desktop Edition and click **Edit...**.
2. In the **Connections** section of the **Configuration** tab, click **Add**. The **Browse for applications** panel opens. Select the **Knowledge Center Cluster application**, then click **OK**.



Knowledge Center WDE Plugin—Browse for applications 1



Knowledge Center WDE Plugin—Browse for applications 2

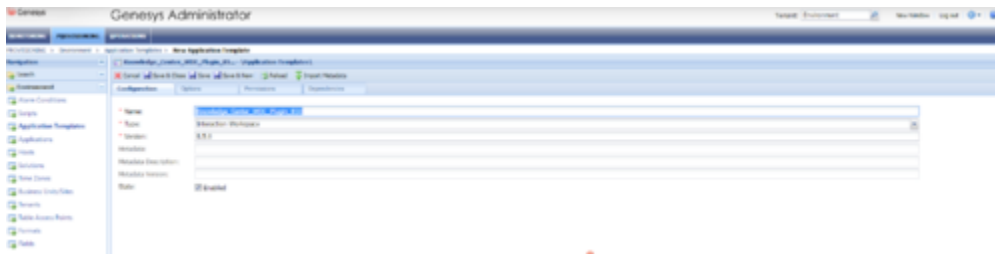
Add Knowledge Center Options to Your WDE Application

To use the Knowledge Center Plugin for WDE, you need to add some options to your WDE application so that it can gather knowledge-related information from incoming interactions. You can add these options to the the **interaction-workspace** section of the WDE application.

Start

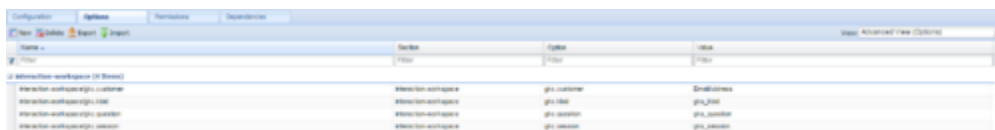
1. Import the template with the additional options:
 1. Open Genesys Administrator and navigate to **Provisioning > Environment > Application Templates**.
 2. In the **Tasks** panel, click **Upload Template**.

3. In the *Click 'Add' and choose application template (APD) file to import* window, click **Add**.
4. Choose the application template (APD) file from the import window and click **Add**.
5. Browse to the *Knowledge_Center_WDE_Plugin_851.apd* file available in the templates directory of your installation CD. The **New Application Template** panel opens.



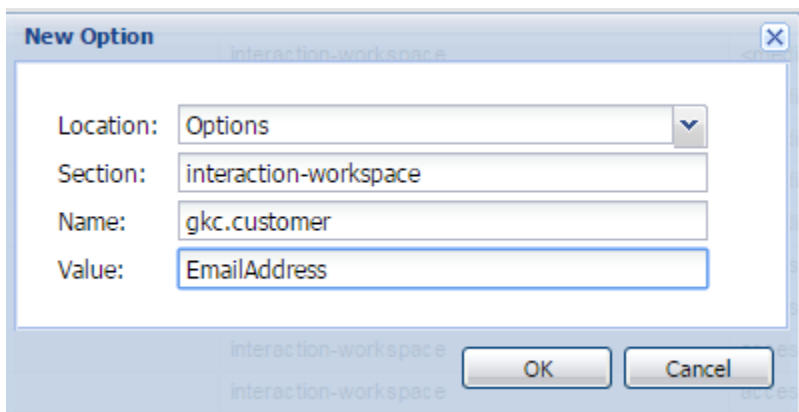
Knowledge Center WDE Plugin—New Application Template panel

6. Click **Save and Close**
2. Open the **Options** tab of the uploaded application and review the new options.



Knowledge Center WDE Plugin—Options tab of uploaded application

3. Navigate to **Provisioning > Environment > Applications**. Select the application defined for Workspace Desktop Edition and click **Edit...**
4. Open the **Options** tab.
5. Add the plugin options to the **interaction-workspace** section using the **New** button.



Knowledge Center WDE Plugin—Add plugin options

End

The Knowledge Center Plugin for WDE uses the following additional options:

- **gkc.question**—This key points to the customer's question for the pre-populated search and is stored in the interaction's user data
- **gkc.kbid**—This key points to the knowledge base ID for the pre-populated search and is stored in the interaction's user data
- **gkc.customer**—This key points to the *customerId* in the interaction's user data (the default value for this key is the customer's email address)
- **gkc.session**—This key stores the session ID in the interaction's user data

Providing Knowledge Center Access to Agents

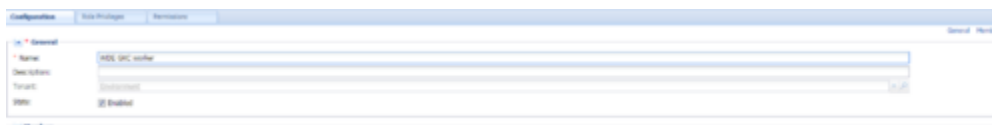
Genesys Knowledge Center supports the following privilege in order to restrict Agent access:

- **Knowledge.WORKER**—Enables access to the Genesys Knowledge Center tab in WDE

To configure the appropriate role for an agent:

Start

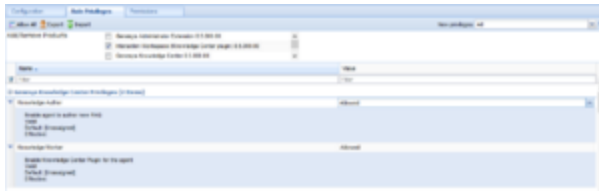
1. Go to **Provisioning > Environment > Application Templates**.
2. Select the application template defined for Workspace Desktop Edition and click **Edit...**
3. Click **Import Metadata**.
4. Click **Add** and select the *Knowledge_Center_WDE_Plugin_851.xml* file.
5. Click **Open**.
6. Information from the metadata file will be added to the template and the appropriate privilege will be added into the framework.
7. Save and Close.
8. Go to **Provisioning > Accounts > Roles**.
9. In the taskbar click **New** to create a new object.
10. Set the name of the role in the **General** section.



Knowledge Center WDE Plugin—Set Role Names

11. Go to the **Role Privileges** tab, and select the set of roles for Genesys Knowledge Center.
12. Open the WDE Knowledge Center Plugin privileges list and select the **Genesys Knowledge Center Privileges** section.
13. Create the appropriate privileges as allowed.

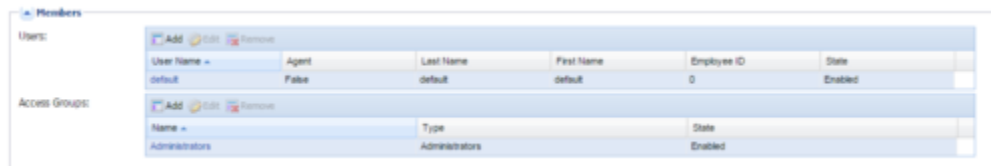
Installing the Workspace Desktop Edition Plugin



Knowledge Center WDE Plugin—Create Privileges

14. Go back to the **Configuration** tab.

15. Add the appropriate Agent to the **Members** section by clicking the **Add** button.



Knowledge Center WDE Plugin—Members Section

16. Save and Close.

End

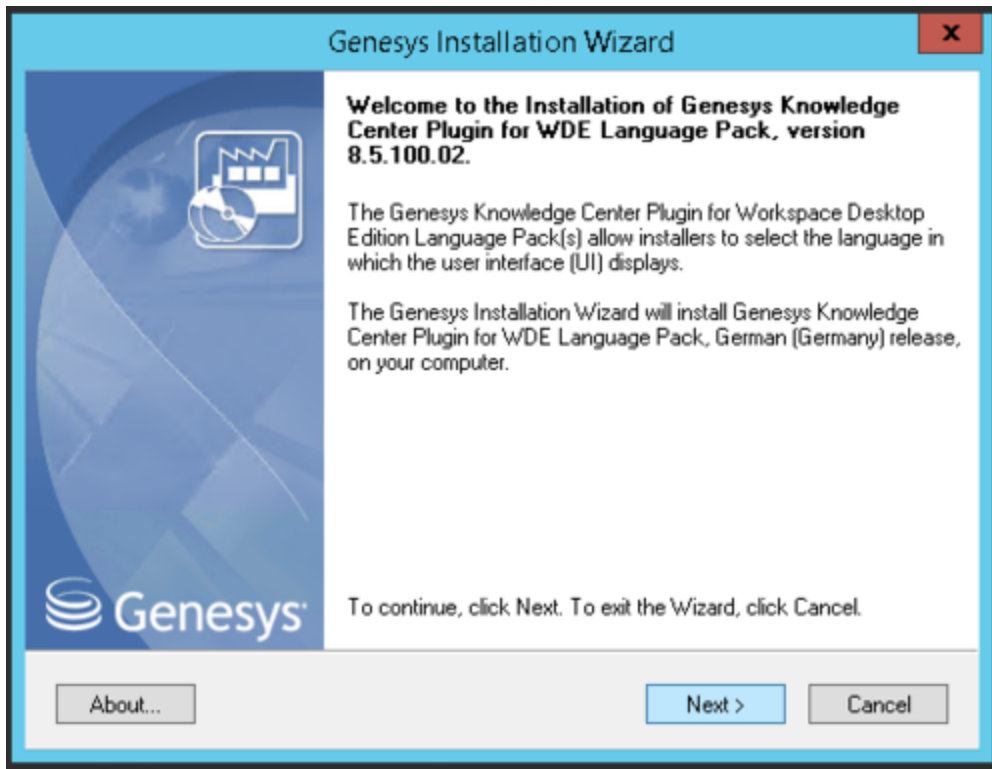
Installing the WDE Language Pack

Prerequisites:

- Must have Genesys Knowledge Center 8.5.1 installed
- Must have Workspace Desktop Edition 8.5.1 installed
- Must have Knowledge Center Workspace Desktop Edition Plugin 8.5.1 installed

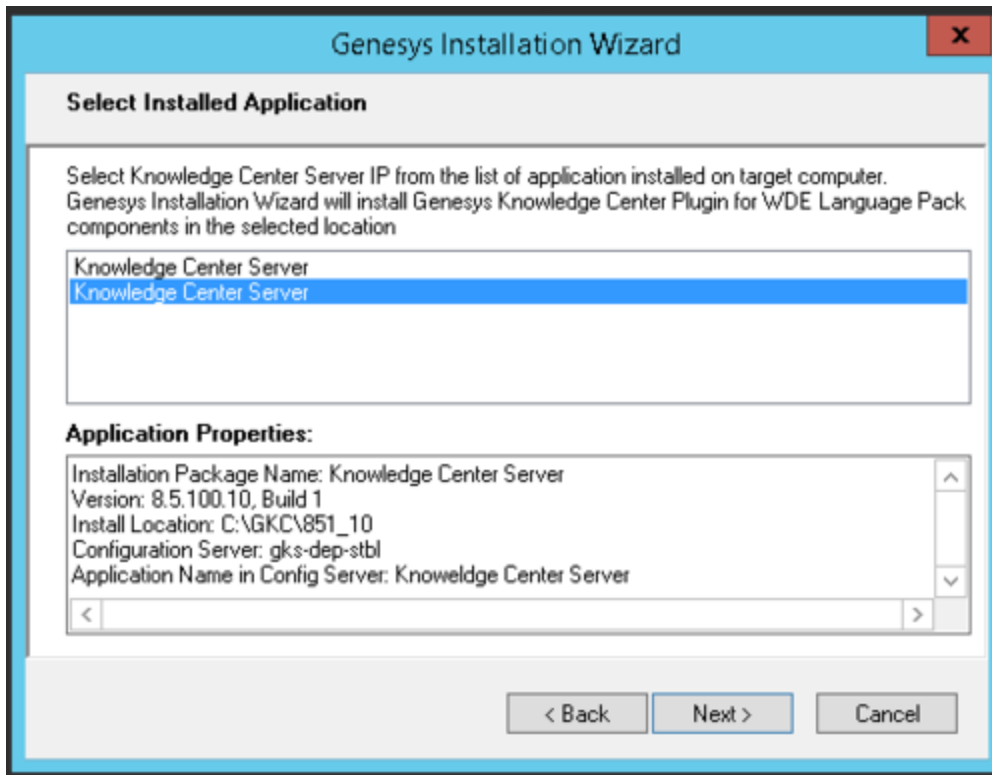
Start

1. In your Language Pack installation package, locate and double-click the **setup.exe** file. The Install Shield opens the welcome screen.



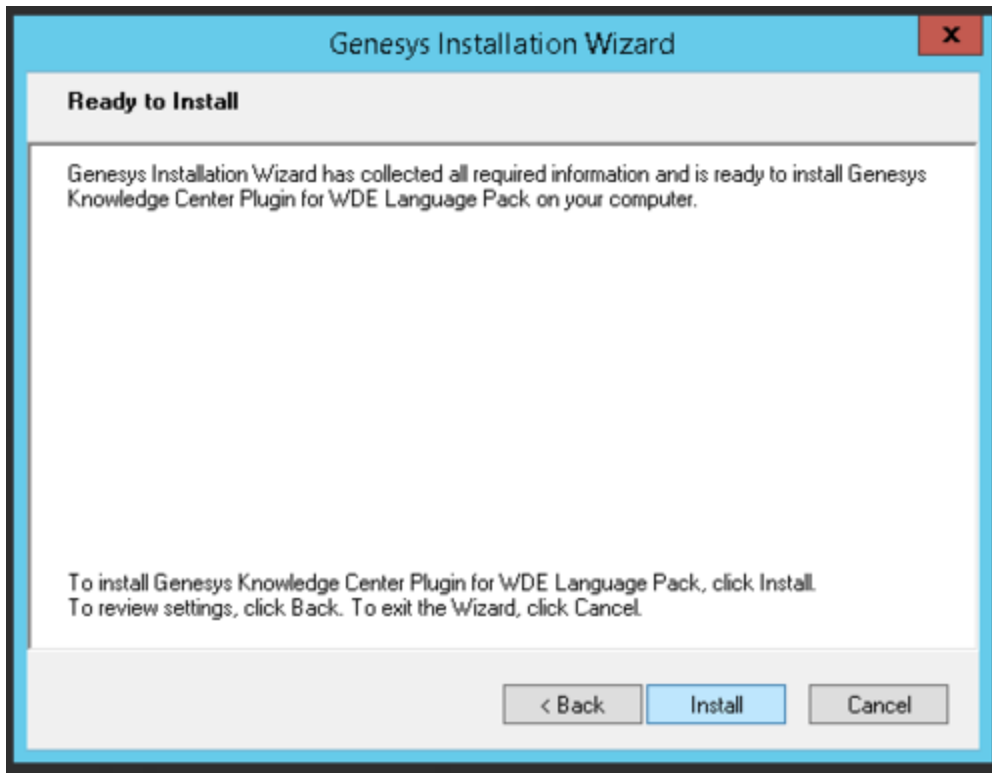
WDE Language Pack Installation Welcome Screen

2. Click **Next**. The **Select Installed Application** screen appears.
3. Select the installed Knowledge Center Server Application for which you want to install the plugin. The **Application Properties** area shows the **Type**, **Host**, **Working Directory**, **Command Line executable**, and **Command Line Arguments** information previously entered in the Server Info and Start Info tabs of the selected Application object.



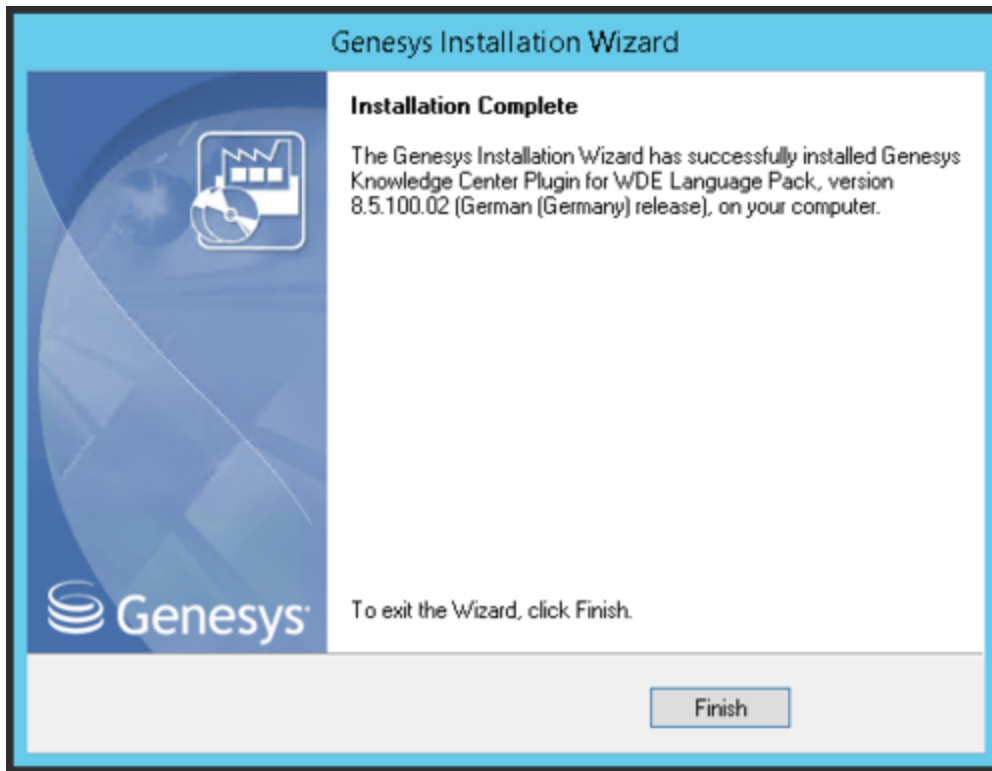
Select Installed Knowledge Center Server Application

4. Click **Next**. The **Ready to Install** screen appears.



WDE Language Pack—Ready to Install

5. Click **Install**. The Genesys Installation Wizard indicates it is performing the requested operation for Backend Server. When through, the **Installation Complete** screen appears.



WDE Language Pack—Installation Complete

6. Click **Finish** to complete your installation.
7. Inspect the directory tree of your system to make sure that the following files, based on the language of your language pack, have been installed in the location that you intended:
 - `<KnowledgeCenterServer_InstallationFolder>\server\resources\wde_de.properties`
 - `<KnowledgeCenterServer_InstallationFolder>\server\resources\wde_fr.properties`
 - `<KnowledgeCenterServer_InstallationFolder>\server\resources\wde_es.properties`
 - `<KnowledgeCenterServer_InstallationFolder>\server\resources\wde_pt.properties`

End

Importing into the Knowledge Center Server

Using Indexer to Import Data

If you are not going to use a CMS you can use the indexer to import data for use with Genesys Knowledge Center.

The indexer is installed during the installation of Knowledge Center Server. It is located inside your Knowledge Center Server installation folder in the `\server\tools\indexer` subdirectory.

Options

POSIX-like options	GNU-like long options	Required	Default	Description
-h	--host	yes	none	Genesys Knowledge Center Server host
-f	--file	no	./	file or directory that contains indexed data for import
-t	--transformer	no		file that contains a *.xsl transformer
-u	--user	yes	none	Authorization message
-a	--authorization	no	none	Credential for Basic Authorization on Knowledge Center Server (if enabled)
-sbt	--subTenantId	no	—	sub-tenant identifier
-l	--loop	no	false	read "-f (--file)" file or folder infinitely

Usage

```
java -jar gks-indexer-${version}.jar
--host "http://%host%:%port%/gks-server"
--file "%path_to_repo%/tools/indexer/xml-datasets"
--transformer "gkc.xsl"
--user "gksuser"
--authorization "user name:user password"
```

XML example

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<documents kbsId="knowledgeFAQ" lang="en">
  <document>
    <lang>en</lang>
    <question>Question from indexer</question>
    <answer>Answer from indexer</answer>
    <categories>
      <category>
        <name>Testing</name>
      </category>
      <category>
        <name>Actors</name>
      </category>
    </categories>
    <validTo>2015-12-31 00:00:00</validTo>
    <alternatives>
      <alternative>Alternative question from indexer</alternative>
    </alternatives>
    <media>
      <media>media indexer</media>
      <media>media indexer</media>
    </media>
    <tags>
      <tag>tag indexer</tag>
      <tag>tag indexer</tag>
    </tags>
    <url>indexer url</url>
    <customFields>
      <entry>
        <key>numfield</key>
        <value>123</value>
      </entry>
      <entry>
        <key>strfield</key>
        <value>Hello GKS</value>
      </entry>
      <entry>
        <key>datefield</key>
        <value>2015-12-31</value>
      </entry>
      <entry>
        <key>unexistingField</key>
        <value>I'm an unexisting custom field</value>
      </entry>
    </customFields>
    <created>2014-10-01 00:00:00</created>
    <modified>2014-12-31 00:00:00</modified>
    <attachments>
      <attachment>
        http://www.the-digital-picture.com/Owners-Manuals/Canon-Speedlite-430ex-Flash-Manual.pdf
      </attachment>
    </attachments>
  </document>
</documents>
```

JSON example

```
{
  "kbsId": "knowledgebaseId",
```

```
"lang": "en",
"documents": [{
  "answer": "answer_1",
  "categories": [{
    "id": "cat_1_id",
    "name": "cat_1"
  }, {
    "id": "cat_2_id",
    "name": "cat_2"
  }],
  "created": "2013-09-08 13:15:33",
  "id": "document_1_id",
  "media": [
    "application",
    "audio"
  ],
  "modified": "2014-01-03 22:03:19",
  "question": "question_1",
  "tags": [
    "tag1",
    "tag2"
  ],
  "url": "google.com"
}, {
  "answer": "answer_2",
  "categories": [{
    "id": "cat_1_id",
    "name": "cat_1"
  }, {
    "id": "cat_3_id",
    "name": "cat_3"
  }],
  "created": "2010-03-09 11:15:21",
  "id": "document_2_id",
  "media": [
    "video",
    "text"
  ],
  "modified": "2013-08-01 05:54:52",
  "question": "question_2",
  "tags": [
    "tag3",
    "tag4"
  ],
  "url": "genesys.com"
}]
}
```

Importing Sample Data

You can use the Import Tool to add sample QNA data to your knowledge base. This tool is located in the `./server/tools` directory in the Knowledge Center installation folder. It comes with the following resources:

- **knowledgeFAQ.xml**—List of basic QNA data, provided with the Knowledge Center Server indexing tool

- **gks-indexer-tool.jar**—Java-based indexing tool
- **importFAQ.bat**—Simple data import script

Data Import Syntax

Important

Users must have **Knowledge.AUTHOR** privileges in order to use the Administrator plugin.

Use the following syntax to import data:

```
- java -jar gks-indexer-${version}.jar
--host "http://%host%:%port%/gks-server"
--file "%path_to_repo%/tools/indexer/xml-datasets"
--transformer "gkc.xml"
--user "gksuser"
--authorization "user name:user password"
```

The authorization parameter is only required if you have enabled the security option for Knowledge Center Cluster.

Sample Import Script

Here is an example of what your import script might look like:

```
java -jar <Path to GKC Server>\GKC_Server\server\tools\gks-indexer-tool.jar
--host "http://sample.com:9001/gks-server"
--file "<Path to GKC Server>\GKC_Server\server\tools\knowledgeFAQ.xml"
--user "gkc_admin"
```

If it works, this script will import sample QNA data into the knowledge base.

Sample QNA Data

Here is an example of the data stored in the XML file:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<documents kbsId="knowledgeFAQ" lang="en">
  <document>
    <question>What Is Knowledge Center?</question>
    <answer>The Genesys Knowledge Center ultimate goal is to convert
your knowledge into the answers on the question your clients or agents have.
It delivers set of the component for administration, authoring and using the
knowledge. The heart of the system is the Knowledge Center Server that
aimed to find the best answer on the question you have asked.</answer>
    <categories>
      <category>
        <name>General</name>
      </category>
    </categories>
  </document>
```

</documents>

Security

Genesys Knowledge Center supports HTTPS/SSL/TLS to protect data over the web.

- All connections can be secured, including connections from the browser to the Knowledge Center servers.
- Applications defined in Configuration Server can have both HTTP and HTTPS connections.

Transport Layer Security (TLS) is supported above Java containers, Jetty and Apache Tomcat. The user data submitted from the browser tier is always sent through secure connections. To support secure (TLS) connections to Configuration Server on Windows OS, if you use the JDK 6 64 bit, it is mandatory to do *one* of the following:

- Update Java Development Kit 6 64 bit with Java SE Development Kit 6, Update 38 (JDK 6u38) or older.
- Set up JDK 7.

Important

Genesys performs security testing with **OWASP Zed Attack Proxy** (ZAPProxy) to make sure the Genesys Knowledge Center solution is invincible to known attacks.

Genesys Knowledge Center includes additional security configurations that can be used with your Knowledge Center installation:

- **Secure Sockets Layer (SSL)** — Load SSL certificates and configure Jetty.
- **Transport Layer Security (TLS)** — Configure TLS for Genesys and Knowledge Center servers.
- **Authentication** — Enable authentication for the Knowledge Center Server and the CMS.

SSL Configuration for Knowledge Center Servers

The Jetty web server supplied with the Genesys Knowledge Center Server and CMS includes a pre-configured, self-signed certificate. This allows you to use HTTPS out of the box in a sandbox deployment. In common case, you should use a certificate issued by a third-party Certificate Authority. The procedures on this page provide examples of ways to load SSL certificates and configure Jetty. These examples may vary depending on your environment.

Important

You must use the Java Development Kit version 1.6.0_29 or higher to support the JSSE keystore.

Loading an SSL Certificate and Private Key into a JSSE Keystore

Important

In a development environment, you can use self-signed certificates, but in a production environment you should use a certificate issued by a third-party Certificate Authority, such as VeriSign.

Prerequisites

- An SSL certificate, either generated by you or issued by a third-party Certificate Authority. For more information on generating a certificate, see http://wiki.eclipse.org/Jetty/Howto/Configure_SSL.

Start

1. Depending on your certificate format, do **one** of the following:
 - If your certificate is in PEM form, you can load it to a JSSE keystore with the keytool using the following command:

```
keytool -keystore keystore -importcert -alias alias -file certificate_file -trustcacerts
```

Where:

keystore is the name of your JSSE keystore.

alias is the unique alias for your certificate in the JSSE keystore.

certificate_file is the name of your certificate file. For example, *jetty.crt*.

- If your certificate and key are in separate files, you must combine them into a PKCS12 file before loading it to a keystore.

1. Use the following command in openssl to combine the files:
`openssl pkcs12 -inkey private_key -in certificate -export -out pkcs12_file`

Where:

private_key is the name of your private key file. For example, *jetty.key*.

certificate is the name of your certificate file. For example, *jetty.crt*.

pkcs12_file is the name of the PKCS12 file that will be created. For example, *jetty.pkcs12*.

2. Load the PKCS12 file into a JSSE keystore using keytool with the following command:
`keytool -importkeystore -srckeystore pkcs12_file -srcstoretype store_type -destkeystore keystore`

Where:

pkcs12_file is the name of your PKCS12 file. For example, *jetty.pkcs12*.

store_type is the file type you are importing into the keystore. In this case, the type is PKCS12.

keystore is the name of your JSSE keystore.

Important

You will need to set two passwords during this process: keystore and truststore. Make note of these passwords because you will need to add them to your Jetty SSL configuration file.

End

Configuring Jetty

Start

1. Open the Jetty SSL configuration file in a text editor: ***jetty_installation/etc/jetty-ssl.xml***.
2. Find the **<New id="sslContextFactory" class="org.eclipse.jetty.http.ssl.SslContextFactory">** element and update the passwords:

```
<Configure id="sslContextFactory" class="org.eclipse.jetty.util.ssl.SslContextFactory">
  <Set name="KeyStorePath"><path to keystore><Property name="jetty.base"
  default="." /></Property name="jetty.keystore" default="etc/keystore"/></Set>
  <Set name="KeyStorePassword">0BF:<obfuscated_keystore_password><Property
  name="jetty.keystore.password"
  default="0BF:1vny1z1o1x8e1vnw1vn61x8g1zlu1vn4"/></Set>
```



```

    <Set name="KeyManagerPassword">OBF:<obfuscated_keymanager_password><Property
name="jetty.keymanager.password"
default="OBF:1u2u1wml1z7s1z7a1wnl1u2g"/></Set>
    <Set name="TrustStorePath">"><path to truststore><Property name="jetty.base"
default="." /></Property name="jetty.truststore"
default="etc/keystore"/></Set>
    <Set name="TrustStorePassword"> OBF:<obfuscated_truststore_password><Property
name="jetty.truststore.password"
default="OBF:1vn1z1o1x8e1vnw1vn61x8g1zlu1vn4"/></Set>
    <Set name="EndpointIdentificationAlgorithm"></Set>
    <Set name="NeedClientAuth"><Property name="jetty.ssl.needClientAuth"
default="false"/></Set>
    <Set name="WantClientAuth"><Property name="jetty.ssl.wantClientAuth"
default="false"/></Set>
    <Set name="ExcludeCipherSuites">
        <Array type="String">
            <Item>SSL_RSA_WITH_DES_CBC_SHA</Item>
            <Item>SSL_DHE_RSA_WITH_DES_CBC_SHA</Item>
            <Item>SSL_DHE_DSS_WITH_DES_CBC_SHA</Item>
            <Item>SSL_RSA_EXPORT_WITH_RC4_40_MD5</Item>
            <Item>SSL_RSA_EXPORT_WITH_DES40_CBC_SHA</Item>
            <Item>SSL_DHE_RSA_EXPORT_WITH_DES40_CBC_SHA</Item>
            <Item>SSL_DHE_DSS_EXPORT_WITH_DES40_CBC_SHA</Item>
        </Array>
    </Set>

```

Note: You can run Jetty's password utility to obfuscate your passwords. See http://wiki.eclipse.org/Jetty/Howto/Secure_Passwords.

3. Save your changes.

End

Choosing a Directory for the Keystore

The keystore file in the example above is given relative to the Jetty home directory. For production, you should keep your keystore in a private directory with restricted access. Even though the keystore has a password, the password may be configured into the runtime environment and is vulnerable to theft.

You can now start Jetty the normal way (make sure that **jcrt.jar**, **jnet.jar** and **jsse.jar** are on your classpath) and SSL can be used with a URL, such as `https://your_IP:8743/`

Transport Layer Security (TLS)

Genesys Knowledge Center supports the Transport Layer Security (TLS) protocol to secure data exchanged with other Genesys components. For details about TLS, see the [Genesys 8.1 Security Deployment Guide](#). You can configure TLS for Knowledge Center by completing the procedures on this page.

Configuring TLS for Genesys Servers

To configure the TLS parameters for Genesys servers, see [Introduction to Genesys Transport Layer Security](#).

Configuring TLS for Genesys Knowledge Center Server

To enable TLS support for the Genesys Knowledge Center Server, you must do the following:

1. Have properly installed a trusted certificates for the Genesys server. For more information, please see [Certificate Generation and Installation](#).
2. Configure TLS options for the Genesys Knowledge Center Server application.
3. Configure the appropriate connections between the Genesys Knowledge Center Server application and the necessary Genesys servers through secure ports. For example, by setting a secure Config Server port in the *Server Installation Folder/server/setenv.bat* file in the **PRIMARY_CFGSERVER_PORT** variable.

Configuring Secure Connections to Configuration Server

To configure a secured connection from Genesys Knowledge Center Server to Configuration Server use the following TLS-related configuration options in the **setenv.bat/sh** configuration:

Parameter Name	Acceptable Values	Purpose
PRIMARY_CFGSERVER_CONNECTION_MODE	TLS, UPGRADE or UNSECURED by default	Set this option to enable secured connection Important Incorrect setting of this parameter can lead to inability to establish a connection with the server
PROVIDER	PEM, JKS, MSCAPI, PKCS11	Type of used security provider
TRUSTED_CA	valid file name (including path)	Path to trusted CA PEM file or JKS truststore file or SHA-1 Thumbprint for MSCAPI storage.

Parameter Name	Acceptable Values	Purpose
		Specifies the name of the trusted store file which holds the public certificate to verify the server. Applicable for PEM and JKS trusted storage types only.
TRUSTSTORE_PASSWORD	n/a	Password for the JKS trusted storage. Provide password only if trusted CA is in the JKS format.
In case of enabled mutual TLS, configure the following options:		
CERTIFICATE	n/a	Client certificate file in PEM format or JKS keystore file or SHA-1 Thumbprint for MSCAPI storage.
PRIVATE_KEY	n/a	Unencrypted private key in PEM format or Certificate SHA-1 Thumbprint for MSCAPI storage. Ignored for JKS storage.
KEYSTORE_PASSWORD	n/a	Provide password if key storage is in the JKS format.
KEYENTRY_PASSWORD	n/a	Provide password if private key encrypted by its own password.

Configuring TLS Options

The Genesys Knowledge Center Server includes the following TLS-related configuration options in its **security** section.

Parameter Name	Acceptable Values	Purpose
tls	Boolean value. Possible values are "1"/"0", "yes"/"no", "on"/"off", "true"/"false". Example: <ul style="list-style-type: none">"tls=1"	Client: 1 - perform TLS handshake immediately after connecting to server. 0 - do not turn on TLS immediately but autodetect can still work.
provider	"PEM", "MSCAPI", "PKCS11" Not case-sensitive. Example: <ul style="list-style-type: none">"provider=MSCAPI"	Explicit selection of security provider to be used. For example, MSCAPI and PKCS11 providers can contain all other parameters in their internal database. This parameter allow configuration of TLS through security provider tools.
certificate	PEM provider: path to a X.509 certificate file in PEM format. Path can use both forward and backward slash characters.	Specifies location of X.509 certificate to be used by application. MSCAPI provider keeps certificates in internal database and can identify them

Parameter Name	Acceptable Values	Purpose
	<p>MSCAPI provider: thumbprint of a certificate – string with hexadecimal SHA-1 hash code of the certificate. Whitespace characters are allowed anywhere within the string. PKCS11 provider: this parameter is ignored.</p> <p>Examples:</p> <ul style="list-style-type: none"> "certificate= C:\certs\client-cert-3-cert.pem" "certificate=A4 7E A6 E4 7D 45 6A A6 2F 15 BE 89 FD 46 F0 EE 82 1A 58 B9" 	<p>by hash code; so called thumbprint.</p> <p>In Java, PKCS#11 provider does not allow selection of the certificate; it must be configured using provider tools.</p> <p>Note: When using autodetect (upgrade) TLS connection, this option MUST be specified in application configuration, otherwise Configuration Server would return empty TLS parameters even if other options are set.</p>
certificate-key	<p>PEM provider: path to a PKCS#8 private key file without password protection in PEM format. Path can use both forward and backward slash characters.</p> <ul style="list-style-type: none"> MSCAPI provider: this parameter is ignored; key is taken from the entry identified by "certificate" field. PKCS11 provider: this parameter is ignored. <p>Examples:</p> <ul style="list-style-type: none"> "certificate-key= C:\certs\client-cert-3-key.pem" 	<p>Specifies location of PKCS#8 private key to be used in pair with the certificate by application.</p> <p>MSCAPI provider keeps private keys paired with certificates in internal database. In Java, PKCS#11 provider does not allow selection of the private key; it must be configured using provider tools.</p>
trusted-ca	<p>PEM provider: path to a X.509 certificate file in PEM format. Path can use both forward and backward slash characters.</p> <p>MSCAPI provider: thumbprint of a certificate – string with hexadecimal SHA-1 hash code of the certificate. Whitespace characters are allowed anywhere within the string. PKCS11 provider: this parameter is ignored.</p> <p>Examples:</p> <ul style="list-style-type: none"> "trusted-ca= C:\certs\ca.pem" "trusted-ca=A4 7E A6 E4 7D 45 6A A6 2F 15 BE 89 FD 46 F0 EE 82 1A 58 B9" 	<p>Specifies location of a X.509 certificate to be used by application to validate remote party certificates. The certificate is designated as Trusted Certification Authority certificate and application will only trust remote party certificates signed with the CA certificate.</p> <p>MSCAPI provider keeps CA certificates in internal database and can identify them by hash code; so called thumbprint. In Java, PKCS#11 provider does not allow selection of the CA certificate; it must be configured using provider tools.</p>
tls-mutual	Boolean value.	Has meaning only for server application. Client applications

Parameter Name	Acceptable Values	Purpose
	<p>Possible values are "1"/"0", "yes"/"no", "on"/"off", "true"/"false".</p> <p>Example:</p> <ul style="list-style-type: none"> "tls-mutual=1" 	ignore this value. When turned on, server will require connecting clients to present their certificates and validate the certificates the same way as client applications do.
tls-crl	<p>All providers: path to a Certificate Revocation List file in PEM format. Path can use both forward and backward slash characters.</p> <p>Example:</p> <ul style="list-style-type: none"> "tls-crl= C:\certs\crl.pem" 	Applications will use CRL during certificate validation process to check if the (seemingly valid) certificate was revoked by CA. This option is useful to stop usage of leaked certificates by unauthorized parties.
tls-target-name-check	<p>"host" or none. Not case-sensitive.</p> <p>Example:</p> <ul style="list-style-type: none"> "tls-target-name-check=host" 	When set to "host", enables matching of certificate's Alternative Subject Name or Subject fields against expected host name. PSDK supports DNS names and IP addresses as expected host names.
cipher-list	<p>String consisting of space-separated cipher suit names. Information on cipher names can be found online.</p> <p>Example:</p> <ul style="list-style-type: none"> "cipher-list= TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA TLS_ECDH_RSA_WITH_3DES_EDE_CBC_SHA" 	Used to calculate enabled cipher suites. Only ciphers present in both the cipher suites supported by security provider and the cipher-list parameter will be
fips140-enabled	<p>Boolean value.</p> <p>Possible values are "1"/"0", "yes"/"no", "on"/"off", "true"/"false".</p> <p>Example:</p> <ul style="list-style-type: none"> "fips140-enabled=1" 	PSDK Java: when set to true, effectively is the same as setting "provider=PKCS11" since only PKCS11 provider can support FIPS-140. If set to true while using other provider type, PSDK will throw exception.
sec-protocol	<p>String value.</p> <p>Possible values are "SSLv23", "SSLv3", "TLSv1", "TLSv1.1", "TLSv1.2".</p> <p>Example:</p> <ul style="list-style-type: none"> "sec-protocol=TLSv1" 	Starting with PSDK release 8.5.1, an application can specify the exact protocol to send and accept secure connection requests on one or more of its connections.

See [Configuring Trusted Stores](#) below for details about configuration for a specific type of store (PEM, JKS, MSCAPI).

Configuring Trusted Stores

PEM Trusted Store

PEM stands for "Privacy Enhanced Mail", a 1993 IETF proposal for securing e-mail using public-key cryptography. That proposal defined the PEM file format for certificates as one containing a Base64-encoded X.509 certificate in specific binary representation with additional metadata headers.

PEM certificate trusted store works with CA certificate from an X.509 PEM file. It is a recommended trusted store to work on Linux systems.

Complete the steps below to work with the PEM certificate trusted store:

Start

1. Configure TLS for Genesys servers to use certificates signed by CA certificate **certificateCA.crt**.
2. Place the trusted CA certificate in PEM format on the Genesys Knowledge Center Server application host. To convert a certificate of another format to .pem format you can use the [OpenSSL tool](#). For example:
 - Convert a DER file (.crt .cer .der) to PEM:
`openssl x509 -inform der -in certificateCA.crt -out certificateCA.pem`
 - Convert a PKCS#12 file (.pfx .p12) containing a private key and certificates to PEM:
`openssl pkcs12 -in certificateCA.pfx -out certificateCA.pem -nodes`

You can add **-nocerts** to only output the private key or add **-nokeys** to only output the certificates.
3. In Genesys Administrator, navigate to **Provisioning > Environment > Applications** and open your Knowledge Center Server application.
4. Click the **Options** tab and navigate to the [security](#) section.
5. Set the **trusted-ca-type** option to PEM.
6. Set the **trusted-ca** option to the path and file name for your trusted CA in PEM format on the Genesys Knowledge Center Server application host.
7. Click **Save & Close**.

End

JKS Trusted Store

A Java KeyStore (JKS) is a repository of security certificates used, for instance, in SSL/TLS encryption. The Java Development Kit provides a tool named [keytool](#) to manipulate the keystore.

Complete the steps below to work with the JKS certificate trusted store:

Start

1. Configure TLS for Genesys servers to use certificates signed by CA certificate **certificateCA.crt**.
2. Import the CA certificate to an existing Java keystore using keytool:

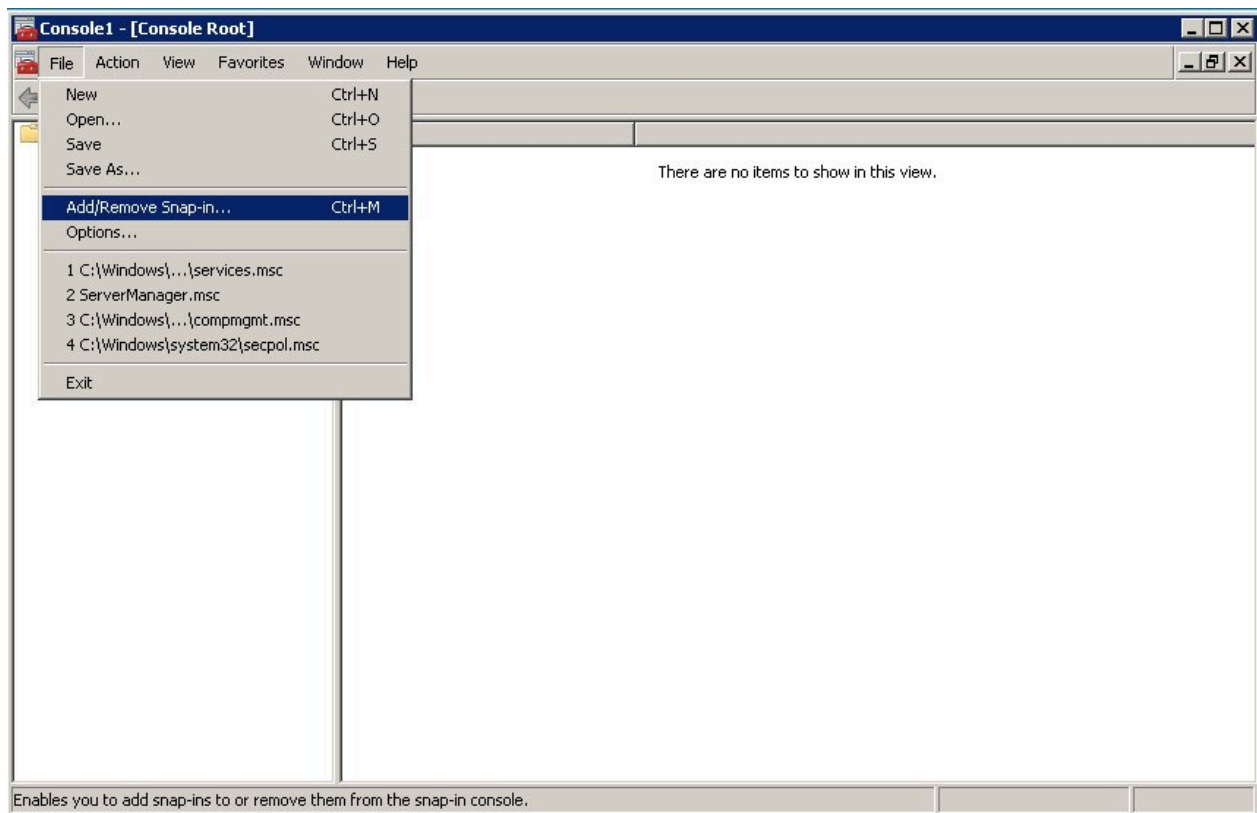
-
- Run the keytool command with option -alias set to root:
`keytool -import -trustcacerts -alias root -file certificateCa.crt -keystore /path/to/keystore/keystore.jks`
 - Enter the keystore password in command line prompt - for example:
Enter keystore password: somepassword
3. In Genesys Administrator, navigate to **Provisioning > Environment > Applications** and open your Knowledge Center Server application.
 4. Click the **Options** tab and navigate to the **security** section.
 5. Set the **trusted-ca-type** option to JKS.
 6. Set the **trusted-ca** option to the path and file name for your JKS trusted storage type on the Genesys Knowledge Center Server application host.
 7. Set the **trusted-pwd** option to the password defined for your keystore in Step 2.
 8. Click **Save & Close**.

End**MSCAPI Trusted Store**

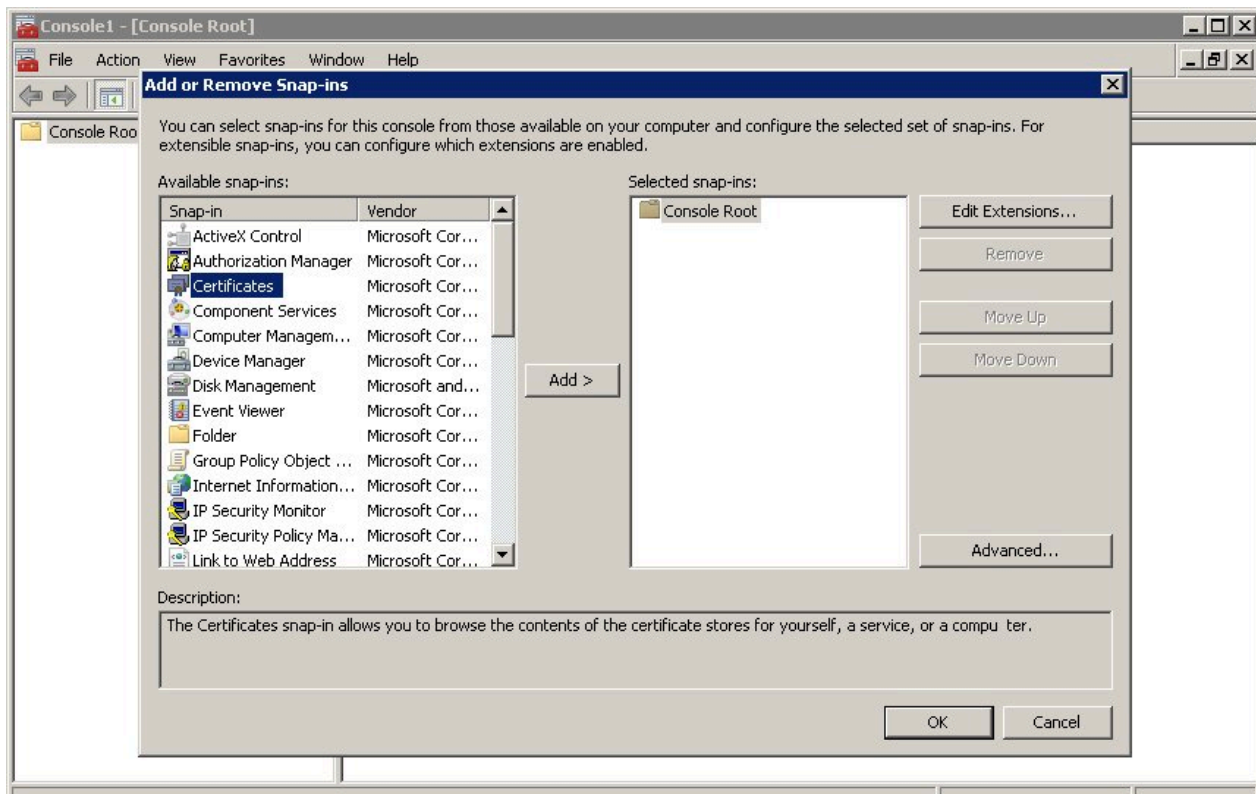
Complete the steps below to work with the MSCAPI certificate trusted store:

Start

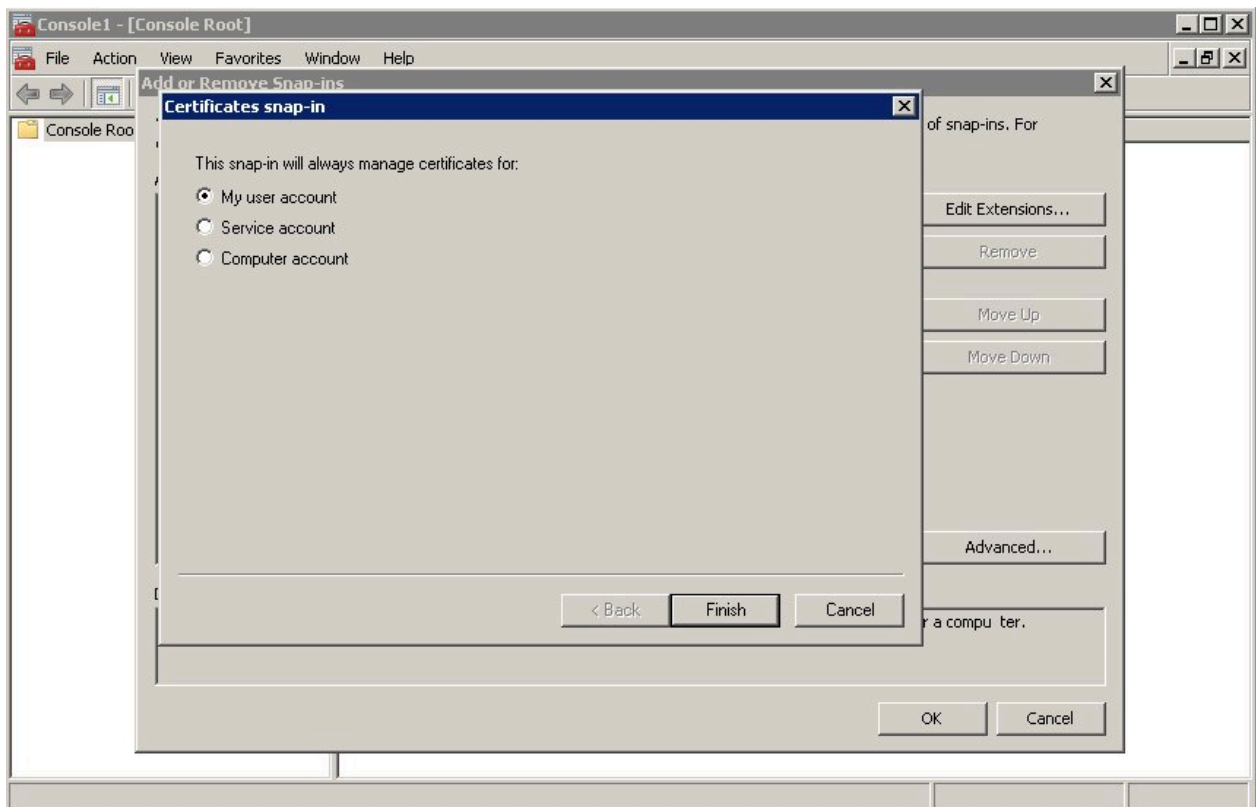
1. Configure and tune TLS for Genesys servers to use certificates signed by the same CA.
2. If the Knowledge Center Server is running on a different host, copy the trusted CA certificate to this host.
3. Import the CA certificate to WCS via Certificates Snap-in on the Knowledge Center Server host by launching the MMC console. Enter mmc at the command line.
4. Select **File > Add/Remove Snap-in...** from the main menu.



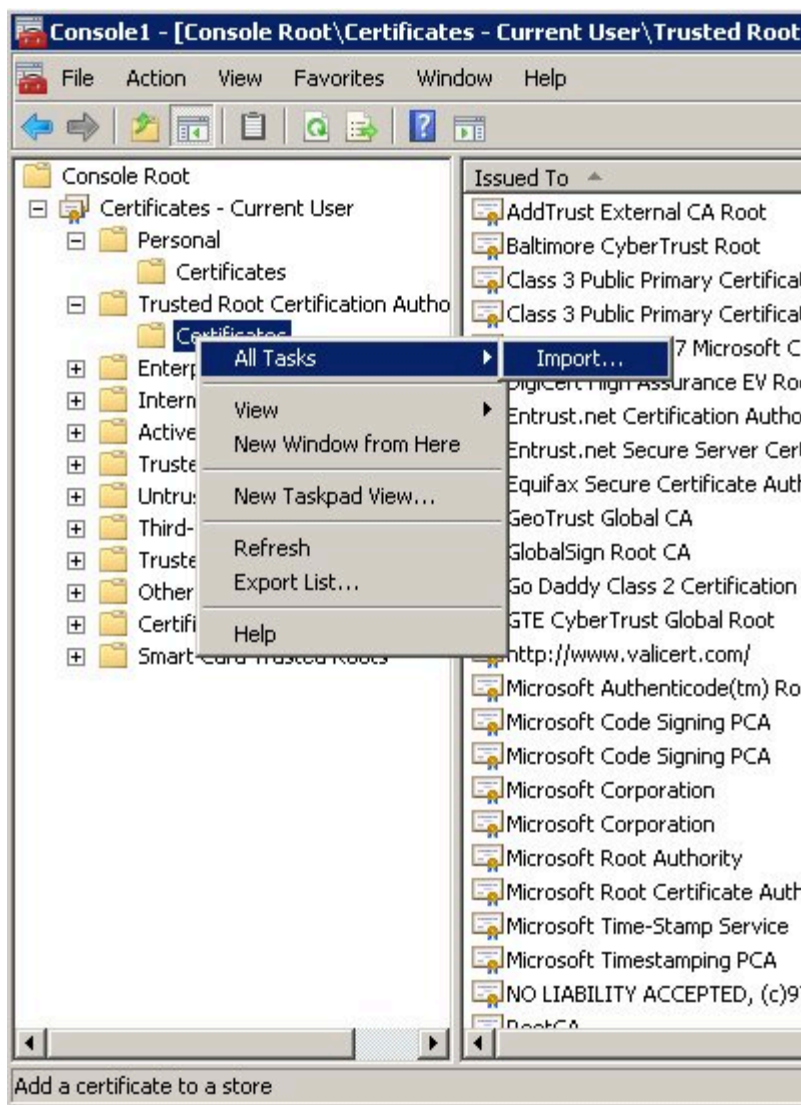
5. Select **Certificates** from the list of available snap-ins and click **Add**.



6. Select the account to manage certificates for and click **Finish**. It is important to place certificates under the correct Windows account. Some applications are run as services under the Local Service or System account, while others are run under user accounts. The account chosen in MMC must be the same as the account used by the application which certificates are configured for, otherwise the application will not be able to access this WCS storage.



7. Click **OK**.
8. Import a certificate. Right-click the "Trusted Root Certification Authorities/Certificates" folder and choose All Tasks > Import... from the context menu. Follow the steps presented by the Certificate Import Wizard. Once finished the imported certificate appears in the certificates list.



9. In Genesys Administrator, navigate to **Provisioning > Environment > Applications** and open your Knowledge Center Server application.
10. Click the **Options** tab and navigate to the **security** section.
11. Set the **trusted-ca-type** option to MSCAP.
12. Click **Save & Close**.

End

Configuring TLS for a server running Windows

By default, Genesys Knowledge Server as a Windows Service runs without a TLS connection. To configure a secure connection from Genesys Knowledge Center Server to a Configuration Server while running as a Windows Service you need to update the installed default service for Genesys

Knowledge Center Server.

In order to do this you will need to:

1. Remove Genesys Knowledge Center Server Windows Service, which was configured in the installation package.
 1. Run the Windows Command Prompt (cmd.exe)
 2. Go to *<Knowledge Center Server installation folder>/server*
 3. Run the next command: **server.bat** remove
2. Configure a secure connection settings in **setenv.bat** to Genesys Configuration Server, as described in [Configuring Secure Connections to Configuration Server](#).
3. Re-install the Windows Service for Genesys Knowledge Center Server, now with the secure connection configured to Genesys Configuration Server.
 1. Run Windows Command Prompt (cmd.exe)
 2. Go to *<Knowledge Center Server installation folder>/server*
 3. Run the next command: **server.bat** install

Authentication

You can enable secure communications with the **Management** and **Reporting** REST APIs by completing the procedures below to implement authentication. If you do enable authentication, then all API clients must use the authentication scheme and credentials. Three common clients of the API are the Genesys Knowledge Center Plugin for Administrator, Genesys Knowledge Center Plugin for Workspace Desktop Edition and Genesys Knowledge Center CMS.

Configuring Authentication in Genesys Knowledge Center

Complete the steps below to enable authentication for the Management and Reporting REST APIs.

Start

1. In Genesys Administrator, navigate to **Provisioning > Environment > Applications**, select the Knowledge Center Cluster application, and click **Edit...**
2. Click the **Options** tab and scroll down to the **[security]** section.
3. Set the following options:
 - **auth-scheme**
 - **user-id**
 - **password**
4. Click **Save & Close**.

End

UTF8

You can configure your Knowledge Center Servers and Knowledge Center CMS to support UTF-8 in Configuration Server, which in turn supports multi-language categories.

Configuring a UTF-8 Connection to Configuration Server

Complete the following steps for your Knowledge Center Servers.

Prerequisites

- Your version of Configuration Server supports UTF-8. For details, see the [compliant versions](#) for mandatory components.

Start

1. Navigate to the installation directory for your Knowledge Center Server and open the **setenv.bat** file for Windows — or the **setenv.sh** file for Linux — with a text editor. For example: *Path to installation directory/server/setenv.bat*.
2. Find the following string: **:: set JAVA_OPTS=%JAVA_OPTS% -Dgenesys.cfgServerUseUtf8=true**.
3. Remove the two colons (::) at the start. This converts the string from a comment to a command to use UTF-8. Your string should now look like this: **set JAVA_OPTS=%JAVA_OPTS% -Dgenesys.cfgServerUseUtf8=true**
4. Save your changes.

End

Sample UI

Overview

Knowledge Center comes with a Sample UI, hosted on a sample website, which provides basic access to your installation of Knowledge Center and your configured knowledge base content. You can use it to test and demonstrate what Knowledge Center can do or as an example of how to integrate Knowledge Center access into your existing website.

The Sample UI is based on independent and easily configurable components. Its website was created using Bootstrap and works on all web browsers that support Bootstrap. See the [Bootstrap documentation](#) for details.

After you install your Knowledge Center Servers and configure the Knowledge Center Cluster, you can access the Sample UI sandbox via the following URLs:

- If you have configured a load-balancer for your cluster: http://host_load_balancer:port_load_balancer/gks-sample-ui
- If you use a Knowledge Center Cluster with a single node: http://gkc_server_host:gkc_server_port/gks-sample-ui

The Sample UI is pre-configured to use a knowledge base with an ID of **knowledgeFAQ**.

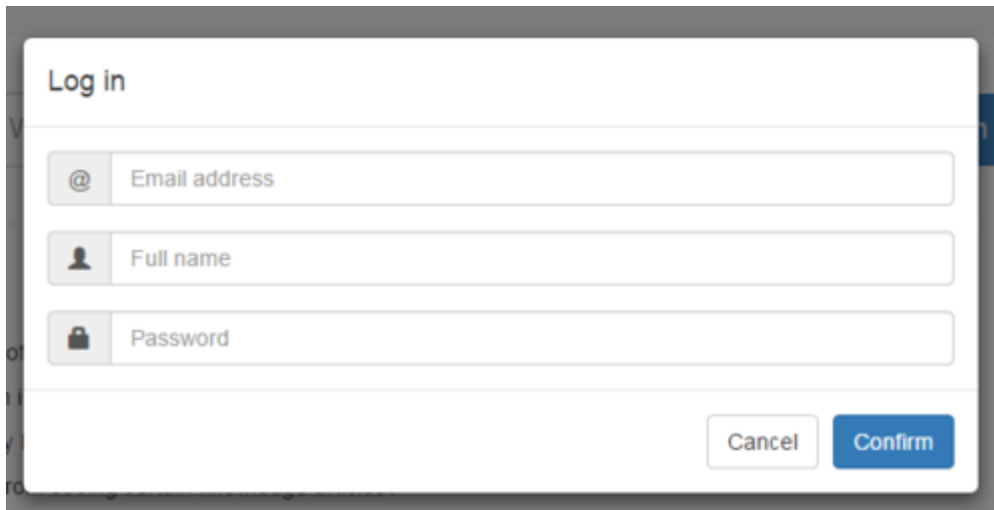
Authorizing

You can use the Sample UI to:

- Browse the site, either as an anonymous user or by authorizing yourself as a customer. To authorize, click the **Log in** link, enter your credentials, and click **Confirm**

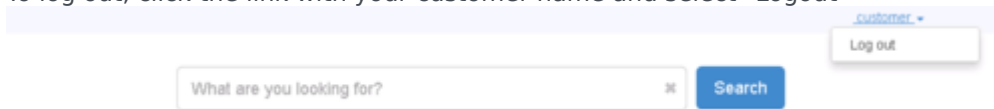
Important

This is not a real site authorization, as Knowledge Center server will only use an email as a *customerId* to identify sessions in History records.

A login form titled "Log in" with three input fields: "Email address" (with an @ icon), "Full name" (with a person icon), and "Password" (with a lock icon). At the bottom right are "Cancel" and "Confirm" buttons.

Sample UI Login

- To log out, click the link with your customer name and select "Logout"

A search bar with the placeholder text "What are you looking for?" and a "Search" button. To the right, a dropdown menu shows "customer" with a downward arrow, and a "Log out" option is visible below it.

Sample UI Logout

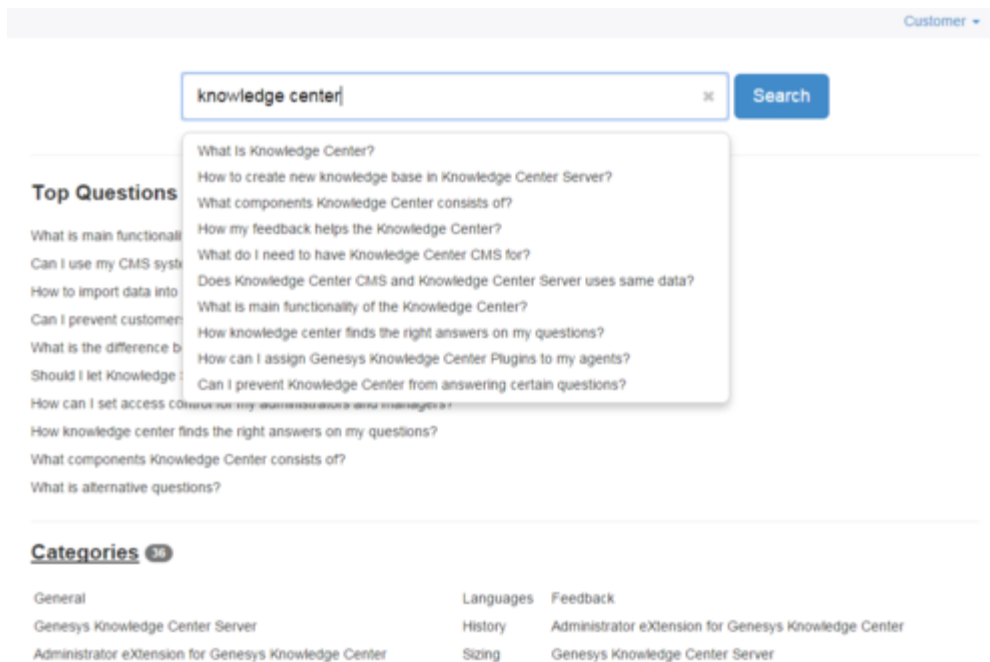
Searching

Search for any QNA document using the search bar.

Conduct a search

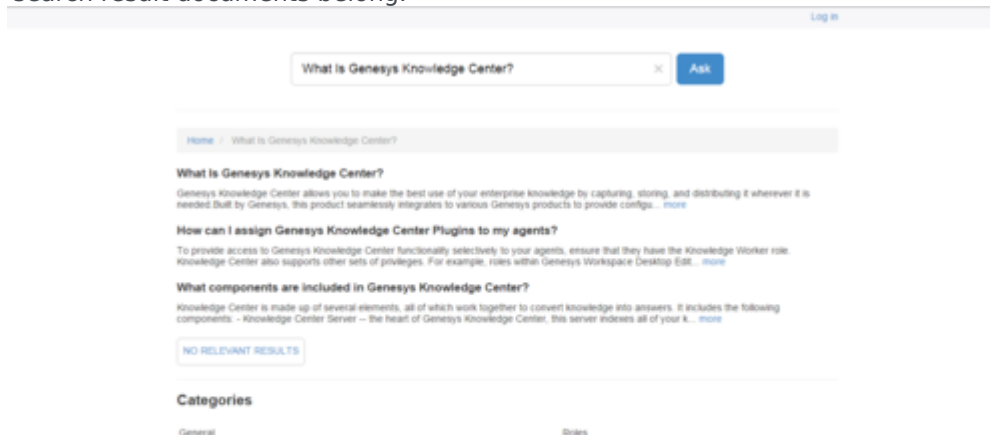
Start

1. Enter a question in the search bar and **Search** or press **Enter**.



Sample UI Search

- Review search results. You can use the **No relevant result** button to let Knowledge Center know that your search was unsuccessful. At the bottom of the page, there is a list of categories to which your search result documents belong.



Sample UI Search Results

End

Open and Review a Document

Important

Documents can be in plain text or rich text

[Log in](#)

[Home](#) / [What is Knowledge?](#)

What is Knowledge?

Theories of knowledge

See also: [Epistemology](#)

“The eventual demarcation of philosophy from science was made possible by the notion that philosophy's core was “theory of knowledge,” a theory distinct from the sciences because it was their foundation... Without this idea of a “theory of knowledge,” it is hard to imagine what “philosophy” could have been in the age of modern science. — Richard Rorty, *Philosophy and the Mirror of Nature*

The definition of knowledge is a matter of ongoing debate among philosophers in the field of epistemology. The classical definition, described but not ultimately endorsed by Plato, specifies that a statement must meet three criteria in order to be considered knowledge: it must be justified, true, and believed. Some claim that these conditions are not sufficient, as Gettier case examples allegedly demonstrate. There are a number of alternatives proposed, including Robert Nozick's arguments for a requirement that knowledge “tracks the truth” and Simon Blackburn's additional requirement that we do not want to say that those who meet any of these conditions “through a defect, flaw, or failure” have knowledge. Richard Kirkham suggests that our definition of knowledge requires that the evidence for the belief necessitates its truth.

In contrast to this approach, Ludwig Wittgenstein observed, following Moore's paradox, that one can say “He believes it, but it isn't so,” but not “He knows it, but it isn't so.” He goes on to argue that these do not correspond to distinct mental states, but rather to distinct ways of talking about conviction. What is different here is not the mental state of the speaker, but the activity in which they are engaged.

Scientific knowledge

The development of the scientific method has made a significant contribution to how knowledge of the physical world and its phenomena is acquired. To be termed scientific, a method of inquiry must be based on gathering observable and measurable evidence subject to specific principles of reasoning and experimentation. The scientific method consists of the collection of data through observation and experimentation, and the formulation and testing of hypotheses. Science, and the nature of scientific knowledge have also become the subject of Philosophy. As science itself has developed, knowledge has developed a broader usage which has been developing within biology/psychology—discussed elsewhere as meta-epistemology, or genetic epistemology, and to some extent related to “theory of cognitive development”.

Other biological domains where “knowledge” might be said to reside include: (iii) the immune system; and (iv) in the DNA of the genetic code.

Example of Rich Text

- To expand the document, click the **more** link.
- Send feedback about the relevance of a search, using the **Yes/No** link to Like or Dislike the quality of the search. IF you dislike an answer, you are asked to provide a comment to improve the Knowledge article.

Log in

What is Genesys Knowledge Center?

Home / What is Genesys Knowledge Center? / How can I assign Genesys Knowledge Center Plugins ...

How can I assign Genesys Knowledge Center Plugins to my agents?

To provide access to Genesys Knowledge Center functionality selectively to your agents, ensure that they have the Knowledge Worker role. Knowledge Center also supports other sets of privileges. For example, roles within Genesys Workspace Desktop Edition can also be configured to provide access to the plugin.

Incorrect answer.

No comment

Categories

Roles

Negative Feedback Comment Field

- Click the **I need more help** button to send a request for proactive help from Genesys Web Engagement.

Important

This feature has been created only for use in conjunction with Genesys Web Engagement. No real message will be sent without integrating your Knowledge Center installation with GWE.

- Click on attachment names to open any attachments in the document. Attachments will open in a new window.

Log in

What is Genesys Knowledge Center?

Home / What is Genesys Knowledge Center? / What is Genesys Knowledge Center?

What is Genesys Knowledge Center?

Genesys Knowledge Center allows you to make the best use of your enterprise knowledge by capturing, storing, and distributing it whenever it is needed. Built by Genesys, this product seamlessly integrates to various Genesys products to provide configuration via Genesys Administrator, reporting and basic analytics via Pulse and agent desktop integration to Workspace Desktop Edition.

[KC-8.5.0-Genesys Knowledge Center Quick Start Guide.pdf](#) [KC-8.5.0-Genesys Knowledge Center API Reference.pdf](#)

Was this helpful? – Yes / No

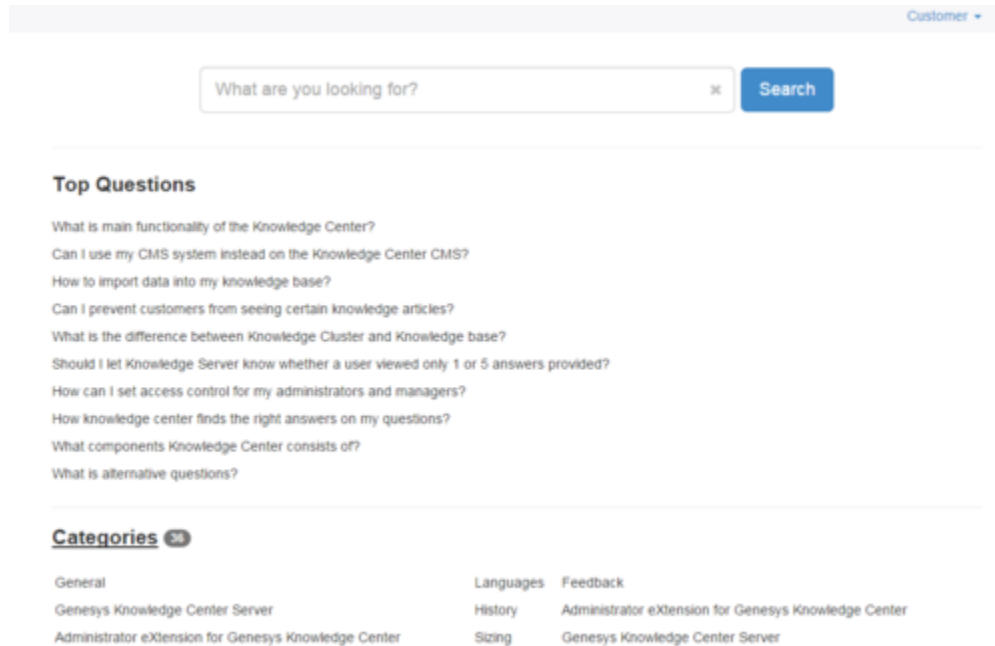
Categories

General

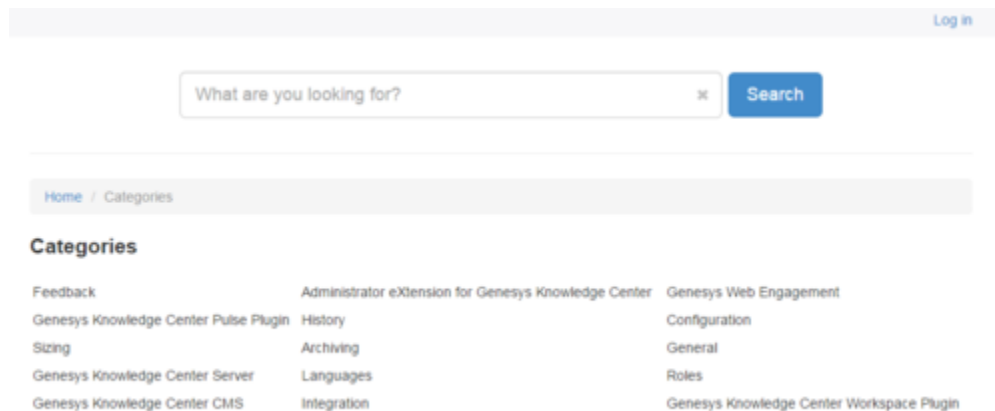
Opening Attachments

Browsing

To browse Categories click the "Categories" link from main page.



Sample UI Main Questions



Sample UI Categories

Customer ▾

Home / Administrator eXtension for Genesys Knowledge Center

What do I need Administrator plugin for?
Knowledge Center Administrator plugin allows to create knowledge bases in knowledge cluster. Please refer to the User's Guide to get more information on the tasks that can be executed in plugin and particular steps of the execution. [more](#)

How to create new knowledge base in Knowledge Center Server?
New knowledge base could be created using Genesys Knowledge Center Plugin for Administrator inside Genesys Administrator Extension application. User Guide will provide you detailed instruction on how to use it. [more](#)

Can I restrict the access to the knowledge base for my agents only?
Yes, knowledge base can be declared as the private and will be accessible to the agent only. Information on how to declare knowledge base to be private can be found in Knowledge Center Administrator Plugin User's Guide. [more](#)

Categories

Genesys Knowledge Center Server

General

General

Genesys Knowledge Center Server

Sample UI Document Categories

Configuring CMS Cluster

Overview

The Knowledge CMS can work in cluster mode (for example, several nodes of the CMS servers can use one data repository). Such configuration is only possible while sorting the data in the Data Base (PostgreSQL, My SQL .etc).

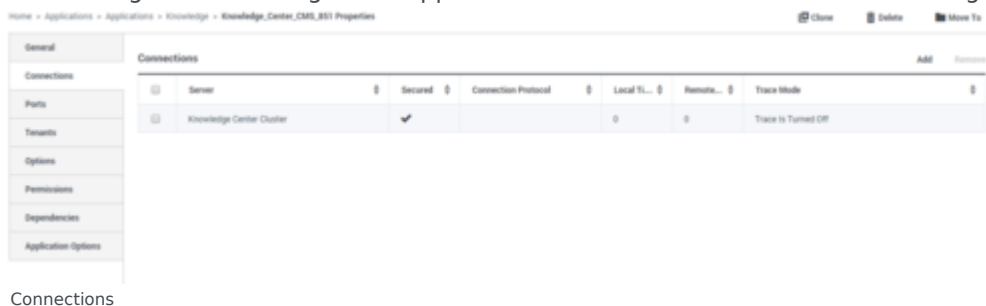
Important

You cannot perform this configuration with a persistent repository file.

How to Configure the Cluster

Start

1. Configure all applications and install several Knowledge CMS servers. Each server works as a node in the Knowledge CMS cluster.
2. Configure Load Balancer on top of the configured nodes to access the CMS cluster.
3. Each configured Knowledge CMS application must be connected to the Knowledge Application Cluster.



4. Add port with name "jgroups" to each CMS application.

Home > Applications > Applications > Knowledge > Knowledge_Center_CMS_R11 Properties

Clone Delete Move To

Ports						
ID	Port	Connection Port...	HA Sync	Listening Mode		
default	7000			Unsecured		
ipgroup	7001			Unsecured		

General
Connections
Ports
Tenants
Options
Permissions
Dependencies
Application Options

Ports

5. Configure Knowledge CMS servers to work with appropriate DB.

- Create Data Base to store CMS data in PostgreSQL, My SQL or other which support JDBC/JNDI (<http://www.oracle.com/technetwork/java/javase/jdbc/index.html> , <http://www.oracle.com/technetwork/java/jndi/index.html>)
- Change jetty.xml each CMS node (<CMS installation folder>/etc/jetty.xml). Change section "datasource for Genesys Knowledge Center CMS" to support particular DB type. For example:

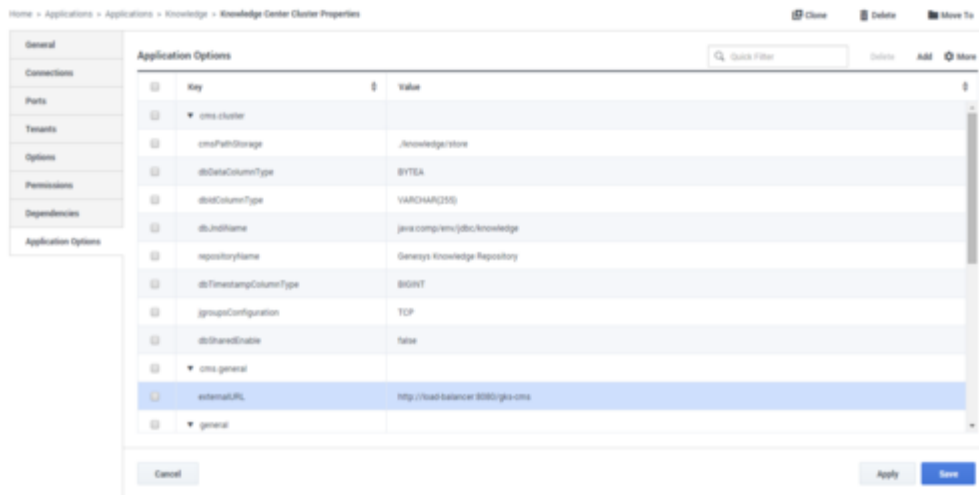
For My SQL:

```
<New id="jdbc/knowledge" class="org.eclipse.jetty.plus.jndi.Resource">
  <Arg></Arg>
  <Arg>jdbc/knowledge</Arg>
  <Arg>
    <New class="com.mysql.jdbc.jdbc2.optional.MysqlConnectionPoolDataSource">
      <Set name="Url">jdbc:mysql://<host and port for access DB>/<CMS DB
name></Set>
      <Set name="User"><username for access CMS DB></Set>
      <Set name="Password"><password for access DB></Set>
    </New>
  </Arg>
</New>
```

For PostgreSQL:

```
<New id="jdbc/knowledge" class="org.eclipse.jetty.plus.jndi.Resource">
  <Arg></Arg>
  <Arg>jdbc/knowledge</Arg>
  <Arg>
    <New class="org.postgresql.ds.PGPoolingDataSource">
      <Set name="User"><username for access CMS DB></Set>
      <Set name="Password"><password for access DB></Set>
      <Set name="DatabaseName"><CMS DB name></Set>
      <Set name="ServerName"><host for access DB></Set>
      <Set name="PortNumber"><post for access DB></Set>
      <Set name="DataSourceName">knowledge</Set>
      <Set name="InitialConnections">5</Set>
      <Set name="MaxConnections">15</Set>
    </New>
  </Arg>
</New>
```

6. Configure the **Application Options** in Knowledge Application Cluster:



Application Options

Name	Description	Value
Section: cms.cluster		
cmsPathStorage	Path for store repository.	Default: ./knowledge/store Valid Values: valid path to folder to store persistent repository file Effective: After restart Use this option for single-node CMS in case of using persistent repository file instead of DB.
dbDataColumnType	Database type for DATA_COLUMN.	Default: BINARY Valid Values: valid type for DATA_COLUMN, BLOB for My SQL, BYTEA for PostgreSQL etc (http://infinispan.org/docs/7.1.x/user_guide/user_guide.html#_jdbc_based_cache_loaders) Effective: After restart
dbIdColumnType	Database type for ID_COLUMN.	Default: VARCHAR(255) Valid Values: valid type for ID_COLUMN (http://infinispan.org/docs/7.1.x/user_guide/user_guide.html#_jdbc_based_cache_loaders) Effective: After restart
dbJndiName	Name of JNDI class in Jetty.	Default: java:comp/env/jdbc/knowledge Valid Values: String "java:comp/env/jdbc/knowledge" or "comp/env/jdbc/knowledge" for running under Jetty8 Effective: After restart
dbSharedEnable	Enables cms instances to store data in shared database.	Default: false

Name	Description	Value
		Valid Values: true, false Effective: After restart To enable CMS cluster set this option to true.
dbTimestampColumnType	Database type for TIMESTAMP_COLUMN.	Default: BIGINT Valid Values: valid type for TIMESTAMP_COLUMN (http://infinispan.org/docs/7.1.x/user_guide/user_guide.html#_jdbc_based_cache_loaders) Effective: After restart
jgroupsConfiguration	Determine the interaction between a server.	Default: TCP Valid Values: JGROUPS_UPD,JGROUPS_TCP,JGROUPS_EC2,TCP,TCP_NIO,TCP_NIO2 Effective: After restart
repositoryName	JNDI database name.	Default: Genesys Knowledge Repository Valid Values: Any string (should not be changed after database creation) Effective: After restart
Section: cms.general		
* Optional externalURL	URL to access CMS cluster via Load balance.	Default: None Valid Values: valid URL Effective: After restart To work with attached document if the CMS cluster is configured.

7. A repository in the provided database is created after this configuration is complete and upon start of the first CMS node. All nodes can then work with this repository as a cluster.

End

Load-Balancing Configuration

Deploying a Cluster

Important

Whenever you deploy a Knowledge Center Server instance, you must configure a Knowledge Center Cluster, even if you only plan on having one server.

Knowledge Center Cluster stores all of the settings and data that are shared by each of the Knowledge Center Server instances that reside within it. This makes it pretty easy to add additional servers as your knowledge needs grow.

Knowledge Center Cluster also serves as the entry point to all client requests sent to Knowledge Center Servers. The cluster application in Genesys Administrator needs to be configured to point to the host and port of the load balancer that will distribute these requests among your Knowledge Center Servers.

Important

If you only have one server deployed in your cluster, you can configure the cluster application to point directly to the host and port of that server.

Configuring Your Load-Balancer Solution

Let's take a look at how you might configure your load balancer to distribute requests between servers. This sample uses an Apache load balancer.

Important

Genesys recommends that you use a round-robin approach to balancing.

Important

If you need more information about load balancing in a Genesys environment, the

Genesys Web Engagement **Load Balancing** page provides some useful background information.

Prerequisites

- Several Knowledge Center Servers should be installed. These servers will be used as cluster nodes (node1, node2, node3, and so on)
- You must have a Genesys Administrator application of type **Application Cluster**
- All Knowledge Center Server applications should be connected to the application cluster

Start

1. Install the Apache HTTP Server (<http://httpd.apache.org/>). The port and host of the installed load balancer should be used in the **Application Cluster** application in Genesys Administrator.
2. Enable these modules (in the `./conf/https.conf` configuration file):
 - `LoadModule proxy_module modules/mod_proxy.so`
 - `LoadModule proxy_ajp_module modules/mod_proxy_ajp.so`
 - `LoadModule proxy_balancer_module modules/mod_proxy_balancer.so`
 - `LoadModule proxy_connect_module modules/mod_proxy_connect.so`
 - `LoadModule proxy_ftp_module modules/mod_proxy_ftp.so`
 - `LoadModule proxy_http_module modules/mod_proxy_http.so`
 - `LoadModule proxy_scgi_module modules/mod_proxy_scgi.so`
3. Configure your proxy settings (http://httpd.apache.org/docs/2.2/mod/mod_proxy_balancer.html):

```
# Proxy
# ProxyPass / balancer://''knowledge_cluster''/ stickysession=JSESSIONID|jsessionid
# ProxyPass / balancer://''knowledge_cluster''/
<Proxy balancer://test_cluster>
    BalancerMember http://host_node_1:port_node_1 route=node1
    BalancerMember http://host__node_2:port_node_1 route=node2
</Proxy>
ProxyRequests On
<Proxy *>
    AddDefaultCharset off
    Order deny,allow
    Allow from all
    #Allow from .example.com
</Proxy>
```

4. In each node in your Jetty server configuration, set `./etc/jetty.xml` like this:

```
<Set name="sessionIdManager">
    <New id="hashIdMgr" class="org.eclipse.jetty.server.session.HashSessionIdManager">
        <Set name="workerName">node1</Set>
    </New>
</Set>
```

-
5. Restart Apache, then restart all of your nodes
 6. All requests that are sent to Apache will be distributed to your cluster nodes. The current configuration supports stickysession mode based on JSESSIONID (http://httpd.apache.org/docs/2.4/mod/mod_proxy_balancer.html#stickyness_implementation)

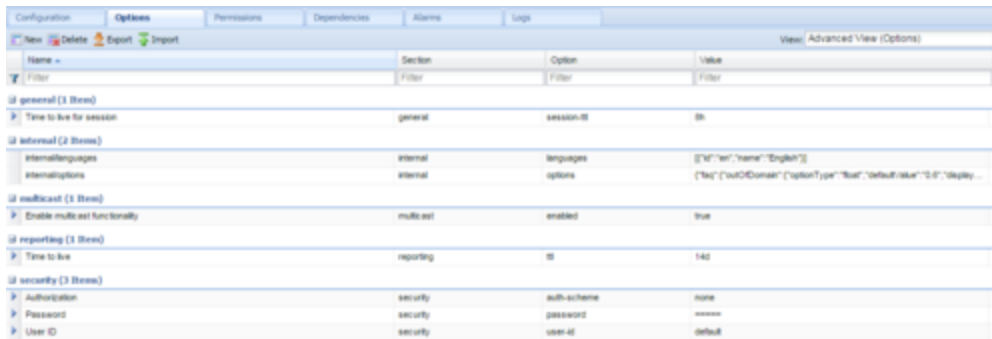
End

Here are couple of sample requests:

- Request to a specific node: http://host_node_1:port_node_1/gks-sample-ui
- Request to the cluster, which will be distributed to any appropriate node:
http://host_load_balancer:port_load_balancer/gks-sample-ui

Configuration Options

Knowledge Center Cluster Application Options



Name	Section	Option	Value
Filter			
general (3 items)			
Time to live for session	general	session-id	30
internal (2 items)			
internalLanguages	internal	languages	[{"id": "en", "name": "English"}]
internalOptions	internal	options	[{"id": "outOfDomain", "optionType": "bool", "defaultValue": "false", "display..."}]
multicast (1 item)			
Enable multicast functionality	multicast	enabled	true
reporting (1 item)			
Time to live	reporting	id	140
security (3 items)			
Authentication	security	auth-scheme	none
Password	security	password	=====
User ID	security	user-id	default

Knowledge Center Cluster Application Configuration Options

Name	Description	Valid Values
Section: cms.cluster		
cmsPathStorage	Path for store repository.	Default: ./knowledge Valid Values: valid path to repository file Effective: After restart Use this option for single-n persistent repository file in
dbDataColumnType	Database type for DATA_COLUMN.	Default: BINARY Valid Values: valid type for SQL, BYTEA for PostgreSQL http://infinispan.org/docs/7.1.x/user_guide/user_guide.html#_jdbc_base Effective: After restart
dbIdColumnType	Database type for ID_COLUMN.	Default: VARCHAR(2) Valid Values: valid type for (http://infinispan.org/docs/7.1.x/user_guide/user_guide.html#_jdbc_base) Effective: After restart
dbJndiName	Name of JNDI class in Jetty.	Default: java:comp/env/jdbc/knowledge Valid Values: String "java:comp/env/jdbc/knowledge" Effective: After restart
dbSharedEnable	Enables cms instances to store data in shared database.	Default: false Valid Values: true, false Effective: After restart

Name	Description	Value
		To enable CMS cluster set t
dbTimestampColumnType	Database type for TIMESTAMP_COLUMN.	Default: BIGINT Valid Values: valid type fo http://infinispan.org/docs/user_guide.html#_jdbc_bas Effective: After restart
jgroupsConfiguration	Determine the interaction between a server.	Default: TCP Valid Values: JGROUPS_UPD,JGROUPS_TC Effective: After restart
repositoryName	JNDI database name.	Default: Genesys Kn Valid Values: Any string (database creation) Effective: After restart
Section: cms.general		
externalURL	Connection to CMS load balancer.	Default: none Valid Values: Valid URL
Section: general		
session-ttl	Specify time that server will store session information while no activities are taking place.	Default: 8h Valid Values: number + u units: d (days), m (minutes) Changes Take Effect: Aft
Section: multicast		
enabled	Specify whether enabled node should use multicast or unicast to discover other servers within the same cluster.	Default: true Valid Values: true, false Changes Take Effect: Aft <div> Important Genesys Knowledge Cent configured to use multica box. Multicast works by s your local network to disc Knowledge Center Server pings and respond. A clus after. This ease of use is disable it in production o could accidentally join yo simply because they rece ping or are misconfigured cluster name. </div>
Section: general		
esReadOnly	Specifies whether a server should use multicast or unicast to discover other servers within the same cluster.	Default: true Valid Values: true, false

Name	Description	Value				
sessionTtl	Specify time that server will store session information while no activities are taking place.	Default: 8h Valid Values: number + units: d (days), m (minutes) Changes Take Effect: After restart				
Section: multicast						
enabled	Specify whether enabled node should use multicast or unicast to discover other servers within the same cluster.	Default: true Valid Values: true, false Changes Take Effect: After restart <div>Important Genesys Knowledge Center Server is configured to use multicast by default. Multicast works by sending a message to your local network to discover other Knowledge Center Servers. Servers respond. A cluster is formed. This ease of use is a disadvantage. This ease of use is a disadvantage. It could accidentally join your network simply because they receive the message or are misconfigured with the cluster name.</div>				
Section: reporting						
geo	Determine the precision of the IP geo-location algorithm.	Default: CITY Valid Values: OFF - Disable geo-location, COUNTRY - Customer's country				
ttl	Specify time that records will be stored in the history.	Default: 14d Valid Values: number + units: d (days), m (minutes) Changes Take Effect: After restart				
Section: log						
all	Specifies the outputs to which an application sends all log events. The log output types must be separated by a comma when more than one output is configured. For example: all = stdout, logfile	Default: stdout Valid Values: (log output types) <table><thead><tr><th>Name</th></tr></thead><tbody><tr><td>stdout</td></tr><tr><td>stderr</td></tr><tr><td>network</td></tr></tbody></table>	Name	stdout	stderr	network
Name						
stdout						
stderr						
network						

Name	Description	V
		<div><div>Name</div><div></div></div>
		memory
		[filename]
		<div>Changes Take Effect: Aft</div>
standard	Specifies the outputs to which an application sends the log events of the Standard level. The log output types must be separated by a comma when more than one output is configured. For example: standard = stderr, network	<div>Default: stdout</div> <div>Valid Values:</div> <div><div>Name</div><div>stdout</div></div> <div><div>stderr</div></div> <div><div>network</div></div>

Name	Description						
		<table><tr><th>Name</th></tr><tr><td></td></tr><tr><td>memory</td></tr><tr><td>[filename]</td></tr></table> <p>Changes Take Effect: Im</p>	Name		memory	[filename]	
Name							
memory							
[filename]							
trace	Specifies the outputs to which an application sends the log events of the Trace level and higher (that is, log events of the Standard, Interaction, and Trace levels). The log outputs must be separated by a comma when more than one output is configured. For example: trace = stderr, network.	<p>Default: stdout</p> <p>Valid Values:</p> <table><tr><th>Name</th></tr><tr><td>stdout</td></tr><tr><td>stderr</td></tr><tr><td>network</td></tr><tr><td>memory</td></tr></table>	Name	stdout	stderr	network	memory
Name							
stdout							
stderr							
network							
memory							

Name	Description							
		<table><tr><th>Name</th></tr><tr><td>[filename]</td></tr></table> <p>Changes Take Effect: Im</p>	Name	[filename]				
Name								
[filename]								
verbose	Determines whether a log output is created. If it is, specifies the minimum level of log events generated. The log events levels, starting with the highest priority level, are Standard, Interaction, Trace, and Debug.	<p>Default: standard</p> <p>Valid Values:</p> <table><tr><th>Name</th></tr><tr><td>all</td></tr><tr><td>debug</td></tr><tr><td>trace</td></tr><tr><td>interaction</td></tr><tr><td>standard</td></tr></table>	Name	all	debug	trace	interaction	standard
Name								
all								
debug								
trace								
interaction								
standard								

Name	Description	
		<div> <div>Name</div> <div> <div>none</div> </div> </div> <div>Changes Take Effect: Im</div>
segment	Specifies whether there is a segmentation limit for a log file. If there is, sets the mode of measurement, along with the maximum size. If the current log segment exceeds the size set by this option, the file is closed and a new one is created. This option is ignored if log output is not configured to be sent to a log file.	<div> <div>Default: 1000</div> <div>Valid Values:</div> <div> <div>Name</div> <div> <div>false</div> <div><number> KB or <number></div> <div><number> MB</div> <div><number> hr</div> </div> </div> <div>Changes Take Effect: Aft</div> </div>
expire	Determines whether log files expire. If they do, sets the measurement for determining when they expire, along with the maximum number of files (segments) or days before the files are removed. This option is ignored if log output is not configured to be sent to a log file.	<div> <div>Default: 3</div> <div>Valid Values:</div> <div> <div>Name</div> <div> <div>false</div> <div><number> file or <number></div> </div> </div> </div>

Name	Description	
		<div data-bbox="1369 302 1624 525"> <p>Name</p> <p><number> day</p> </div> <div data-bbox="1369 583 1624 609"> <p>Changes Take Effect: Aft</p> </div> <div data-bbox="1369 642 1624 730"> <p>Important</p> <p>If an option's value is not a valid value, it will autom</p> </div>
affectedLoggers	<p>Verbosity settings are explicitly applied for the following loggers:</p> <ul style="list-style-type: none"> Loggers that are not declared explicitly in the <i>log4j2.xml</i> configuration file. Loggers that are specified explicitly in the <i>log4j2.xml</i> and are specified in the value for this affectedLoggers option. <p>For other loggers specified in <i>log4j2.xml</i>, but not mentioned in the value for this option, the verbosity level is not re-applied.</p> <p>Here is a use case for when you might need to set this option:</p> <ul style="list-style-type: none"> Cassandra needs to write error messages to a log file, and at the same time, Genesys components also need to write debug messages to the log file. <p>To resolve this use case, you would:</p> <ol style="list-style-type: none"> Specify the following logger in <i>log4j2.xml</i>: <code><logger name="org.apache.cassandra" level="error" additivity="false"></code> Do not include <i>org.apache.cassandra</i> in the value for the affectedLoggers option. The default <i>log4j2.xml</i> file contains the following logger: <code><logger name="com.genesyslab.platform" level="info" additivity="false"></code> Include <i>com.genesyslab.platform</i> in the value for the affectedLoggers option. Set the verbose option to <i>debug</i>. <p>In the sample above, the value of affectedLoggers should be <i>com.genesyslab.platform</i>. Error (but not debug</p>	<p>Default: None</p> <p>Valid Values: The names of loggers, separated by a semicolon (;), specified in the <i>log4j2.xml</i> file. Examples: <i>com.genesyslab.wmcbcore</i>, <i>org.elasticsearch</i>, <i>com.genesys.knowledge.as</i>, <i>com.genesys.knowledge.se</i>, <i>com.genesys.elasticsearch</i>, <i>com.genesys.knowledge.as</i>, <i>com.genesys.knowledge.se</i>, <i>com.genesys.knowledge.as</i>, <i>com.genesys.knowledge.se</i>.</p> <p>Changes Take Effect: Im</p>

Name	Description	Valid Values				
	or info) messages from Cassandra will be available in logs, and debug messages from <i>com.genesyslab.platform</i> will be available in logs.					
time_format	Specifies how to represent, in a log file, the time when an application generates log records. A log record's time field in the ISO 8601 format looks like this: 2001-07-24T04:58:10.123	<div>Default: time</div> <div>Valid Values:<table><tr><th>Name</th></tr><tr><td>time</td></tr><tr><td>locale</td></tr><tr><td>ISO8601</td></tr></table></div> <div>Changes Take Effect: Immediately</div>	Name	time	locale	ISO8601
Name						
time						
locale						
ISO8601						
time_convert	Specifies the system in which an application calculates the log record time when generating a log file. The time is converted from the time in seconds since 00:00:00 UTC, January 1, 1970.	<div>Default: local</div> <div>Valid Values:<table><tr><th>Name</th></tr><tr><td>local</td></tr><tr><td>utc</td></tr></table></div>	Name	local	utc	
Name						
local						
utc						

Name	Description	Valid Values
Section: security		
auth-scheme	Specifies the HTTP authentication scheme used to secure REST API requests to the Knowledge Server. With the Basic scheme, clients must be authenticated with a user ID and password.	Default: none Valid Values: none, basic Changes Take Effect: After restart
user-id	The user identifier (login) used in authentication for the REST API.	Default: n/a Valid Values: string Changes Take Effect: After restart
password	The user password used in authentication for the REST API.	Default: n/a Valid Values: string Changes Take Effect: After restart
Section: internal		
Important Knowledge Center Server uses this section to store internal initialization parameters. Do not attempt to change these options.		

Knowledge Center Server Application Options

Name	Section	Option	Value
archiving (4 Rows)			
archive type	archiving	type	tar
enable archiving functionality	archiving	enabled	true
local path archives stored in	archiving	path	
archiving	archiving	archiving	true
log (6 Rows)			
log all	log	all	stdout,log_node,log
log engine	log	engine	20
log segment	log	segment	10000
log standard	log	standard	
log trace	log	trace	
log verbose	log	verbose	all

Knowledge Center Server Application Configuration Options

Name	Description	Valid Values
Section: archiving		
enabled	Specifies whether a node will allow to execute archiving using its API. Enabling archiving on the node does not affect other nodes of the cluster. Archiving is resource consuming functionality - use it wisely.	Default: true Valid Values: true, false Changes Take Effect: After restart

Name	Description	Default: tar Valid Values: tar, zip, cpio Changes Take Effect: After restart
type	Defines format of resulted archive will be stored in.	
path	Path to the stored archive. The archive will be stored as <path>/history_<requested_date_range>.<archive>	Default: none Valid Values: string Changes Take Effect: After restart
Section: security		
tls	Client: 1 - perform TLS handshake immediately after connecting to server. 0 - do not turn on TLS immediately but autodetect can still work.	Boolean value. Possible values are "1"/"0", "true"/"false". Example: <ul style="list-style-type: none">"tls=1"
provider	Explicit selection of security provider to be used. For example, MSCAPI and PKCS11 providers can contain all other parameters in their internal database. This parameter allow configuration of TLS through security provider tools.	"PEM", "MSCAPI", "PKCS11" Not case-sensitive. Example: <ul style="list-style-type: none">"provider=MSCAPI"
certificate	Specifies location of X.509 certificate to be used by application. MSCAPI provider keeps certificates in internal database and can identify them by hash code; so called thumbprint. In Java, PKCS#11 provider does not allow selection of the certificate; it must be configured using provider tools. Note: When using autodetect (upgrade) TLS connection, this option MUST be specified in application configuration, otherwise Configuration Server would return empty TLS parameters even if other options are set.	PEM provider: path to certificate file in PEM format. Path can contain backward slash characters. MSCAPI provider: thumbprint of certificate in hexadecimal SHA-1 hash code. Whitespace characters are not allowed. PKCS11 provider: thumbprint of certificate in hexadecimal. Examples: <ul style="list-style-type: none">"certificate= C:\certs\cert-3-cert.pem""certificate=A4 7E 15 BE 89 FD 46 F0"
certificate-key	Specifies location of PKCS#8 private key to be used in pair with the certificate by application. MSCAPI provider keeps private keys paired with certificates in internal database. In Java, PKCS#11 provider does not allow selection of the private key; it must be configured using provider tools.	PEM provider: path to private key file without password. Path can use both forward and backward slash characters. <ul style="list-style-type: none">MSCAPI provider: thumbprint of private key is taken from "certificate" field.PKCS11 provider: thumbprint of private key. Examples: <ul style="list-style-type: none">"certificate-key= C:\certs\cert-3-cert-key.pem"

Name	Description	Value
		cert-3-key.pem"
trusted-ca	<p>Specifies location of a X.509 certificate to be used by application to validate remote party certificates. The certificate is designated as Trusted Certification Authority certificate and application will only trust remote party certificates signed with the CA certificate.</p> <p>MSCAPI provider keeps CA certificates in internal database and can identify them by hash code; so called thumbprint. In Java, PKCS#11 provider does not allow selection of the CA certificate; it must be configured using provider tools.</p>	<p>PEM provider: path to certificate in PEM format. Path can contain backward slash characters.</p> <p>MSCAPI provider: thumbprint of certificate in hexadecimal SHA-1 hash code. Whitespace characters are not allowed in string. PKCS11 provider: the certificate name.</p> <p>Examples:</p> <ul style="list-style-type: none"> "trusted-ca= C:\certs\cert-3-key.pem" "trusted-ca=A4 7E 15 BE 89 FD 46 F0 ..."
tls-mutual	Has meaning only for server application. Client applications ignore this value. When turned on, server will require connecting clients to present their certificates and validate the certificates the same way as client applications do.	<p>Boolean value.</p> <p>Possible values are "1"/"0", "true"/"false".</p> <p>Example:</p> <ul style="list-style-type: none"> "tls-mutual=1"
tls-crl	Applications will use CRL during certificate validation process to check if the (seemingly valid) certificate was revoked by CA. This option is useful to stop usage of leaked certificates by unauthorized parties.	<p>All providers: path to CRL file in PEM format. Path can contain forward and backward slash characters.</p> <p>Example:</p> <ul style="list-style-type: none"> "tls-crl= C:\certs\crl.pem"
tls-target-name-check	When set to "host", enables matching of certificate's Alternative Subject Name or Subject fields against expected host name. PSDK supports DNS names and IP addresses as expected host names.	<p>"host" or none. Not case sensitive.</p> <p>Example:</p> <ul style="list-style-type: none"> "tls-target-name-check=host"
cipher-list	Used to calculate enabled cipher suites. Only ciphers present in both the cipher suites supported by security provider and the cipher-list parameter will be valid.	<p>String consisting of space separated cipher suite names. Information about cipher suites can be found online.</p> <p>Example:</p> <ul style="list-style-type: none"> "cipher-list= TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 TLS_ECDH_RSA_WITH_AES_128_GCM_SHA256"
fips140-enabled	PSDK Java: when set to true, effectively is the same as setting "provider=PKCS11" since only PKCS11 provider can support FIPS-140. If set to true while using other provider type, PSDK will throw exception.	<p>Boolean value.</p> <p>Possible values are "1"/"0", "true"/"false".</p>

Name	Description	Valid Values					
		Example: <ul style="list-style-type: none">"fips140-enabled=					
sec-protocol	Starting with PSDK release 8.5.1, an application can specify the exact protocol to send and accept secure connection requests on one or more of its connections.	String value. Possible values are "SSLv2" and "TLSv12". Example: <ul style="list-style-type: none">"sec-protocol=TLS					
Section: log							
all	Specifies the outputs to which an application sends all log events. The log output types must be separated by a comma when more than one output is configured. For example: all = stdout, logfile	<div>Default: stdout</div> <div>Valid Values: (log output)</div> <table><thead><tr><th>Name</th></tr></thead><tbody><tr><td>stdout</td></tr><tr><td>stderr</td></tr><tr><td>network</td></tr><tr><td>memory</td></tr></tbody></table>	Name	stdout	stderr	network	memory
Name							
stdout							
stderr							
network							
memory							

Name	Description							
		<table><tr><th>Name</th></tr><tr><td>[filename]</td></tr></table>	Name	[filename]				
Name								
[filename]								
		<p>Changes Take Effect: After the next restart of the service.</p> <p>Default: stdout</p> <p>Valid Values:</p> <table><tr><th>Name</th></tr><tr><td>stdout</td></tr><tr><td>stderr</td></tr><tr><td>network</td></tr><tr><td>memory</td></tr><tr><td>[filename]</td></tr></table>	Name	stdout	stderr	network	memory	[filename]
Name								
stdout								
stderr								
network								
memory								
[filename]								
standard	Specifies the outputs to which an application sends the log events of the Standard level. The log output types must be separated by a comma when more than one output is configured. For example: standard = stderr, network							

Name	Description	Valid Values						
		Changes Take Effect: Immediately						
trace	Specifies the outputs to which an application sends the log events of the Trace level and higher (that is, log events of the Standard, Interaction, and Trace levels). The log outputs must be separated by a comma when more than one output is configured. For example: trace = stderr, network.	Default: stdout Valid Values: <table><tr><th>Name</th></tr><tr><td>stdout</td></tr><tr><td>stderr</td></tr><tr><td>network</td></tr><tr><td>memory</td></tr><tr><td>[filename]</td></tr></table> Changes Take Effect: Immediately	Name	stdout	stderr	network	memory	[filename]
Name								
stdout								
stderr								
network								
memory								
[filename]								
verbose	Determines whether a log output is created. If it is, specifies the minimum level of log events generated. The log events levels, starting with the highest priority level, are Standard, Interaction, Trace, and Debug.	Default: standard Valid Values: <table><tr><th>Name</th></tr><tr><td>all</td></tr></table>	Name	all				
Name								
all								

Name	Description	V						
		<table><tr><th>Name</th></tr><tr><td>debug</td></tr><tr><td>trace</td></tr><tr><td>interaction</td></tr><tr><td>standard</td></tr><tr><td>none</td></tr></table>	Name	debug	trace	interaction	standard	none
		Name						
		debug						
		trace						
		interaction						
		standard						
none								
Changes Take Effect: Im								
segment	Specifies whether there is a segmentation limit for a log file. If there is, sets the mode of measurement, along with the maximum size. If the current log segment exceeds the size set by this option, the file is closed and a new one is created. This option is ignored if log output is not configured to be sent to a log file.	<div>Default: 1000</div> <div>Valid Values:</div> <table><tr><th>Name</th></tr><tr><td>false</td></tr><tr><td><number> KB or <number></td></tr></table>	Name	false	<number> KB or <number>			
		Name						
		false						
<number> KB or <number>								

Name	Description	
		<div> <div>Name</div> <div><number> MB</div> <div><number> hr</div> </div> <div>Changes Take Effect: Aft</div>
expire	Determines whether log files expire. If they do, sets the measurement for determining when they expire, along with the maximum number of files (segments) or days before the files are removed. This option is ignored if log output is not configured to be sent to a log file.	<div> <div>Default: 3</div> <div>Valid Values:</div> <div> <div>Name</div> <div>false</div> <div><number> file or <number></div> <div><number> day</div> </div> </div> <div>Changes Take Effect: Aft</div> <div> <div>Important</div> <div>If an option's value is not valid values, it will autom</div> </div>
affectedLoggers	<p>Verbosity settings are explicitly applied for the following loggers:</p> <ul style="list-style-type: none"> Loggers that are not declared explicitly in the <i>log4j2.xml</i> configuration file. Loggers that are specified explicitly in the <i>log4j2.xml</i> and are specified in the value for this affectedLoggers option. 	<div>Default: None</div> <div>Valid Values: The names semicolon (;), specified in t</div> <div>com.genesyslab.webme.co</div> <div>Changes Take Effect: Im</div>

Name	Description					
	<p>For other loggers specified in <i>log4j2.xml</i>, but not mentioned in the value for this option, the verbosity level is not re-applied.</p> <p>Here is a use case for when you might need to set this option:</p> <ul style="list-style-type: none">Cassandra needs to write error messages to a log file, and at the same time, Genesys components also need to write debug messages to the log file. <p>To resolve this use case, you would:</p> <ol style="list-style-type: none">Specify the following logger in <i>log4j2.xml</i>: <i><logger name="org.apache.cassandra" level="error" additivity="false"></i>Do not include <i>org.apache.cassandra</i> in the value for the affectedLoggers option.The default <i>log4j2.xml</i> file contains the following logger: <i><logger name="com.genesyslab.platform" level="info" additivity="false"></i>Include <i>com.genesyslab.platform</i> in the value for the affectedLoggers option.Set the verbose option to <i>debug</i>. <p>In the sample above, the value of affectedLoggers should be <i>com.genesyslab.platform</i>. Error (but not debug or info) messages from Cassandra will be available in logs, and debug messages from <i>com.genesyslab.platform</i> will be available in logs.</p>					
time_format	<p>Specifies how to represent, in a log file, the time when an application generates log records. A log record's time field in the ISO 8601 format looks like this: 2001-07-24T04:58:10.123</p>	<div><div>Default: time</div><div>Valid Values:</div><table><thead><tr><th>Name</th></tr></thead><tbody><tr><td>time</td></tr><tr><td>locale</td></tr><tr><td>ISO8601</td></tr></tbody></table></div>	Name	time	locale	ISO8601
Name						
time						
locale						
ISO8601						

Name	Description	
		<div><div><div>Name</div><div></div></div><div>Changes Take Effect: Im</div></div>
time_convert	Specifies the system in which an application calculates the log record time when generating a log file. The time is converted from the time in seconds since 00:00:00 UTC, January 1, 1970.	<div><div><div><div>Default: local</div><div>Valid Values:</div><div><div><div>Name</div><div></div></div><div><div>local</div><div>utc</div></div></div><div>Changes Take Effect: Im</div></div></div></div>

Knowledge Center CMS Application Options

Name	Section	Option	Value
affectedLoggers	log	affectedLoggers	c:\an\genesys\knowledge.cms.asp\03\oldmonit... c:\an\genesys\knowled...
logoff	log	all	stdout_cms_log.log
logonpipe	log	engine	20
logsegment	log	segment	10000
logstandard	log	standard	
logtrace	log	trace	
logverbose	log	verbose	all

Knowledge Center CMS Application Options

Name	Description	
Section: log		
all	Specifies the outputs to which an application	Default: stdout

Name	Description	Valid Values: (log output						
		<table><tr><th>Name</th></tr><tr><td>stdout</td></tr><tr><td>stderr</td></tr><tr><td>network</td></tr><tr><td>memory</td></tr><tr><td>[filename]</td></tr></table>	Name	stdout	stderr	network	memory	[filename]
Name								
stdout								
stderr								
network								
memory								
[filename]								
	sends all log events. The log output types must be separated by a comma when more than one output is configured. For example: all = stdout, logfile	Changes Take Effect: Aft						

Name	Description	Valid Values:						
standard	Specifies the outputs to which an application sends the log events of the Standard level. The log output types must be separated by a comma when more than one output is configured. For example: standard = stderr, network	Default: stdout						
		Valid Values:						
		<table><tr><th>Name</th></tr><tr><td>stdout</td></tr><tr><td>stderr</td></tr><tr><td>network</td></tr><tr><td>memory</td></tr><tr><td>[filename]</td></tr></table>	Name	stdout	stderr	network	memory	[filename]
		Name						
		stdout						
		stderr						
network								
memory								
[filename]								
trace	Specifies the outputs to which an application sends the log events of the Trace level and higher (that is, log events of the Standard, Interaction, and Trace levels). The log outputs must be separated by a comma when more than one output is configured. For example: trace = stderr, network.	Default: stdout						
		Valid Values:						
		<table><tr><th>Name</th></tr><tr><td>stdout</td></tr><tr><td>stderr</td></tr><tr><td>network</td></tr></table>	Name	stdout	stderr	network		
		Name						
stdout								
stderr								
network								

Name	Description													
		<table><tr><th>Name</th></tr><tr><td></td></tr><tr><td>memory</td></tr><tr><td>[filename]</td></tr><tr><td colspan="2">Changes Take Effect: Im</td></tr></table>	Name		memory	[filename]	Changes Take Effect: Im							
Name														
memory														
[filename]														
Changes Take Effect: Im														
verbose	Determines whether a log output is created. If it is, specifies the minimum level of log events generated. The log events levels, starting with the highest priority level, are Standard, Interaction, Trace, and Debug.	<table><tr><td colspan="2">Default: standard</td></tr><tr><td colspan="2">Valid Values:</td></tr><tr><th>Name</th><td></td></tr><tr><td>all</td><td></td></tr><tr><td>debug</td><td></td></tr><tr><td>trace</td><td></td></tr></table>	Default: standard		Valid Values:		Name		all		debug		trace	
Default: standard														
Valid Values:														
Name														
all														
debug														
trace														

Name	Description	Valid Values					
		<table><tr><th>Name</th></tr><tr><td>interaction</td></tr><tr><td>standard</td></tr><tr><td>none</td></tr></table> <p>Changes Take Effect: Immediately</p>	Name	interaction	standard	none	
Name							
interaction							
standard							
none							
segment	Specifies whether there is a segmentation limit for a log file. If there is, sets the mode of measurement, along with the maximum size. If the current log segment exceeds the size set by this option, the file is closed and a new one is created. This option is ignored if log output is not configured to be sent to a log file.	<p>Default: 1000</p> <p>Valid Values:</p> <table><tr><th>Name</th></tr><tr><td>false</td></tr><tr><td><number> KB or <number></td></tr><tr><td><number> MB</td></tr><tr><td><number> hr</td></tr></table> <p>Changes Take Effect: After the next log file is created</p>	Name	false	<number> KB or <number>	<number> MB	<number> hr
Name							
false							
<number> KB or <number>							
<number> MB							
<number> hr							
expire	Determines whether log files expire. If they do, sets the measurement for determining when	<p>Default: 3</p>					

Name	Description	Valid Values:				
	they expire, along with the maximum number of files (segments) or days before the files are removed. This option is ignored if log output is not configured to be sent to a log file.	<table><tr><th>Name</th></tr><tr><td>false</td></tr><tr><td><number> file or <number></td></tr><tr><td><number> day</td></tr></table> <p>Changes Take Effect: Aft</p> <p>Important If an option's value is not valid values, it will autom</p>	Name	false	<number> file or <number>	<number> day
Name						
false						
<number> file or <number>						
<number> day						
affectedLoggers	<p>Verbosity settings are explicitly applied for the following loggers:</p> <ul style="list-style-type: none">Loggers that are not declared explicitly in the <i>log4j2.xml</i> configuration file.Loggers that are specified explicitly in the <i>log4j2.xml</i> and are specified in the value for this affectedLoggers option. <p>For other loggers specified in <i>log4j2.xml</i>, but not mentioned in the value for this option, the verbosity level is not re-applied. Here is a use case for when you might need to set this option:</p> <ul style="list-style-type: none">Cassandra needs to write error messages to a log file, and at the same time, Genesys components also need to write debug messages to the log file. <p>To resolve this use case, you would:</p> <ol style="list-style-type: none">Specify the following logger in <i>log4j2.xml</i>: <i><logger name="org.apache.cassandra" level="error" additivity="false"></i>Do not include <i>org.apache.cassandra</i> in the value for the affectedLoggers option.	<p>Default: None</p> <p>Valid Values: The names semicolon (;), specified in <i>com.genesyslab.webme.co</i></p> <p>Changes Take Effect: Im</p>				

Name	Description	Value				
	<div>3. The default <i>log4j2.xml</i> file contains the following logger: <i><logger name="com.genesyslab.platform" level="info" additivity="false"></i></div> <div>4. Include <i>com.genesyslab.platform</i> in the value for the affectedLoggers option.</div> <div>5. Set the verbose option to <i>debug</i>.</div> <div>In the sample above, the value of affectedLoggers should be <i>com.genesyslab.platform</i>. Error (but not debug or info) messages from Cassandra will be available in logs, and debug messages from <i>com.genesyslab.platform</i> will be available in logs.</div>					
time_format	Specifies how to represent, in a log file, the time when an application generates log records. A log record's time field in the ISO 8601 format looks like this: 2001-07-24T04:58:10.123	<div>Default: time</div> <div>Valid Values:</div> <table><thead><tr><th>Name</th></tr></thead><tbody><tr><td>time</td></tr><tr><td>locale</td></tr><tr><td>ISO8601</td></tr></tbody></table> <div>Changes Take Effect: Immediately</div>	Name	time	locale	ISO8601
Name						
time						
locale						
ISO8601						
time_convert	Specifies the system in which an application calculates the log record time when generating a log file. The time is converted from the time in seconds since 00:00:00 UTC, January 1, 1970.	<div>Default: local</div> <div>Valid Values:</div> <table><thead><tr><th>Name</th></tr></thead><tbody><tr><td>local</td></tr></tbody></table>	Name	local		
Name						
local						

Name	Description	Value
		<div><div><div>Name</div><div></div></div><div><div>utc</div></div></div> <div>Changes Take Effect: Immediately</div>
Section: security		
tls	<p>Client:</p> <p>1 - perform TLS handshake immediately after connecting to server. 0 – do not turn on TLS immediately but autodetect can still work.</p>	<p>Boolean value.</p> <p>Possible values are "1"/"0", "true"/"false".</p> <p>Example:</p> <ul style="list-style-type: none">"tls=1"
provider	<p>Explicit selection of security provider to be used. For example, MSCAPI and PKCS11 providers can contain all other parameters in their internal database. This parameter allow configuration of TLS through security provider tools.</p>	<p>"PEM", "MSCAPI", "PKCS11"</p> <p>Not case-sensitive.</p> <p>Example:</p> <ul style="list-style-type: none">"provider=MSCAPI"
certificate	<p>Specifies location of X.509 certificate to be used by application.</p> <p>MSCAPI provider keeps certificates in internal database and can identify them by hash code; so called thumbprint.</p> <p>In Java, PKCS#11 provider does not allow selection of the certificate; it must be configured using provider tools.</p> <p>Note: When using autodetect (upgrade) TLS connection, this option MUST be specified in application configuration, otherwise Configuration Server would return empty TLS parameters even if other options are set.</p>	<p>PEM provider: path to certificate in PEM format. Path can contain backward slash characters.</p> <p>MSCAPI provider: thumbprint of certificate in hexadecimal SHA-1 hash code. Path can contain whitespace characters and backslash characters. PKCS11 provider: thumbprint of certificate in hexadecimal SHA-1 hash code.</p> <p>Examples:</p> <ul style="list-style-type: none">"certificate= C:\certs\cert-3-cert.pem""certificate=A4 7E 8D 9A 15 BE 89 FD 46 F0 8A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A"
certificate-key	<p>Specifies location of PKCS#8 private key to be used in pair with the certificate by application.</p> <p>MSCAPI provider keeps private keys paired with certificates in internal database. In Java, PKCS#11 provider does not allow selection of the private key; it must be configured using provider tools.</p>	<p>PEM provider: path to private key file without password. Path can use both forward and backward slash characters.</p>

Name	Description	Value
	must be configured using provider tools.	<ul style="list-style-type: none"> • MSCAPI provider: thumbprint key is taken from "certificate" field. • PKCS11 provider: thumbprint key is taken from "certificate" field. <p>Examples:</p> <ul style="list-style-type: none"> • "certificate-key= C:\cert-3-key.pem"
trusted-ca	<p>Specifies location of a X.509 certificate to be used by application to validate remote party certificates. The certificate is designated as Trusted Certification Authority certificate and application will only trust remote party certificates signed with the CA certificate.</p> <p>MSCAPI provider keeps CA certificates in internal database and can identify them by hash code; so called thumbprint. In Java, PKCS#11 provider does not allow selection of the CA certificate; it must be configured using provider tools.</p>	<p>PEM provider: path to certificate in PEM format. Path can contain backward slash characters.</p> <p>MSCAPI provider: thumbprint. hexadecimal SHA-1 hash code. Whitespace characters are not allowed in string. PKCS11 provider: thumbprint.</p> <p>Examples:</p> <ul style="list-style-type: none"> • "trusted-ca= C:\cert-3-key.pem" • "trusted-ca=A4 7E 15 BE 89 FD 46 F0"
tls-mutual	Has meaning only for server application. Client applications ignore this value. When turned on, server will require connecting clients to present their certificates and validate the certificates the same way as client applications do.	<p>Boolean value.</p> <p>Possible values are "1"/"0", "true"/"false".</p> <p>Example:</p> <ul style="list-style-type: none"> • "tls-mutual=1"
tls-crl	Applications will use CRL during certificate validation process to check if the (seemingly valid) certificate was revoked by CA. This option is useful to stop usage of leaked certificates by unauthorized parties.	<p>All providers: path to CRL file in PEM format. Path can contain forward and backward slash characters.</p> <p>Example:</p> <ul style="list-style-type: none"> • "tls-crl= C:\certs\crl.pem"
tls-target-name-check	When set to "host", enables matching of certificate's Alternative Subject Name or Subject fields against expected host name. PSDK supports DNS names and IP addresses as expected host names.	<p>"host" or none. Not case sensitive.</p> <p>Example:</p> <ul style="list-style-type: none"> • "tls-target-name-check=host"
cipher-list	Used to calculate enabled cipher suites. Only ciphers present in both the cipher suites supported by security provider and the cipher-list parameter will be valid.	<p>String consisting of space separated cipher suite names. Information about cipher suites can be found online.</p> <p>Example:</p> <ul style="list-style-type: none"> • "cipher-list= TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256"

Name	Description	Value
		TLS_ECDHE_RSA_... TLS_ECDHE_RSA_... TLS_ECDH_RSA_W...
fips140-enabled	PSDK Java: when set to true, effectively is the same as setting "provider=PKCS11" since only PKCS11 provider can support FIPS-140. If set to true while using other provider type, PSDK will throw exception.	Boolean value. Possible values are "1"/"0" "true"/"false". Example: <ul style="list-style-type: none">"fips140-enabled=
sec-protocol	Starting with PSDK release 8.5.1, an application can specify the exact protocol to send and accept secure connection requests on one or more of its connections.	String value. Possible values are "SSLv2" "TLSv12". Example: <ul style="list-style-type: none">"sec-protocol=TLS

Sizing

Important

The exact deployment architecture and solution size will vary depending on your hardware and your ability to fine-tune the deployed system to get the best performance on your equipment and with your particular user load. However, the following estimates may give you some basic ideas on how to size your deployment.

Hardware Sizing Information

Genesys Knowledge Center Server

	Minimal	Recommended
CPU	Multicore (8+)	
RAM	8GB	16GB
Disk Space	100GB or more, depending on the number of knowledge bases and the depth of the history	

Genesys Knowledge Center CMS

	Minimal	Recommended
CPU	Multicore (8+)	
RAM	8GB	16GB
Disk Space	10GB or more, depending on the number of knowledge bases	

Recommended Software Configuration

- **OS version**—Linux 6 x64 or higher, Windows Server 2008R2 x64 or higher
- **Java version**—Java version 1.7 or higher, 64-Bit Server VM

Java Options	Initial heap size (Xms)	Maximum heap size (Xmx)
Genesys Knowledge Center Server (without archiving)	4096m	4096m

Java Options	Initial heap size (Xms)	Maximum heap size (Xmx)
Genesys Knowledge Center Server (with archiving)	4096m	8192m
Genesys Knowledge Center CMS	1024m	1024m

XMS and Xms parameters used by Knowledge Center Server and Knowledge Center CMS are changed in the setenv.bat or setenv.sh files.

Important

It is strongly advised to not set the Xmx larger than 32Gb.

Important

In most cases it is recommended to plan your deployment with 50% of your memory allocated to the running application. The remaining 50% is used for the OS filesystem cache allowing for different software (including Genesys Knowledge Center) to work faster and minimizing disk operations.