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 Administrator Extension Deployment Guide

Genesys Administrator 8.1.3

12/29/2021

Table of Contents

Genesys Administrator Extension Deployment Guide

Welcome to the Framework 8.1 Genesys Administrator Extension Deployment Guide. This document describes the deployment, starting and stopping, and troubleshooting procedures that are relevant to Genesys Administrator Extension.

About Genesys Administrator Extension

Genesys Administrator Extension (GAX), part of the Genesys Framework, is a web-based graphical user interface (GUI) that provides advanced administrative and operational functionality that is targeted to Hosted Service Providers as well as Enterprise customers. In brief, you will find the following information in this guide:

- How to deploy Genesys Administrator Extension.
- How to access Genesys Administrator Extension.
- Suggestions for troubleshooting your Genesys Administrator Extension installation.

Intended Audience

This document is intended primarily for system integrators, system administrators, contact center managers, and operations personnel. It has been written with the assumption that you have a basic understanding of:

- Computer-telephony integration (CTI) concepts, processes, terminology, and applications
- Network design and operation
- Your own network configurations

You should also be familiar with Genesys Framework architecture and functions, as described in the Framework Deployment Guide.

Contacting Genesys Technical Support

If you have purchased support directly from Genesys, please contact Genesys Technical Support.

Before contacting technical support, please refer to the Genesys Care Program Guide for complete contact information and procedures.

About This Book

The following list explains different features of GAX:

Overview

This chapter introduces you to the core features of GAX and its architecture.

Setting up GAX

This chapter explains how to deploy GAX.

Accessing GAX

This chapter explains how to access GAX.

Troubleshooting

This chapter explains how to troubleshoot GAX.

Role Privileges

This chapter explains all of the role privileges used by GAX.

Configuration Options

This chapter explains all of the configuration options used by GAX.

Overview

This chapter provides a brief description of Genesys Administrator Extension and its architecture.

This chapter contains the following sections:

- Genesys Administrator Extension
- Architecture
- Database Size Requirements

Genesys Administrator Extension

Genesys Administrator Extension (GAX) is an application that provides additional administrative capabilities to both technical and business users of Genesys Administrator. Currently, the following GAX core modules are supported:

- Solution Deployment
- Operational Parameter Management
- Audio Resource Management
- License Usage Reporting (for GAX 8.1.301 releases or lower)
- Account Management
- GAX/Genesys Administrator Single-Sign-on

Genesys Administrator Extension also supports plug-in resources from other Genesys products, such as EZPulse. Refer to Managing GAX Compatible Plug-ins for more information.

Genesys Administrator is unaffected by Genesys Administrator Extension. It provides the interface to configure, monitor, and control the management environment.

The following subsections describe some of the features of the GAX interface.

Tenant Filtering

GAX comprises a set of modules that are selected and viewed in a browser interface. Each of the modules enables you to filter the information that you view about the applications that you have configured and deployed in the Genesys environment.

In a multi-tenant environment, GAX enables you to filter your views by a single tenant or by multiple tenants. By default, when you log in the view is of your default tenant. You can use the tenant selector to change the view so that you can view by one or more tenants.

Filtering and Sorting Lists and Tables

All lists and tables in the GAX interface can be sorted by clicking on the column headings. Tables and lists can also be filtered by appropriate criteria, for example:

- Tenant
- Date
- Date range
- Name
- Deployed by
- Deployed date

Field Auto-completion

All fields in the GAX interface that have predefined values support auto-completion. When you start to enter a value in the field, GAX searches for an existing value in the database and completes the entry. You can override auto-completion by continuing to enter the value. You can accept the auto-completion value by pressing `Enter`.

Localization

GAX supports the installation of multiple language packs for the user interface. You can choose to configure one default language across all GAX instances, while each user can select a different language. Default and user-specific language selection is done in the Preferences menu. See Preferences for more information.

You can install language packs by using the plug-in installation procedure. See Installing a GAX compatible plug-in by using the Software Import Wizard for more information, or refer to the Help pages by clicking the Help button in GAX (also available here).

Solution Deployment

Solution Deployment enables you to fully deploy Solution Definitions and Installation Packages (IPs) to remote locations. This includes installation and configuration of all of the necessary Applications and updates to existing multi-tenant Applications, where appropriate.

> **Note**: Genesys Deployment Agent (GDA) does not support multiple concurrent deployments on the same host. Therefore, multiple users cannot deploy a solution by using GAX on the same host at the same time that GDA is deploying. This limitation has always existed for GDA.

A Solution Definition consists of none, one, or multiple IPs for Genesys components. For Hosted Provider Edition, the IPs to be deployed must be primarily related to `Tenant` objects, and should contain object definitions, access permissions, and role privileges.

A Solution Definition consists of an XML file that defines the steps to install, upgrade, or configure IPs and system configurations to successfully deploy a solution. For information about authoring Solution Definition files, see the Authoring Solution Definitions page.

Solution Deployment can make changes to tenant objects in Configuration Server, perform installations of IPs, or execute external scripts, such as database scripts.

For each Deployed Solution, from the Deployed Solutions window you can export a file that contains the properties, summary, and actions for auditing purposes.

> **Note**: Not all browsers enable you to use filenames that are not US-ASCII compatible; therefore, Genesys recommends that you use only filenames that are US-ASCII compatible.

Defined Privileges

Roles and their privileges define what you can do in a given application. In Genesys Administrator Extension, roles and their privileges are controlled by the use of Role objects, which are assigned to Users (including Agents) and Access Groups.

Privileges are imported into GAX during the upload of an Installation Package (IP). All privileges that are defined in the metadata of the IP are imported into the GAX database. Privileges are defined as "task" elements in the metadata XML of the IP.

> **Note**: This functionality is only available in releases 8.1.3 and higher.

Solution Package Definition File Version Tracking

During normal use, Solution Package Definition files (also called SPDs or just Solution Definitions) are added, upgraded, revised, and removed. Solution Deployment supports versioning, auditing, and tracking of changes of SPDs from within the GAX interface. The tracking report can be exported to a

CSV file for use outside of GAX.

Solution Deployment enables you to view and access past versions of SPDs. You can also add custom comments and notes to any version.

You can filter and sort the SPD history by one or more of the following criteria:

- Solution—Group results by deployed solutions.
- Tenant—Group results by tenant and select a subset of a tenant or tenants by Solution and version.
- Date—Group results by date range.
- Result—Group by successful and failed deployments.

You can generate reports for both individual Solutions as well as for individual tenants.

You can configure the reports by specific criteria, including the following parameters:

- Solution Definition name
- Solution Definition version
- Tenant name
- Profile
- Date deployed
- Deployed by (name of the individual who performed the deployment)
- Result of deployment (Success, Fail, Unknown)
- Latest (true or false)
- Application name (IP Xref)

External Script Support

Solution Deployment passes arguments to external scripts when executing them, and can receive back results from the execution of a script. For example, if you have a script to create a new virtual host by using the VMware API, you can specify a name or naming convention from within an SPD. You will then receive confirmation that the creation was successful and the name of the new host that was created.

Operational Parameter Management

Operational Parameter Management enables the creation of parameters that can be used in parameterized Routing Strategies, in which the values of the parameters are defined at runtime and integrated into the call flow. In most cases, parameter creation and assignment proceeds as follows:

1. The Solution Provider defines the parameters by specifying the type of parameter and a name that can be referenced in a strategy.
2. The Solution Provider groups parameters into a Parameter Group Template. A parameter can be associated with one or more templates.
3. The Solution Provider deploys Parameter Group Templates to one or more Tenants.
4. The tenant administrator, or a user with the appropriate roles and permissions, then enters values for the parameters in the Parameter Group, enabling control of active strategies. Genesys Administrator Extension stores those values in the Configuration Database as part of a `Transaction` object.
5. The Universal Routing Server `Application` object (or any other interaction routing application such as GVP) executes a Routing Strategy to read those values and integrate them into the call flow. Orchestration Server and GVP Media Server `Application` objects are also supported.

Routing Strategies

In select cases, a Tenant may create its own Routing Strategy. The Solution Provider then grants the Tenant permission to define parameters and create the Group templates. Provide a tenant all the required privileges to create parameters, group templates, and deploy groups (refer to Genesys Administrator Extension Role Privileges).

Parameters

Operational Parameter Management can be used to update a parameter group after it has been deployed. You can add, remove, re-order, and modify parameters that have already been deployed to a parameter group. All modifications are tracked as part of the audit trail.

Objects and strategies can be associated with specific Parameter Group Templates to ensure that they are not deployed with the incorrect objects or strategies. Operational Parameter Management provides a view of all of the objects and strategies that are associated with a specific Parameter Group so that you know where the objects are used, including information about Tenant ownership and associated applications and scripts.

You can specify the application type or the specific application object for which the Parameter Group Template is compatible. If the type is set, it becomes a permanent attribute of the application. If there are multiple simple-routing-type routing scripts in the system, you can specify that only one matches the Parameter Group Template and is therefore compatible, rather than all scripts of a type.

When you create the Parameter Group Template, you can select an existing application of a particular type to associate the Parameter Group Template with the application. This ensures that the correct applications are deployed at deployment time.

GVP

Operational Parameter Management can be used to deploy parameters that can be used by Genesys Voice Platform (GVP) and other VXML applications. You can use Operational Parameter Management to deploy a set of parameters to create a new Configuration Layer object that is associated with a specified VXML application that is used by GVP.

Orchestration Applications

Operational Parameter Management can also be used to deploy parameters that can be used by Orchestration Applications (SCXML).

Audio Resource Management

Genesys Administrator Extension provides an interface for Audio Resource Management. This enables you to manage audio resources for both announcements and music files. This module also enables the conversion of audio files (`.wav` using PCM encoding), and the deployment of audio files to Media Servers throughout the network.

> **Note**: Audio Resource Management supports only WAV files that use PCM encoding. If you use non-PCM encoded files, there might be conversion artifacts, or the conversion might fail completely.

Generally, audio resources are handled as follows:

1. The Solution Provider maintains Audio Resources. Each Audio Resource contains one or more Audio Resource Files. Each Audio Resource File is associated with one Personality.

   In select cases, Tenants may also create their own Audio Resource Files, Personalities, and Audio Resources. To create Audio Resources as a tenant, provide the corresponding role privilege to the tenant user.

2. The Solution Provider deploys Audio Resources to Tenants.

   If a Tenant has created its own Audio Resource Files and Personalities, they can add them to the Audio Resources deployed by the Solution Provider.

3. A Routing Strategy that is executed by the Tenant selects an Audio Resource and a Personality. The Routing Strategy might use Operational Parameter Management to make the selection.

> **Note**: All Audio Resources are treated as *system* announcements — even when they are created by a tenant user. This means that a routing block must address the audio resource as resources of type `System Announcement`, not `Tenant Announcement`.

Audio Resource Management supports deployment of multiple audio resources to multiple tenants in a single step. You can select multiple audio resources from the tenant list and execute the deployment to several tenants simultaneously.

License Usage Reporting

Note: This section only applies to GAX 8.1.301 releases or lower. In GAX 8.1.310 releases or higher, License Usage Reporting functionality is provided by the License Reporting Manager (LRM) plug-in for GAX.

License Usage Reporting is a Graphical User Interface (GUI) application that is provided by Genesys Administrator Extension. It accesses data from a central License Reporting Manager (LRM) database to provide on-demand license utilization reporting to Solution Providers and tenant administrators. The central LRM database contains data that is collected from all local LRM databases.The data collection is done by the LRM system by using automated nightly processes. If the data is not collected, contact the LRM Administrator. See License Usage Reports Not Available.

The License Usage Reporting module also enables you to define provisioned counts for each tenant. Here you can select sellable items as well as the provisioned quantity and the start date.

LRM maintains a full history of Provisioned Counts in the system database in the `LRM_PROVISIONED_COUNT` table. This table keeps one Provisioned Count value for each unique triple: `TENANT_ID`, `SELLABLE_ITEM_TYPE`, `EFFECTIVE_DATE`. The `lrm-provisioned-counts` configuration layer object might have as many records as is required by the GAX application.

Account Management

Account Management enables the general management of configuration objects, such as:

- Users
- Agent Groups
- Skills
- Access Groups
- Roles
- Capacity Rules

In addition, Account Management enables you to create, edit, and manage these objects one at a time or in bulk.

Users

The `User Accounts` panel is a central location for creating, provisioning, and managing user accounts.

Users are the contact center personnel, including agents, who need access to Genesys applications. Agents are users who handle customer interactions directly.

Agent Groups

The `Agent Group` panel lists the agent groups in your environment.

An agent group is a logical grouping of agents. Agent groups are typically set up to provide particular sets of contact center services.

Skills

The `Agent Skills` panel provides a streamlined interface for the creation and management of agent skills.

Skills are qualities or abilities that agents possess and that affect the placement of each agent in a contact center hierarchy. Common skills include abilities in different languages, particular categories of product knowledge, or ability in particular types of sales.

Agents can be associated with a set of skills. For each skill, the agent is also given a skill level, or level of competency with this skill.

Access Groups

The `User Access Group` panel lists the user access groups in your environment.

Access groups are groups of users who have the same set of permissions for Configuration Database objects.

In many cases, users fall into a small number of categories with similar access needs. A team of agents all performing the same tasks often has identical access needs. Two or three people who are responsible for maintaining a specific site of the contact center might also have identical access needs. You can greatly simplify access control by adding individuals to access groups and then setting permissions for those groups.

Roles

The `Roles` panel lists the roles in your environment.

Roles define what you can do in a given application. In Genesys Administrator Extension, roles and their privileges are controlled by the use of role objects, which are assigned to users (including agents) and access groups.

Roles are application-specific, and must be defined for each application that supports them.

Role management allows GAX users to configure and distribute roles by tenant. Role management provides the following features:

- A single-click model for adding a role privilege to a role
- Define which role privileges can be modified by tenant administrator users, enabling tenant administrators to manage their user accounts and create new roles as necessary

The privileges available to each Role are determined by the settings in the Solution Deployment module.

A default role is created during the installation package (IP) process. Typically, the default role is an administrator or super user. The default role contains a user name and a list of privileges. You can recreate the default role if a required role is unavailable.

Capacity Rules

The `Capacity Rules` panel enables you to set capacity rules for various operations in your environment. For example, you might choose to set capacity rules for the number of voice interactions or e-mail interactions, or a combination of both, that can be processed at one time.

Auditing

The auditing feature writes data to Message Server about activities in Operational Parameter Management and Audio Resource Management, and Message Server writes the data to the Genesys Log database. Auditing data is made available to the GAX user by selecting the `History` option in the `Related` menu in the panel of certain items in the GAX user interface. The auditing feature reads the information from the Log database and enables you to view the change history of objects such as Personalities and Parameter Groups.

Architecture

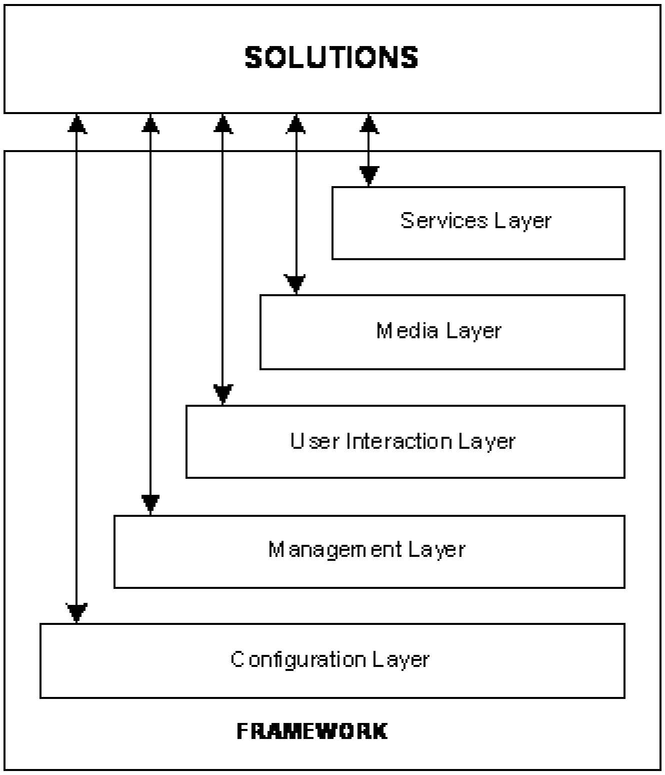
This section describes the architecture of Genesys Administrator Extension as it resides in the User Interface Layer of the Genesys Framework, and the architecture and connections within a Genesys Administrator Extension configuration.

User Interface Layer

Genesys Administrator Extension resides in the User Interaction Layer of the Genesys Framework. This Layer provides comprehensive user interfaces to:

- Configure, monitor, and control the management environment.
- Perform specific tasks related to Solution Deployment, Operational Parameter Management, Audio Resource Management, Account Management, and License Usage Reporting (in GAX 8.1.301 releases or lower).

The figure below illustrates how the User Interaction Layer is positioned within the Framework architecture.



Framework Architecture

Refer to the Framework 8.1 Deployment Guide or Framework 8.0 Architecture Help for more information about Framework architecture as a whole.

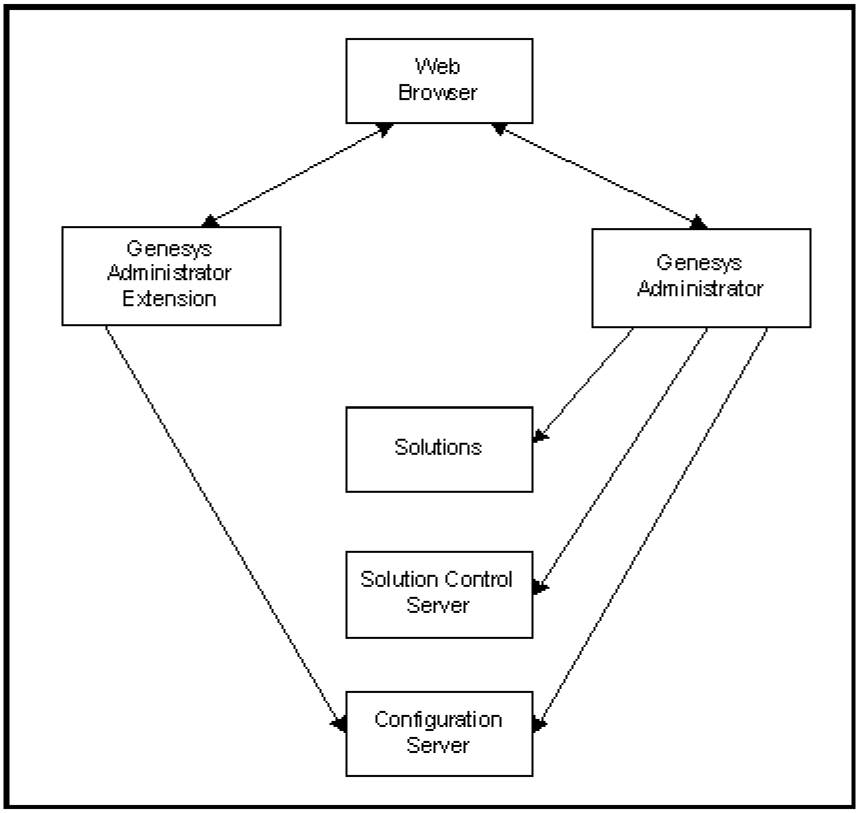
Functions

The User Interaction Layer provides centralized web-based functionality and interfaces for the following:

- Remote deployment of Genesys components by using the Genesys Deployment Agent (a Management Layer component).
- Configuration, monitoring, and control of applications and solutions.

Architecture

The figure below shows a more detailed diagram of the architecture of Genesys Administrator Extension.



User Interaction Layer Architecture

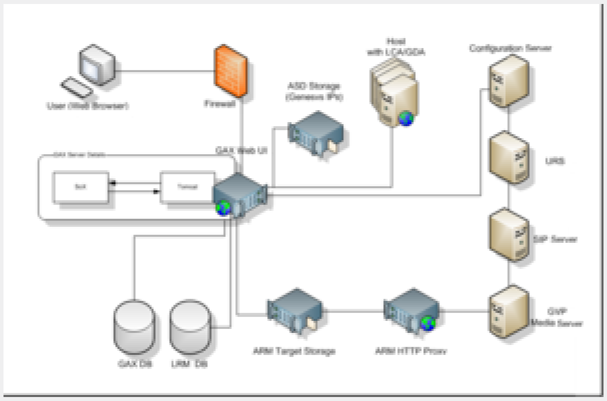
- The browser-based Genesys Administrator Extension includes a comprehensive user interface to perform tasks that are related to Solution Deployment, Operational Parameter Management, Audio Resource Management, Account Management, and License Usage Reporting (for GAX 8.1.301 releases or lower).
- Currently, Genesys Administrator and Genesys Administrator Extension are the only components in the User Interaction Layer.

- Genesys Administrator Extension:
    - Communicates with the Configuration Server (a Configuration Layer component) to exchange configuration data.
    - Uses the GAX Database to store non-configuration information, such as operational parameter templates and audio resource metadata.
    - Reads license utilization information from the LRM Database to generate License Usage reports (in GAX 8.1.301 releases or lower).
    - Uses Sound eXchange (SoX) to encode audio files.
    - Sends encoded audio files to the Audio Resource Manager (ARM) Storage. From the ARM storage, the ARM Web Server distributes them to GVP Media Servers.
    - Reads the Genesys IPs in Solution Deployment (ASD) storage to remotely deploy solutions to Hosts on which Local Control Agent is installed, and the Genesys Deployment Agent is running.
- Genesys Administrator:
    - Communicates with the Configuration Server (a Configuration Layer component) to exchange configuration information.
    - Communicates with the Solution Control Server (a Management Layer component) to exchange status, operations, and control information.
    - Reads logs from the Centralized Log Database (a Management Layer component).
    - Provides the web services for Genesys Administrator Extension.
    - Uploads IPs to ASD storage for use by Genesys Administrator Extension.
    - Depending on the solutions that are deployed in the system, Genesys Administrator and Genesys Administrator Extension might also communicate with other back-end servers to retrieve solution-specific information.

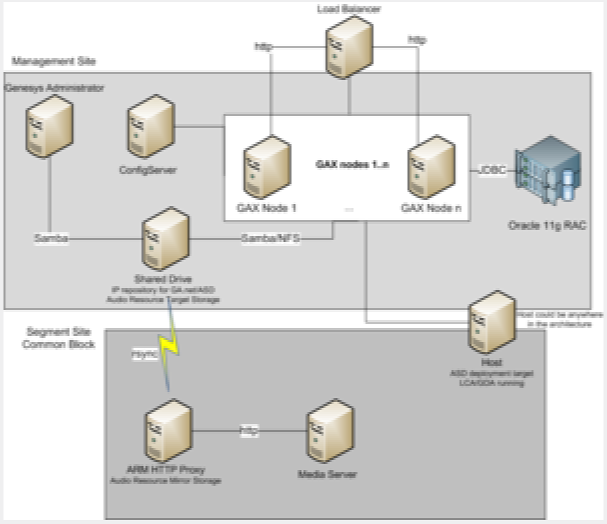
	Note: Both TCP/IP v4 and TCP/IP v6 communications are supported between GAX and other Genesys components.

Configurations

Genesys Administrator Extension can be deployed as a single instance or in a load-balanced environment. The left figure below shows how Genesys Administrator Extension connects with its modular components. The right figure below shows the connections that Genesys Administrator Extension makes to other components in a load-balanced environment. When deployed in a load-balanced environment, Genesys Administrator Extension is located in the Management Site.



Genesys Administrator Extension Architecture



Genesys Administrator Extension Architecture in a Load-Balanced Hosted Provider Edition environment

Database Size Requirements

To help you plan to manage your space requirements for audio resources, this section provides information about space allocation for a 100-tenant system with an average of 100 announcement files per segment, including personalities.

Original Audio Resource Files

The space required for the original audio resource files that are uploaded by tenants can be calculated as:

```
Original Files Storage Requirements = <# of tenants> x <avg # of announcement files>
x <avg file size>
```

For example, if you have 100 tenants with 100 audio files of an average size of 3 MB you would have to calculate 30 GB of space for just the original audio files:

```
Original Files Storage Requirements =
```

100 x 100 x 3 MB =

30,000 MB = 30 GB

Processed Audio Resource Files

The original files are stored both in the database and on the disk (unless database storage is turned off by using the configuration options). The processed files are located only on the disk. Therefore, the raw storage that is required on the disk can be calculated as:

Processed Files Storage Requirements = ((<# of tenants> x <# of announcement files> x <avg file size>) / <compression factor>) x (<# of conversion formats>)

In the example with 100 tenants, the requirement for Processed files is also 30 GB:

Processed Files Storage Requirements =

((100 x 100 x 3 MB) / 3) x (3) =

30,000 MB = 30 GB

Reserved Space

For the database, which holds only the original files, additional space should be reserved to allow for short time peaks and better database performance. Genesys recommends that 50% (1.5 times) of additional space should be reserved for this purpose:

```
Database Size Requirements = <Original Files Storage Requirements> x <reserve
percentage>
```

In this example, the suggested database space requirement is:

```
Database Size Requirements =
```

30 GB x 1.5 = 45 GB

Your disk space requirement should also include reserved space to prevent degraded performance, which can occur if drives become too full.

Genesys recommends that the reserved space allocation is 25% (1.25) of the actual raw requirements:

```
Disk Size Requirements = (<Original Files Storage Requirements> + <Processed Files
Storage Requirements>) x <reserve percentage>
```

Therefore, in total, for the original files, the converted files, and reserved space, 75 GB are required:

```
Disk Size Requirements =
```

(30 GB + 3 0 GB) x 1.25 = 75 GB

Setting Up Genesys Administrator Extension

This chapter describes how to install and configure Genesys Administrator Extension. It also describes the prerequisites and other information for setting up Genesys Administrator Extension to perform the tasks that are described in the Overview chapter.

This chapter contains the following sections:

- Overview
- Deploying Genesys Administrator Extension
- Prerequisites for Genesys Administrator Extension Modules
- Configuring System Security
- Configuring the Auditing Feature
- Managing GAX Compatible Plug-ins
- Upgrading GAX
- Customizing the GAX Homepage
- Cleaning the GAX Database After a Tenant is Deleted

Overview

Genesys Administrator Extension is deployed on a web application server, and can be accessed by using a web browser. It does not have to be deployed in the same environment with Genesys Administrator, and nothing needs to be installed on client machines.

Note: GAX is normally deployed in a multiple tenant environment; however, single-tenant environment deployment is supported as of version 8.1.2. If you deploy GAX in a single-tenant environment, the Tenant Management features and filtering are not applicable.

Prerequisites

Before you deploy Genesys Administrator Extension, you should review the planning information in the Framework 8.1 Deployment Guide. This will help you to deploy Genesys Administrator and other components of the Framework in a manner that is most appropriate to your situation.

Genesys Administrator Extension requires Management Framework. To use the Role-based Access Control feature, Configuration Server 8.1.x is required.

Note: A new application type, `Genesys Administrator Server`, was introduced in Genesys Framework release 8.1.1 for use with Genesys Administrator Extension release 8.1.2 or higher. Previous versions of GAX do not support this new application type and must use the `Genesys Generic Server` application type.

Note: To avoid issues with role assignments, you should upgrade the application, metadata, and the roles to the new type when you migrate to GAX 8.1.2 or perform a fresh install (see Upgrading to the latest Genesys Administrator Extension for Management Framework 8.1.1 or higher)

The computer on which you install Genesys Administrator Extension must be capable of acting as a web application server, and must be running the following:

- Red Hat Enterprise Linux 5.5 (64-bit) - Enterprise Edition, with `Updates from RHN` enabled;

Or,

Windows Server 2008 R2, with 64-bit applications running natively on a 64-bit OS.

- Java 6 Runtime (JRE) from Oracle. See the Setting up the host for Genesys Administrator Extension server for information about obtaining and installing Java, if necessary.
- Tomcat 6.0.37 from Apache. When setting up Tomcat, Genesys strongly recommends that you enable gzip compression for responses. Follow the procedure Installing Tomcat.

In addition, each module of Genesys Administrator Extension might have additional prerequisites. Refer to Prerequisites for Genesys Administrator Extension Modules for more information.

Browser Requirements

Genesys Administrator Extension includes a web-based GUI with which you can manage Genesys applications and solutions. It is compatible with the following browsers:

- Microsoft Internet Explorer 8.x or 9.x
- Mozilla Firefox 5 or higher
- Safari 5 or higher
- Chrome 8 or higher

Note: Genesys Administrator Extension supports all major browsers, but it is optimized for Chrome.

Genesys Administrator Extension is designed to be viewed at a minimum screen resolution of 1024x768, although higher resolutions are recommended. If you are working in 1024x768 mode, maximize your browser to ensure that you can see all of the interface. In addition, all windows of the browser must be set to a resolution of 1024x768 or greater.

Note: If the download of Audio Resource Files, Encoded Files, and other GAX downloads are blocked by the Microsoft Internet Explorer 8 or 9 information bar and, after you confirm the download, you are redirected to the main page and you must repeat the download request, you can adjust your browser settings to prevent this scenario (Browser Issues).

Required Permissions and Role Privileges

Genesys Administrator Extension uses a permission-based mechanism and a role-based access control system to protect your data. Before installing and using Genesys Administrator Extension, ensure that all users have the necessary access permissions and role privileges to do their work. The following are examples of scenarios that require permissions:

- A tenant user must have write (`Update`) permission on his or her `User` object to set and save his or her user preferences in Genesys Administrator Extension.
- To log in to Genesys Administrator Extension, a user must have Read permission on his or her `User` object, Read and `Execute` permissions on his or her `Tenant` object, and Read and `Execute` permissions on the Genesys Administrator Extension client `Application` object. These permissions are usually assigned by adding the users to access groups.

There are no role privileges required to log in to GAX. However, GAX-specific functions might require additional role privileges to be enabled. Refer to Role Privileges for more information about role privileges.

Deploying Multiple Instances of GAX with Shared Resources

You can install multiple instances of GAX to support both High Availability (HA) and load balancing. You can also install multiple instances of GAX to take advantage of the GAX plug-in architecture. Each instance of GAX can be deployed with a different combination of plug-ins.

In either scenario, the multiple instances of GAX share the same data resources, such as Configuration Server, the GAX database, and audio resources, but are executed independently by different users on different hosts.

Minimum Required Firewall Permissions and Settings for GAX Deployment

Your firewall must allow incoming connections on the Tomcat http and https ports. (for example 8080, 80, 433, and so on, based on your setup). Tomcat can listen on more than one port at once.

You must allow outgoing connections to allow GAX to establish connections; however, you can restrict the connections to networks that contain the following components:

- GDA hosts
- Databases
- Genesys configuration layer servers: Configuration Server, Message Server, and Solution Control Server

Minimum Required File System Permissions and Settings for GAX Deployment

The GAX operating system user is the user that runs the GAX process. The GAX operating system user must be the owner of the Tomcat folder and have the following permissions:

- Write permission on the log file folder
- Read/write access to the folder configured for ARM

Note: If Tomcat was extracted from the `.tar` file, the operating system user would already have these permissions.

Deploying Genesys Administrator Extension

The following table summarizes the steps necessary to perform the basic deployment of Genesys Administrator Extension. Before beginning your installation, make sure that you have met the prerequisites listed in Prerequisites. If you plan to install any of the modules in Genesys Administrator Extension, refer to Prerequisites for Genesys Administrator Extension Modules before using them.

> **Important**
>
> Unless specified otherwise, all commands that are entered on a command-line in this section should be issued as a root user (command prompt of #) or as a regular user (command prompt of $).

Deploying Genesys Administrator Extension

1. Create and configure the configuration objects required for Genesys Administrator Extension.

Creating the necessary configuration objects for Genesys Administrator Extension

Purpose

To create the following configuration objects required by Genesys Administrator Extension:

- `Host` object for the computer on which Genesys Administrator Extension is to be installed (Step 1)
- `Host` object for the computer on which the database to be used by Genesys Administrator Extension will be located (Step 2)
- `Database Access Point` to provide database access to the database used by Genesys Administrator Extension (Step 3)
- `Application` object for Genesys Administrator Extension with a connection to Configuration Layer to retrieve configuration information (Step 4)
- `Application` object to provide the capability to log in to Genesys Administrator Extension (Step 5)

> Important
>
> - All tasks in this procedure are completed by using Genesys Administrator or a similar configuration utility to create the necessary configuration objects in the Configuration Database. This procedure assumes that you are using Genesys Administrator.
> - In this procedure, use the instructions that are provided in Genesys Administrator 8.1 Help or the Framework 8.1 Deployment Guide, and add the object-specific configuration requirements listed here.

Prerequisites

- Management Framework 8.0.0 or higher is installed and running. You must have Configuration Server 8.0.300.42 or higher.
- If you are using Configuration Server 8.1.1 or higher, you must use Genesys Administrator 8.1.2 or higher, as previous versions do not support the GAX application type for configuring role privileges.
- Genesys Administrator 8.1 or higher is installed and running.

Start

1. Create and configure a `Host` object for the computer on which Genesys Administrator Extension will be installed, as follows:
   1. Use the instructions in Genesys Administrator 8.1 Help or the Framework 8.1 Deployment Guide to create and configure a `Host` object.
   2. On the `Configuration` tab, specify a Solution Control Server `Application` object.
   3. Click `Save & Close` to save the new object and its configuration.

4. Use the instructions in Genesys Administrator 8.1 Help or the Framework 8.1 Deployment Guide to create and configure a `Host` object for the computer on which the database to be used by Genesys Administrator Extension will be installed.

> Important
>
> When using Genesys Administrator in a load-balanced environment, make sure that all nodes have shared-access to the application metadata. See the Framework 8.1 Genesys Administrator Deployment Guide for details about how to set this up.

5. Use Genesys Administrator to create and configure a Database Access Point (DAP) `Application` object, which is necessary for connectivity to the database that will be used by Genesys Administrator

Extension, as follows:

a. Use the instructions in the Framework 8.1 Deployment Guide to create and configure a DAP `Application` object.

b. Open the `Configuration` tab.

c. In the `Server Info` section, enter the following information:

   i. In the `Tenants` list, add the Environment Tenant.

   ii. In the `Host` field, select the `Host` object on which the database is to be installed, and that was configured in Step 2. If you do not use a non-standard port, enter 1521 for an Oracle database, 1433 for a Microsoft SQL Server 2008 database, or 5432 for a PostgreSQL database.

c. In the `DB Info` section, enter the following:

   i. In the `Connection Type` field, select `JDBC`.

   ii. In the `Role` field, select `Main`.

   iii. In the `Debug` field, select `false`.

   iv. In the `JDBC Query Timeout` field, enter 15.

   v. In the `DBMS Type` field, select `Oracle` for an Oracle database, `mssql` for a Microsoft SQL Server 2008 database, or `postgre` for a PostgreSQL database.

   vi. In the `Database Name` field, enter the Solution name of the database instance.

   vii. In the `User Name` field, enter the user name required to access the database.

   viii. In the `User Password` field, enter the password required for the user name specified in the previous step to access the database.

   ix. In the `Case Conversion` field, select `any`.

j. Open the `Options` tab and complete the following steps:

   i. Create a new section called `GAX`.

   ii. In this new section, add the configuration option `role` and set its value to `main`. This identifies this DAP as the one for the main database that is used by Genesys Administrator Extension.

c. Click `Save & Close` to save the new object and its configuration.

4. Create and configure a Server `Application` object for Genesys Administrator Extension, as follows:

   a. Import the `Application Template` object for Genesys Administrator Extension. Refer to Genesys Administrator 8.1 Help for detailed instructions.

      i. Upload one of the following files from the installation package, depending on which version of Management Framework you are running:

   - For Configuration Server up to version 8.1.0: `Genesys_Administrator_Extension_MF810_813.apd`
   - For Configuration Server from version 8.1.1 on: `Genesys_Administrator_Extension_813.apd`

- Import the XML metadata file, which contains the GAX privilege information and default settings, by clicking `Import Metadata`, then navigate to the folder in which the application template was deployed. There are two templates available, depending on which version of Management Framework you are running:

   - For Configuration Server up to version 8.1.0: `Genesys_Administrator_Extension_MF810_813.xml`

  - For Configuration Server from version 8.1.1 on: `Genesys_Administrator_Extension_813.xml`

- Click `Save & Close` to save the new object.
- Use the instructions in the Framework 8.1 Deployment Guide to create and configure an `Application` object by using the template imported in the previous step and on the `Host` object configured in Step 1. This new object will appear as being of type `Generic Genesys Server` if you are running Management Framework <=8.1.0 and of type `Genesys Administrator Server` if you are running Management Framework >8.1.0.
- Open the `Configuration` tab.
- In the `General` section, in the list of `Connections`, add connections to the following components:
  - Primary Solution Control Server
  - Main DAP (configured in Step 3)
  - Auditing DAP. This should be linked to the database where the auditing data will be written. The configuration (refer to Step 3) is the same as the Main DAP; however, the Role property of the Auditing DAP should be set to the value `auditing` instead of the value `main`.

> **Important**
>
> Both the Auditing DAP and the LRM DAP are not mandatory for every installation. If you configure GAX to use auditing, then you must have a DAP configured. If you remove the LUR from the installation, the DAP is not required.

  - LRM DAP. This should be linked to the database that will hold the LRM data that is displayed by License Usage Reporting. The configuration (refer to Step 3) is the same as the Main DAP; however, the Role property of the LRM DAP should be set to the value `lrm` instead of the value `main`.

> **Important**
>
> In GAX 8.1.310 releases or higher, License Usage Reporting functionality is provided by the License Reporting Manager (LRM) plug-in for GAX.

- In the `Server Info` section, enter the following information:
  i. In the `Working Directory` field, enter the path to your working directory.
     - (Linux) For example: `/home/gcti/apache-tomcat-6.0.37/bin/`.
     - (Windows Server) For example: `C:\GCTI\Tomcat6_GAX_812\bin`
  ii. In the `Command Line` field, enter the following:
     - Linux:`./gax_startup.sh`
     - Windows Server:`.\gax_startup.bat`

iii. In the Command Line Arguments field, enter the following (all on one line):

```
-host <Configuration Server name or IP address> -port <Configuration Server port> -app <GAX
 Generic Server Application name>
```

where the values in `<brackets>` are replaced by the values used by your deployment.

> ### Important
>
> **Limitation:**
>
> If Configuration Server has several independent ports configured, the port that GAX should use cannot be freely chosen if GAX is started by Management Framework tools such as Solution Control Server, Genesys Administrator, or Solution Control Interface. In that case GAX will always connect to the port that Solution Control Server uses to connect to Configuration Server.
>
> **Workaround:**
>
> If GAX should not use the same Configuration Server port as Solution Control Server, GAX should not be started by using Management Framework tools. GAX should only be started manually or as a service.

- Select the host object where GAX is to be deployed.
- Specify the listening port by entering `8080` (the typical value for Genesys; you can also specify another port) in the `Listening Port` field.

> ### Important
>
> Setting this port value does not change the port that is used by GAX; it is overridden by the Tomcat configuration.

- On the `Options` tab, verify or update the name of your client object (to be created in Step 5) given by the following option:

```
general.client_app_name=<name>
```

- Click `Save & Close` to save the new object and its configuration.

> ### Important
>
> The creation of a client is optional. The default client will be used in a standard installation, such as cases in which

> general.client_app_name is set to default. Perform the next step only if you need to allow access to GAX for users that should not be able to access Genesys Administrator.

- Create and configure an Application object to allow users to log in to Genesys Administrator Extension. The name of this object must be exactly the same as that specified in Step 4h above. All users must have Read and Execute permissions for this Application object.

  Use the instructions in the Framework 8.1 Deployment Guide to create and configure an Application object of type Configuration Manager based on the Configuration Manager Application Template.

  This object acts as a client application for the Genesys Administrator Extension server.

- Configure GAX logging by using the Genesys Log Wizard from Genesys Administrator or from the Genesys Solution Control Interface. The Log Wizard creates a set of configuration options in the log section of the GAX Server application object.

  (Optional) You can also create the log options manually by using the values in the table below.

GAX Logging value

Option	Description	Value	Required	Default
all	Defines the types of logging to be executed as a comma-separated list	stdout, <filename>	Yes	stdout
verbose	Defines the log level	all, trace, interaction, standard, none	No	standard
segment	Defines the maximum file size for file logging	<file size in KB>	No	""
expire	Number of backup log files to be maintained	<number of files >	No	""

- Set up a user on the host to create a new user named gcti and a group named gcti, which is the primary group for the new user and set /bin/bash as the default shell. This user will be used to run the Tomcat service and to run LCA (unless you have configured LCA to run under the root or another user). Refer to the Genesys Administrator 8.1 Help for information about creating a new group and a new user.

End

2. Set up the user on the Host machine.

Refer to the Genesys Administrator 8.1 Help for information about creating a new group and a new user.

3. Set up the host on which Genesys Administrator Extension server will run.

Setting up the host for Genesys Administrator Extension server

Purpose: To set up Oracle Java Server JRE (Java Runtime Environment) version 6 or 7. (**Note**: GAX only supports the 64-bit version of Oracle Java HotSpot Server VM.)

Start

1. If Java JRE 6 is not already installed on the host machine where Genesys Administrator Extension will be installed, install it now as follows:
    a. Download the Oracle Java Runtime Environment Kit (JRE) from the following website:

        http://www.oracle.com/technetwork/java/javase/downloads/index.html

        (Linux) Select the `.bin` package that does not have `rpm` in its name. That is, select `*.bin`. Do *not* select `*-rpm.bin`.

    b. (Linux) Put the downloaded file into the directory `/usr/lib/java`.
        i. Make the file executable by entering the following command:

        ```
        chmod +x <filename>.bin
        ```

        ii. Run the file to install Java, by entering the following command at the # prompt:

        ```
        ./<filename>.bin
        ```

        The contents will be installed in the same directory as the file.

    c. (Windows) Double click the Java installer. The contents will be installed in the directory that you specify during the installation.

3. Set the following environment variables for your host, as follows:
    a. (Linux) Insert the following lines into the `/etc/profile` file:

        ```
        export JRE_HOME=/usr/lib/java/jre-<version of Java downloaded>/jre
        ```

        Log out and log in again to activate the new environment variables in the current session.

    b. (Windows) Create a new System Variable named JRE_HOME and use the path that was used during installation as the value (for example, `C:\Programs\Java\jre1.6.0_23`). To do this, right-click your Computer icon. Select `Properties > Advanced System Settings > Environment Variables`, and then create the JRE_HOME variable.

3. Install Local Control Agent on this host. For detailed instructions, refer to the Framework 8.1 Deployment Guide.

End

4. Install Tomcat.

Installing Tomcat

Prerequisites

- JRE 6 is installed on your host and JRE_HOME is configured correctly.

Start

1. Download Tomcat 6.0.37 as a ZIP archive from the following location:

   ```
   http://archive.apache.org/dist/tomcat/tomcat-6/v6.0.37/
   ```

   > **Important**
   >
   > You must use the ZIP archive to install Tomcat. GAX does not work properly if the Windows installer is used to install Tomcat.

2. (Linux) Open a terminal as the user `gcti` by entering the following command at the # prompt:

   ```
   su gcti
   ```

3. Extract the downloaded archive to the following directory:

- (Linux) `home/gcti/apache-tomcat-6.0.37`
- (Windows Server) `C:\GCTI\Apache Tomcat 6.0.37\`

> **Important**
>
> Ensure that the user on the host on which GAX Tomcat is running can read all files and execute all `*.bat` scripts (on Windows) or `*.sh` scripts (on Linux) in this directory.

- Set the environment variable:

- (Linux)
  1. Set the following environment variable for your host by inserting the following line into the `/etc/profile` file:

     ```
     CATALINA_HOME=/home/gcti/apache-tomcat-6.0.37
     ```

- (Windows)
  1. Navigate to Control Panel.
  2. Double-click on `System`.
  3. Click `Advanced system settings`.
  4. Click `Environment Variables`.
  5. In the `System variables` section, set the system environment variable `CATALINA_HOME` to the path where your Tomcat instance is installed.

> Important
>
> `$CATALINA_HOME` refers to the installation directory for Tomcat.
>
> On Windows, it might be: `c:\Program Files\Apache Tomcat 6.0.37\`.
>
> On Linux, it might be: `/home/gcti/apache-tomcat-6.0.37/`

- In the file `/home/gcti/apache-tomcat-6.0.37/conf/tomcat-users.xml`, add the following line in the section `<tomcat-users>`:

  ```
  <user username="manager" password="<password>" roles="manager"/>
  ```

  where `<password>` is the password to access Tomcat.

- To test the installation:
  - Linux:
  1. Enter the following command at the # prompt:

     ```
     bin/startup.sh
     ```

  2. Point your web browser to `http://<host>:8080`
  - Windows Server:
  1. Run the following file:

     ```
     C:\GCTI\Apache Tomcat 6.0.37\bin\startup.bat
     ```

  2. Point your web browser to `http://<host>:8080`

- (Optional) (Linux) Since Tomcat log files do not rotate by default, set up Tomcat log file rotation by creating the file `/etc/logrotate.d/tomcat` as the root, containing the following lines:

```
/home/gcti/apache-tomcat-6.0.37/logs/catalina.out {
```

```
copytruncate
daily
rotate 7
compress
missingok
size 256M
}
```

This also adds the file `catalina.out` to the system log daily rotation.

GAX does not write a lot of data to the catalina.out log file; however, it is good practice to enable log rotation to prevent long-term issues with the size of the log file.

> Important
>
> If you want detailed information about how to configure TLS/SSL for Tomcat connections, refer to the following website: http://tomcat.apache.org/tomcat-6.0-doc/ssl-howto.html

End

(Optional) Configure Tomcat to enable HTTPS-Only mode.

Configuring Tomcat to enable HTTPS-Only mode

Purpose

- To set up Tomcat to work in HTTPS-only mode and thereby improve system security.

There are three general steps that are required to set up Tomcat to operate in HTTPS-only mode:

1. Generate the keystore file. This file contains the certificate. A certificate is required. The certificate can be either a certificate authorized by a Certificate Authority, which you have to import into your keystore file, or, a self-signed certificate that you create and use in the keystore file.

   The last approach is simpler in that it offers the same level of security; however, the end user must trust your GAX authority.

   No matter which authority is used, all information is encoded and transmitted by using the HTTPS protocol.

2. Configure the SSL connector in Tomcat.
3. Configure Tomcat to use HTTPS-only for the GAX application.

More detailed information about HTTPS setup, certificates and authorization is available in the Tomcat

6 SSL documentation:

http://tomcat.apache.org/tomcat-6.0-doc/ssl-howto.html#Configuration

Prerequisites

- JRE 6 and Tomcat 6.0.xx are installed on the web host.

Start

1. Create a self-signed certificate, or import an existing certificate.

- To create and use a self-signed certificate, follow these steps:

i. To create a keystore, use the Keytool application (this is automatically installed with the JRE). Alternatively, you can use the OpenSSL application, as described in the Tomcat documentation.

ii. From the command line, navigate to the `JRE_HOME\bin` directory:

```
cd $JRE_HOME/bin/
```

iii. Execute the following, replacing all values of the placeholders, by using the same value for `<password_for_certificate>` and `<password_for_keystore_file>`.

```
keytool -genkeypair -alias tomcat -keypass
<password_for_certificate> -keystore
<keystore_file_location> -storepass
<password_for_keystore_file> -validity
<number_of_days_for_validity>
```

For example:

```
keytool -genkeypair -alias tomcat -keypass Genesys -keystore
/home/gcti/keystore.key -storepass genesys -validity 365
```

iv. At the prompts, provide the requested information about your certificate, including company, contact name, and so on. This information is displayed to users who attempt to access a secure page in your application.

> Important
>
> Enter the fully qualified domain name of the machine where your instance of Tomcat is running, as you are prompted for your first and last name.

The keystore file is created and it can be referenced by Tomcat.

- To import an existing certificate that is based on a certification request, follow these steps:

i. Create a local Certificate (refer to the previous bullet).

> Important
>
> In some cases you will have to enter the domain of your website (for example: `www.<example>.com`) in the `first- and lastname` field to create a working Certificate.

ii. Use the following command to create a Certification Request (CSR):

```
keytool -certreq -keyalg RSA -alias tomcat -file certreq.csr
 -keystore <your_keystore_filename> -keypass
 <password_for_certificate> -storepass
 <password_for_keystore_file>
```

The `certreq.csr` file is created. You can submit this certification request to a Certification Authority to obtain a certificate.

After you obtain the certificate signed by a Certification Authority, you can import it into your local keystore. Before you can import your certificate, you must import a Chain Certificate or a Root Certificate into your keystore.

iii. Use the following command to import the Chain Certificate into your keystore:

```
keytool -import -alias root -keystore
 <your_keystore_filename> -trustcacerts -file
 <filename_of_the_chain_certificate>
```

iv. Use the following command to import your Certificate:

```
keytool -import -alias tomcat -keystore
 <your_keystore_filename> -file
 <your_certificate_filename>
```

- Configure the SSL connector in Tomcat.

a. From the `conf` directory of your Tomcat installation, open the `server.xml` file in a text editor. If set, you can use the `CATALINA_HOME` variable:

```
$CATALINA_HOME/conf/server.xml
```

b. Find in the file the following code for setting the SSL connector:

```
<!--
    <Connector port="8443" protocol="HTTP/1.1" SSLEnabled="true"
               maxThreads="150" scheme="https" secure="true"
               clientAuth="false" sslProtocol="TLS" />
  -->
```

c. Uncomment this code and add the following line of code:

```
keystoreFile="<keystore_file_location>"
 keystorePass="<password_for_keystore_file>"
```

d. Replace the placeholders with the correct values from Step 1. For example:

```
<Connector port="8443" protocol="HTTP/1.1" SSLEnabled="true"
 maxThreads="150" scheme="https" secure="true"
```

```
keystoreFile="/home/gcti/keystore.key" keystorePass="genesys"
clientAuth="false" sslProtocol="TLS" />
```

e. Restart your Tomcat server, and verify that is it listening on port 8443 (or whatever port you have specified) in HTTPS mode. The HTTP port 8080 remains open. Connections to that port are not redirected to the HTTPS port 8443.

- Configure Tomcat to accept HTTPS only:

a. From the same directory as the `server.xml` file in Step 2, open the Tomcat `web.xml` file in text editor.

b. Above the <web-app> tag, add the following code:

```
<display-name>Security Constraint</display-name>
   <web-resource-collection>
      <web-resource-name>Protected Area</web-resource-name>
      <!-- Define the context-relative URL(s) to be protected
-->
      <url-pattern>/*</url-pattern>
      <!-- If you list http methods, only those methods are
protected -->
   </web-resource-collection>
<user-data-constraint>
   <transport-guarantee>CONFIDENTIAL</transport-guarantee>
</user-data-constraint>
</security-constraint>
```

c. Restart Tomcat.

d. Confirm that HTTP requests are redirected to your HTTPS port by attempting to access the GAX application over HTTP by entering the following URL in a web browser window:

```
http://<yourhostname>:8080/gax
```

The browser should be redirected to your HTTPS connection:

```
https://<yourhostname>:8443/gax
```

End

5a. Install Genesys Administrator Extension server on a Linux host.

Installing Genesys Administrator Extension server on a Linux host

Prerequisites

- The `Application` object for Genesys Administrator Extension server exists (see Step 4 of Creating the necessary configuration objects for Genesys Administrator Extension).
- The environment variable for JRE_HOME has been configured (see Step 2 of Setting up the host for

Genesys Administrator Extension server).

Start

1. Copy the IP to the host machine.
2. Navigate to the folder to which you copied the IP, and change the permissions of the installation file by entering the following command:

   ```
   chmod 755 install.sh
   ```

3. Run the installation file to extract and copy the necessary files by entering the following command:

   ```
   ./install.sh
   ```

> Important
>
> When you install Genesys Administrator Extension, you might receive the following error message that indicates that installation was unsuccessful:
>
> ```
> Unable to find configuration information. Either you have not used
> configuration wizards and the GCTISetup.ini file was not created or the
> file is corrupted.
> ```
>
> Ignore this message; Genesys Administrator Extension was installed successfully.

4. Enter information as prompted by the installation file, as follows:
   a. Enter the name of this host machine, or press Enter to select the default.
   b. Enter the name of the host where Configuration Server is installed.
   c. Enter the port number used by Configuration Server.
   d. Enter the username and password used to access Configuration Server.
   e. Select n to not use Client Side Port Option (the listening port of the application; refer to Genesys 8.0 Security Deployment Guide).
   f. When prompted to select which application to install, enter the number associated with the Genesys Administrator Extension server object.

      The following prompt is displayed:

      ```
      "Press ENTER to confirm /opt/genesys/gax as the destination directory or enter a new one =>'
      ```

      You can specify the GAX_HOME folder here:

      ```
      GAX_HOME=/home/gcti/gax
      ```

      By default, the installation puts a `startup` and a `setenv` script in the Tomcat `bin/` directory and the Genesys Administrator Extension application in the Tomcat `webapps/` directory. Additional resources and the database creation script are installed in the folder given by the GAX_HOME environment variable (see Setting up the host for Genesys Administrator Extension server). Press y to accept this, or press n to cancel setup.
   g. Make the `setenv` script executable by entering the following command at the # prompt:

```
chmod 755 setenv.sh
```

Important

- The GAX installer creates a `setenv.sh` file that enables you to adjust the memory settings for GAX. The `setenv.sh` file defines the memory (RAM) settings for GAX to 1024 MB. You can change the memory setting in the `setnev.sh` file to a different value. If you enable TLS encryption, ensure that you make the following updates to the `setenv.sh` file. The `setenv.sh` file contains the following lines:

```
# Uncomment the following lines only if you are going to use TLS. Don't forget to set the
correct path and password.
#export JAVA_OPTS="$JAVA_OPTS -Djavax.net.ssl.trustStore=/path_to_jre/jre6/lib/security/
cacerts"
#export JAVA_OPTS="$JAVA_OPTS -Djavax.net.ssl.trustStorePassword=secret_password"

# This line defines the memory (RAM) settings for Tomcat. If you have more RAM available
for Tomcat, adjust both values accordingly
export JAVA_OPTS="$JAVA_OPTS -Xms1024m -Xmx1024M"

# Uncomment following line to activate psdk.logs, it's recommended to let this option
deactivated
#export JAVA_OPTS=%JAVA_OPTS%
-Dcom.genesyslab.platform.commons.log.loggerFactory=com.genesyslab.platform.commons.log.Log4JLoggerFactoryI
# Enable this option for SSL Debugging
#export JAVA_OPTS=%JAVA_OPTS% -Djavax.net.debug=all
```

  Follow the instructions in the first line by uncommenting the indicated lines below it and setting the path and password.

- To start GAX manually by using `gax_startup.sh`, you might have to modify this file by replacing the following line:

  ```
  export GAX_CMD_LINE_ARGS=$*
  ```

  with the following command (use arguments that match your system):

  ```
  GAX_CMD_LINE_ARGS="-host <configuration server host> -port <configuration server port> -app
   <application name>"

  export GAX_CMD_LINE_ARGS
  ```

End

5b. Install Genesys Administrator Extension server on a Windows Server 2008 host.

Installing Genesys Administrator Extension server on a Windows Server host

Prerequisites

- The `Application` object for Genesys Administrator Extension server exists (see Step 4 of Creating the necessary configuration objects for Genesys Administrator Extension).
- The environment variable for `JRE_HOME` has been configured (see Step 2 of Setting up the host for Genesys Administrator Extension server).

Start

1. Copy the IP to the host machine.
2. Run the installation file to extract and copy the necessary files by entering the following command:

   ```
   ./setup.exe
   ```

   If there is an existing installation of GAX on the host, the installer will display a dialog box that prompts you to confirm whether or not you want to maintain the existing installation.

   If there is not an existing installation of GAX on the host, then you must specify the location of the Tomcat folder (refer to Installing Tomcat).
3. Enter information as prompted by the installation file, as follows:
   a. Enter the name of the host where Configuration Server is installed.
   b. Enter the port number used by Configuration Server.
   c. Enter the username and password used to access Configuration Server.

Important

- To start GAX manually by using `gax_startup.bat`, you might have to modify this file by replacing the following line:

  ```
  set GAX_CMD_LINE_ARGS=%*
  ```

  with the following command (use arguments that match your system):

  ```
  set GAX_CMD_LINE_ARGS=-host confserv -port 2020 -app gaxappobjname
  ```
- The GAX installer creates a `setenv.bat` file that enables you to adjust the memory settings for GAX. The `setenv.bat` file defines the memory (RAM) settings for GAX to 1024 MB. You can change the memory setting in the `setnev.bat` file to a different value. If you enable TLS encryption, ensure that you make the following updates to the `setenv.bat` file. The `setenv.bat` file contains the following lines:

  ```
  REM Uncomment the following lines only if you are going to use TLS. Don't forget to set
  the correct path and password.
  REM set JAVA_OPTS=%JAVA_OPTS% -Djavax.net.ssl.trustStore="C:\Program Files\Java\jre6\lib\
  security\cacerts"
  REM set JAVA_OPTS=%JAVA_OPTS% -Djavax.net.ssl.trustStorePassword=secret_password
  ```

  Follow the instructions in the first line by uncommenting the indicated lines below it and setting the path and password.

End

6. Choose which GAX functionality is available on the host.

Choosing which GAX functionality is available on the host

Purpose

- To add or remove GAX internal modules to control which GAX functionality is available for users on a specific host.

You can install multiple instances of GAX on multiple hosts to support HA, load balancing, and the availability of functionality. Each deployment of GAX shares the same data resources, such as Configuration Server, the GAX database, audio resources, and so on.

The plug-in architecture of GAX enables you to add or remove modules to control the availability of functionality.

Start

1. Deploy GAX as described in Installing Genesys Administrator Extension server on a Linux host on each host (this procedure is performed on a Linux install, but works the same on Windows if you adjust the paths according to your installation).
2. Navigate to the following folder: `<Tomcat Home>/webapps/gax/WEB-INF/lib`
3. Remove the `gax-*.jar` files that contain the functionality that you want to restrict. For example, if you do not want the user on the host to deploy audio resources, remove the `gax-opm-arm*.jar` file.

> **Warning**
>
> Do not remove the `gax-core*.jar` and `gax-plugin-api*.jar` files. These files are required to run the core functionality of GAX. Also, do not remove any of the JAR files that do not begin with `gax-`.

4. Add any external plug-in `*.jar` files that will use the functionality of the plug-in by copying the `*.jar` file to the `<Tomcat Home>/webapps/gax/WEB-INF/lib` folder.

> **Important**
>
> You can choose to remove all of the standard GAX plug-in modules, except the core plug-ins, to run GAX with only the plug-in functionality.

5. Restart Tomcat after adding or removing plug-ins for changes to take effect.

End

7a. Set up the database for Oracle.

Setting up the Genesys Administrator Extension database (for Oracle)

Purpose

- To set up the Oracle database that is used by Genesys Administrator Extension.

If you prefer to use PostgreSQL or Microsoft SQL Server, see Setting up the Genesys Administrator database (for Microsoft SQL Server) or Setting up the Genesys Administrator database (for PostgreSQL).

Start

1. Refer to the Oracle documentation to install the Oracle Database Management System on the host machine that corresponds to the `Host` object that you configured in Step 2 of the procedure Creating the necessary configuration objects for Genesys Administrator Extension.
2. Use the following SQL commands to create the users and ensure that they do not have excessive permissions:

   ```
   create user <username> identified by <password>;

   grant connect, resource to <username>;
   ```

3. Initialize the database by executing the following three scripts in the order below. The scripts are available in the following folder: `<installation folder>/resources/sql_scripts/oracle`

- `core_init_ora.sql`
- `opm_arm_init_ora.sql`
- `asd_init_ora.sql.`

> **Important**
> Error messages about unsuccessful execution of DROP statements might be displayed. Ignore these error messages. To verify that there are no real errors, execute the scripts twice. No errors should be displayed the second time.

- Connect the GAX Server application object to the DAP that you created in Step 3 of Creating the necessary configuration objects for Genesys Administrator Extension.

End

> Important
>
> To enable UTF-8 character encoding, see Enabling UTF-8 character encoding (for Oracle).

7b. Set up the database for Microsoft SQL.

Setting up the Genesys Administrator Extension database (for Microsoft SQL Server)

Purpose

- To set up the Microsoft SQL Server database that is used by Genesys Administrator Extension.

If you prefer to use Oracle or PostgreSQL, see Setting up the Genesys Administrator database (for Oracle) or Setting up the Genesys Administrator database (for PostgreSQL).

Start

1. Refer to the Microsoft SQL Server 2008 R2 documentation to create the Microsoft SQL Server Database for GAX on the host machine that corresponds to the `Host` object that you configured in Step 2 of the procedure Creating the necessary configuration objects for Genesys Administrator Extension. You can create the login and database, then execute the database scripts for MSSQL Server by using SQL Server Management Studio.
2. Start SQL Server Management Studio.
3. Connect to Microsoft SQL Server 2008 as `sa`.

   - Server type: `Database Engine`
   - Server name: `Local`
   - Authentication: `SQL Server Authentication`

- Create a login and password for the GAX database. For example: `gax812admin` with the password `password`.
- Create the GAX database (for example, `gax812`) by using the login to make this login the owner of the database.

> ### Important
> When you create the login, uncheck the `Enforce password policy` check box.

- Verify that you can connect to the database with the login that you created:
  - Server type: `Database Engine`
  - Server name: `Local`
  - Authentication: `SQL Server Authentication`
- Execute the following scripts in the order below, by using the Microsoft SQL Server 2008 Query Editor. The scripts are available in the following folder: `<installation folder>/resources/sql_scripts/mssql`

  a. `core_init_mssql.sql`
  b. `opm_arm_init_mssql.sql`
  c. `asd_init_mssql.sql`

> ### Warning
> The following fields must have a combined size of 900 bytes or less:
>
> - `asd_sd table (folder, version, svc_name, tenant_id)`
> - `asd_ip table (folder, nickname, os, tenant_id, version, localeid, buildnumber)`
>
> This constraint is indicated during execution of the `asd_init_mssql.sql` script with the following warning:
>
> `The maximum key length is 900 bytes.`
>
> Microsoft SQL Server cannot store indexes that have a size greater than 900 bytes.

Error messages might be displayed during script execution, because there are `DROP` statements in the script that might be trying to drop tables or constraints that do not exist.

You can verify that the errors do not exist by executing the scripts twice.

End

7c. Set up the database for PostgreSQL.

Setting up the Genesys Administrator Extension database (for PostgreSQL)

Purpose

- To set up the PostgreSQL database that is used by Genesys Administrator Extension.

> **Important**
>
> - This procedure applies only to GAX 8.1.310 releases or higher.
> - It is recommended to use PostgreSQL version 9.1.8.

If you prefer to use Oracle or Microsoft SQL Server, see Setting up the Genesys Administrator database (for Oracle) or Setting up the Genesys Administrator database (for Microsoft SQL Server).

Start

1. Refer to the PostgreSQL 9.1 documentation to create the PostgreSQL Database for GAX on the host machine that corresponds to the `Host` object that you configured in Step 2 of the procedure Creating the necessary configuration objects for Genesys Administrator Extension.

   Create the login account and database, then execute the database scripts for PostgreSQL by using pgAdmin.

2. Start pgAdmin.
3. Select the PostgreSQL 9.1 connection and connect to the PostgreSQL database with the following user name: `postgres`.

   > **Important**
   >
   > If a PostgreSQL 9.1 connection is not available, you can create it by clicking the `Add Server` button.

4. Create a login and password for the GAX database.

   For example: `gax813admin` with the password `password`.

   You can execute queries by clicking the `Query Tool` button. For example:

   ```
   CREATE USER gax WITH PASSWORD 'gax813admin' CREATEDB;
   ```

5. Create the GAX database (for example, gax813) by using the login created in Step 4 to make this login the owner of the database.

   ```
   create database gax813 owner gax;
   ```

6. Connect to the database with the login that you created in Step 4.
7. Execute the following scripts in the order below, by using pgAdmin. The scripts are available in the following folder:

   ```
   <installation folder>/resources/sql_scripts/postgres
   ```

   a. `core_init_postgres.sql`
   b. `opm_arm_init_postgres.sql`
   c. `asd_init_postgres.sql`

End

(Optional) Enable UTF-8 character encoding for Oracle databases.

Enabling UTF-8 character encoding (for Oracle)

To enable UTF-8 character encoding for Oracle databases in Genesys Administrator Extension releases 8.1.3 and higher, you must ensure that:

- Configuration Server 8.1.2 is installed.
- UTF-8 string encoding is enabled on Configuration Server 8.1.2.

The database character set must be set to `AL32UTF8` to support the use of UTF-8 character encoding. To verify the character set, use the following SQL command:

```
SELECT * FROM NLS_DATABASE_PARAMETERS;
```

In the response, if `NLS_CHARACTERSET` is set to `AL32UTF8`, no additional actions are required. Otherwise, refer to the Oracle support guide for more information about character set migration:

http://docs.oracle.com/cd/B28359_01/server.111/b28298/ch11charsetmig.htm

> Warning
>
> Character-set migration is a non-reversible process. Incorrect data conversion can lead to data corruption, so always perform a full backup of the database before attempting to migrate the data to a

> new character set.

> Important
>
> In most cases, a full export and import is recommended to properly convert all data to a new character set.

8. Configure Genesys Administrator Extension.

Configuring Genesys Administrator Extension

Purpose

- To set up role privileges and logging for Genesys Administrator Extension.

Start

1. Stop Tomcat, if it is running.
2. In Genesys Administrator, create at least one new `Role` object to provide access to the functionality in Genesys Administrator Extension. Follow the instructions in Genesys Administrator 8.1 Help.
   a. Define the privileges that are granted by the role on the `Role Privileges` tab.
   b. Assign the role to users and access groups on the `Members` tab as required.

      Refer to the Genesys 8.0 Security Deployment Guide for more information about roles and role privileges.

End

9. Start Genesys Administrator Extension.

Logging In

The Genesys Administrator Extension web-based interface runs on a web application server. It is loaded into your browser each time that you open the website where you installed Genesys Administrator Extension. You then log in.

> Important
>
> Genesys Administrator Extension supports the use of blank passwords only if Configuration Server is configured to allow blank passwords. Refer to the Genesys 8.0 Security Deployment Guide for information about using blank passwords.

Logging in to Genesys Administrator Extension

Prerequisites

- Configuration DB Server and Configuration Server are installed and running.
- An instance of a Genesys Administrator Extension `Application` object, configured as a server in Step 4 of Creating the necessary configuration objects for Genesys Administrator Extension, is connected to Configuration Server and running.
- Your browser and its windows are set to a resolution of 1024x768 or greater. If you are working in 1024x768, maximize the browser.
- The user logging in must have Read permission to their own `User` object and Read and Execute permissions on the Genesys Administrator Extension client object. Refer to the Genesys 8.0 Security Deployment Guide for information about permissions. Genesys Administrator Extension respects read-write permissions that are set for Environments and Tenants. You can only access those objects that you have permission to see.

Start

1. Start GAX by using Genesys Administrator.
2. Navigate to the `Application` object for the GAX instance that you intend to start/log in to.
3. Start the application by using the `Start` button in the icon bar.
4. Open a web browser.
5. Enter the following URL in the address bar of the browser:

   ```
   http://<Host name>:8080/gax/
   ```

   where <Host name> is the name of the computer on which you installed Genesys Administrator Extension. The port number is the port that was defined when setting up Tomcat in Installing Tomcat.

6. Log in to Genesys Administrator Extension with your assigned user name and password, and click `Log in`.

   > **Important**
   >
   > Each instance of Genesys Administrator Extension is associated with a single instance of Management Framework; Configuration Server and Port selection is not required during login, nor is it possible to select it.

   If you get a permissions error, refer to Required Permissions for instructions.

   Your login name and the tenant to which you are logged in is displayed in the top Header Bar of the Genesys Administrator Extension window. The time of your last login is displayed in the Preferences menu. See Preferences for more information.

   > **Important**
   >
   > The date and time of the local machine and the Management Framework machine must be synchronized for the last login time to be accurate.

7. Your account might be configured to set a new password the first time that you log in, or after a system administrator has reset your password. The `Change Password` dialog box is displayed:

   a. Enter a new password in the `New Password` field.

   b. Enter the same password in the `Confirm Password` field.

   c. Click `OK`.

   > **Important**
   >
   > Please see the Genesys 8.0 Security Deployment Guide for more information about resetting passwords.

End

Prerequisites for Genesys Administrator Extension Modules

This section describes prerequisites to be met before installing or using the functional modules of Genesys Administrator Extension. These are in addition to the basic prerequisites listed here, and are specific to the corresponding module.

Note: Unless specified otherwise, all commands that are entered on a command line in this section should be issued as a root user (command prompt of #) or as a regular user (command prompt of $).

Solution Deployment

Before using Solution Deployment to deploy Solutions to local and remote hosts, you must ensure that the following prerequisites are met:

- Hosts are set up and running at the remote locations, and are running Local Control Agent (LCA) and Genesys Deployment Agent (GDA). Use the instructions in Genesys Administrator 8.1 Help.
- The following configuration options are defined on the `Options` tab of the Genesys Administrator Extension server `Application` object in the `asd` section:
    - `silent_ini_path`
    - `local_ip_cache_dir`

  Refer to Configuration Options for more information about these options.

- Samba (or an equivalent Network File Server) is installed to enable communication between Genesys Administrator (Windows-based) and Genesys Administrator Extension (Linux-based). To install Samba, use the procedure Installing Samba. To install a Network File System (NFS), refer to the documentation specific to the server.
- SQL*Plus is installed. Use the procedure Installing SQL*Plus.

Installing Samba

Note:

- This procedure does not apply if you are using GAX as part of the Hosted Provider Edition.
- This procedure is optional if you are using GAX 8.1.3 releases or higher.

Purpose

- To allow Genesys Administrator Extension (Linux-based) to access files located on a Windows-based host, such as Genesys Administrator.

Start

On the command line interface:

1. Install Samba by entering the following at the # prompt:

   ```
   yum install samba system-config-samba
   ```

2. Set Samba to start up at boot by entering the following at the # prompt:

```
chkconfig smb on
```

3. Create a directory `/opt/gax` with Read/Write permissions for everyone by entering the following commands at the # prompt:

   ```
   mkdir /opt/gax/chmod 777 -R /opt/gax
   ```

4. If you have SELinux installed and active, make this directory accessible by entering the following command at the # prompt:

   ```
   chcon -t samba_share_t /opt/gax
   ```

5. To enable a shared directory called `repository` that is accessible by guests, modify the file `/etc/samba/smb.conf` file as shown:

   ```
   #/etc/samba.smb.conf
    # smb.conf file for use with GAX
    # this configuration allows sharing of IP packages between the GA and GAX system.
    ##============== Global Settings =============================
    [global]
    # you may change the workgroup name, but make sure that the GA.net
    # windows host is in the same workgroup!
    workgroup = HOME
    netbios name = SAMBA
    server string = Samba Server %v
    map to guest = Bad User
    log file = /var/log/samba/log.%m
    max log size = 50
    socket options = TCP_NODELAY SO_RCVBUF=8192 SO_SNDBUF=8192
    preferred master = No
    local master = No
    dns proxy = No
    security = share

    # Share is accessible via the name in [brackets]
    [repository]
    path = /opt/gax
    writeable = yes
    guest only = yes
    guest ok = yes
    create mask = 0777
    directory mask = 0777
    case sensitive = no
    }
   ```

End

Installing SQL*Plus

Purpose

- To set up SQL*Plus to enable database manipulation during setup of Solutions.

Start

1. Install the library required by SQL*Plus by entering the following command at the # prompt: `yum install libaio`

2. Download Oracle Instant Client from:

   http://download.oracle.com/otn/linux/instantclient/112020/oracle-instantclient11.2-basic-11.2.0.2.0.x86_64.rpm

3. Download SQL*Plus from:

   http://download.oracle.com/otn/linux/instantclient/112020/oracle-instantclient11.2-sqlplus-11.2.0.2.0.x86_64.rpm

4. Set the following environment variables for your host:

   a. Inserting the following lines into the `/etc/profile` file:

   ```
   # add these for Oracle Instantclient / SQL*Plus
    export ORACLE_HOME=/usr/lib/oracle/11.2/client64
    export LD_LIBRARY_PATH=$ORACLE_HOME/lib:${LD_LIBRARY_PATH}
    export PATH=$ORACLE_HOME/bin:${PATH}
    export SQLPATH=$ORACLE_HOME/lib
   ```

   b. Make these environment variables effective to the current session by logging out, and then logging in again.

3. If SQL*Plus is installed correctly, you can connect by entering the following command at the $ prompt:

   ```
   sqlplus <username>/<password>@host:<port>/<Solutionname>
   ```

End

Operational Parameter Management

For the deployment of Parameter Groups, ensure that you have write permissions to the `Transactions` folder of the tenant on which the Parameter Group is deployed. You must also have write privileges for the `Voice Platform Profiles` folder to deploy the Voice application and/or write privileges for the `Routing Scripts` folder to deploy Genesys IRD or SCXML routing strategies.

There are no additional prerequisites for using Operational Parameter Management in Genesys Administrator Extension. However, ensure that your Interaction Routing Designer (IRD) Routing Strategies reference the `Transaction` objects correctly.

Operational Parameter Management works together with Routing strategies, SCXML routing strategies, GVP voice applications, and Genesys Business Rules.

Note: Operational Parameter Management does not load strategies on DNs or upload applications to application servers. You must do this manually for all parameterized objects.

Audio Resource Management

To use Audio Resource Management in Genesys Administrator Extension, you must do the following:

- Add the configuration option section and options for Linux or Windows Server 2008.

  Linux:

  ### [+] Click here to reveal code

  ```
  [arm]
   local_announcement_folder=announcement
   local_music_folder=music
   local_os=RHEL5
   local_path=/opt/gax/arm
   local_sox_path=/usr/bin/sox
   target_announcement_folder=announcement
   target_music_folder=music
   target_os=RHEL5
   target_path=/mnt/arm/target
   delete_from_db_after_processing=false
  ```

  Windows Server 2008:

  ### [+] Click here to reveal code

  ```
  [arm]
   local_announcement_folder=announcement
   local_music_folder=music
   local_os=Windows
   local_path=C:\GCTI\GAX\arm\local
   local_sox_path=C:\GCTI\GAX\sox\sox.exe
   target_announcement_folder=announcement
   target_music_folder=music
   target_os=Windows
   target_path=C:\GCTI\GAX\arm\target
   delete_from_db_after_processing=false
  ```

  See Configuration Options for a detailed description of the configuration options.

- If you will be converting audio file formats, you must install SoX (Sound Exchange) before doing any conversions. Genesys Administrator Extension supports the bundled SoX in RedHat 5 (version 12) only. On Windows, SoX version 14.3.1 is supported.
- In the current release, Genesys Administrator Extension supports only SoX version 14.3.1. Follow the procedure Installing SoX below.
- Set up the target storage for Audio Resource Management by following the procedure Setting up ARM Runtime Web Server below. This procedure sets up an Apache web server on a Red Hat Enterprise Linux host. On this host, it creates a shared directory from which audio files are retrieved by Audio Resource Management, and to which Genesys Administrator Extension writes audio resource files as they are uploaded by users. The shared directory is accessible from the Genesys Administrator Extension host and is referred to as "target storage".

Installing SoX

Purpose

- To install SoX to enable conversion of audio resources to µ-law, a-law, and gsm formats. This procedure can be run at any time before or after Genesys Administrator is installed.

Start

1. Download SoX for your server operating system (Linux or Windows Server 2008).

   The Windows Server 2008 version is available here: http://sourceforge.net/projects/sox/files/sox/14.3.1/

2. To install SoX on Linux, enter the following command at the # prompt: `yum install sox`

   Or,

   To install SoX on Windows Server 2008, execute the installer application and install `sox.exe` into the following directory:

```
C:\Program Files\SoX\sox.exe
```

> **Note**: The user of the host on which the GAX Tomcat is running must be configured to read and execute the sox binary.

End

Setting up ARM Runtime Web Server

Purpose

- To set up the target storage for Audio Resource Management by setting up a shared directory on an Apache web server on a Red Hat Enterprise Linux host, from which audio files are retrieved by Audio Resource Management and to which Genesys Administrator Extension writes audio resource files as they are uploaded by users.

> **Note**: The ARM Runtime Web Server is sometimes referred to as ARM HTTP Proxy.

Prerequisites

- Genesys Administrator Extension Host is running.
- A dedicated host machine is available for the ARM Runtime Web Server.
- Media Server is available.

Start

1. Set up your Network File System (NFS) to share data between Genesys Administrator Extension and the

ARM Runtime Web Server.

a. (Linux) On the ARM Runtime Web Server, create the required folders and subfolders by entering the following commands at the # prompt:

```
mkdir /opt/genesys/arm
mkdir /opt/genesys/arm/music
 mkdir /opt/genesys/arm/announcements
```

> **Note**: Ensure that the user of the host on which the GAX Tomcat is running is configured to read and write these directories. GAX treats all directories as local. If the target directory and the sub-directories reside physically on a remote host and are used as network directories, or mapped as a local drive, the user must have network access configured.

b. On the Genesys Administrator Extension host, open the `/etc/exports` in an editor and add the folder `/opt/genesys/arm` as a shared directory. When added, the file should contain the following line:

```
/opt/genesys/arm * (rw,sync)
```

To limit access to only certain machines, change the asterisk (*) to the fully qualified domain name or address of the Genesys Administrator Extension host. If you have multiple Genesys Administrator Extension hosts in your environment, you can create one line per host.

3. On the ARM Runtime Web Server, make sure that NFS and the supporting portmap processes have started by entering the following commands at the # prompt:

```
chkconfig portmap on
 chkconfig nfs on
```

If necessary, you can manually start the processes by entering the following commands at the # prompt:

```
Solution nfs start
 Solution portmap start
```

4. Mount the shared drive on the Genesys Administrator Extension host (or hosts) as follows:

   a. On the host, create a new directory by entering the following command at the # prompt:

   ```
   mkdir -p /mnt/arm/target
   ```

5. Open the file `/etc/fstab` in an editor and add the following line:

```
<address of the ARM Runtime Web Server>/opt/genesys/arm
 /mnt/arm/target nfs rsize=8192,wsize=8192,timeo=14,intr
```

6. Mount the target manually by entering the following command at the # prompt:

```
mount /mnt/arm/target
```

The target is mounted automatically when the server restarts.

- Install Apache Web Server as follows:

a. Install Apache by entering the following command at the # prompt:

```
yum install httpd
```

b. Make sure that Apache starts when the host starts by entering the following command at the # prompt:

```
chkconfig httpd on
```

Alternately, you can start Apache manually by entering the following command at the # prompt:

```
Solution httpd start
```

c. Start or restart Apache to test that it works.

- To have Apache serve the media files for the Media Server, open the file `/etc/httpd/conf/httpd.conf` in an editor and make the following changes:

Change This Line ...	... to this Line
`DocumentRoot "/var/www/html"`	`DocumentRoot "/opt/genesys/arm"`
`<Directory "/var/www/html">`	`<Directory "/opt/genesys/arm">`

- Update your Media Server configuration to use the ARM Runtime Web Server (`address:http://<address of ARM Runtime Web Server>/`) instead of the local file storage. Refer to the Genesys Media Server 8.1 Deployment Guide.

End

License Usage Reporting

> **Note**: This section only applies to GAX 8.1.301 releases or lower. In GAX 8.1.310 releases or higher, License Usage Reporting functionality is provided by the License Reporting Manager (LRM) plug-in for GAX.

Before using License Usage Reporting in Genesys Administrator Extension to generate and view reports of License Resource Manager data, you must ensure that the following prerequisites are met.

- There must be a License Resource Manager (LRM) system running that aggregates data nightly by using a `cron` job. Refer to LRM documentation for details about this process.
- A Database Access Point for the LRM Database is configured, and added to the connections of the Genesys Administrator Extension. Use the following procedure:

Configuring a Database Access Point for the LRM database

Start

1. Create and configure a Database Access Point (DAP) `Application` object, which is necessary for connectivity to the LRM Database by Genesys Administrator Extension.
2. Add this DAP to all Genesys Administrator Extension server objects by using the instructions in the Framework 8.1 Deployment Guide.
3. In addition to following those instructions, do the following:
    a. Open the `Options` tab.
    b. Create a new section called `GAX`.
    c. In this new section, add the configuration option `role` and set its value to `LRM`. This identifies this DAP as the one for the LRM Database that is used by Genesys Administrator Extension.
4. Add this DAP to all Genesys Administrator Extension server objects.

End

Configuring System Security

GAX has many features that enhance your system security. This section discusses GAX security features and describes how to configure and/or use them.

Default Account Support

Genesys uses a default user account. This is a special account that always has full privileges to all objects and can perform any action. This account ensures that there is always at least one account that enables the administrator to correct permissions and access issues if other administrative accounts are deleted, disabled, or otherwise compromised.

GAX supports the default user account. The default user account always has full access to all the functions that are specified for the GAX Role, even if this account does not have any role privileges or explicit permissions specified. When the default account is created during the installation of Configuration Server, it has full control over all configuration objects; however, this account might be deleted or its permissions on objects might be revoked. If this happens, GAX cannot work around the permissions. The default account must have the permissions set to write objects in the Configuration Server.

Use the default_account_dbid option to configure the actual account to be used, and that has all privileges assigned, in case the original default user account is disabled for security reasons or has been deleted.

Transport Layer Security (TLS)

GAX employs Transport Layer Security (TLS), a cryptographic protocol that provides security and data integrity for communications over networks such as the Internet. TLS encrypts the segments of network connections at the transport layer from end to end.

GAX supports TLS-enabled connections to the following Genesys servers:

- Configuration Server
- Solution Control Server
- Message Server
- Genesys Deployment Agent

GAX also supports TLS-enabled connections to the GAX database and the LRM database.

For the GAX database connection (either Oracle, Microsoft SQL Server, or PostgreSQL), the database driver and database must also support TLS. For information about configuring your GAX database, refer to the documentation that is specific to the database that you are using:

- Oracle: Oracle Database Advanced Security Administrator's Guide
- Microsoft SQL Server 2008 R2: Use the documentation that came with your database application.
- PostgreSQL: Use the documentation that came with your database application.

For information about TLS and detailed instructions about configuring secure connections, and creating and managing certificates, refer to the Genesys TLS Configuration chapter of the Genesys 8.0 Security Deployment Guide.

Follow the instructions to create a certificate, assign that certificate to a host object (which is required for Genesys Server to run in TLS mode), and configure the use of a secured port for the GAX application.

Next, import the server certificate to the trust storage for GAX to enable authentication for TLS connections.

By default, trust storage is in the JRE folder at the following location:

```
C:\Program Files\Java\jre6\lib\security\cacerts
```

The default password is "`changeit`".

Genesys recommends that you create a separate trust store for GAX.

Perform the procedure below to create a trust store and import the certificates.

Creating a keystore and managing the trust store

Purpose

- To create storage that is separate from the default keystores that come with Java.

Genesys recommends that you do not use the default keystores that are shipped with Java. To ensure a clean separation, you should create a separate storage. If you use a standard `cacert` file, you must re-import the certificates after each JVM update.

The trust store should contain only the certificates of servers that GAX should trust. If a server sends GAX its certificate during a TLS Handshake, GAX will search for a matching certificate in this keystore. If the certificate is found, the connection is accepted; otherwise, the connection is rejected.

Prerequisites

- Your Keytool should be configured to your path.
- You have JRE or JDK installed.

Start

1. To create an empty keystore, execute the following command lines on your shell: `keytool -genkey -alias initKey -keystore trusted.keystore -storetype jks` `keytool -delete -alias initKey -keystore trusted.keystore`
2. Make the `trusted.keystore` file readable for the user that owns the GAX process.
3. Set a strong password on your keystore.
4. Add a certificate to the trust store by executing the following command line: `keytool -import -alias mssql -keystore trusted.keystore -file "cert/demosrc.cer"`

   `Alias` is a name under that the certificate. It can be addressed within the trust store. The option `-keystore` specifies the keystore file and the option `-file` specifies the certificate to be imported.

5. To display the whole content of a keystore, execute the following command line: `keytool -list -keystore trusted.keystore`
6. To display a specific certificate, execute the following command line: `keytool -list -v -alias mssql -keystore trusted.keystore`
7. To delete a certificate from the keystore, execute the following command line: `keytool -delete -alias mssql -keystore trusted.keystore`

End

	💡 **Note**: Most systems have multiple trusted stores. You must always use the same store for GAX.

The following options must be set to configure the trust store location for GAX. The options also enable authentication on a global level for all connections that use a secured port.

The best way to set these options is by using the `setenv.sh` or `setenv.bat` script:

```
set JAVA_OPTS=%JAVA_OPTS% -Djavax.net.ssl.trustStore="D:\certificates\trusted.keystore"
set JAVA_OPTS=%JAVA_OPTS% -Djavax.net.ssl.trustStorePassword=changeit
```

Note: GAX does not support Client Authentication. GAX will not authenticate itself by sending a certificate to the server.

TLS: Preparing Genesys Management Framework

To enable GAX to connect securely to Genesys servers, you must configure the Genesys Framework as described in the Genesys 8.0 Security Deployment Guide. Follow the instructions in this guide to create and manage certificates and make them usable within Genesys Framework.

Configuration Server

You must meet the following conditions to create a secure connection to Configuration Server:

1. Create a an `Auto Detect` listening port for your Configuration Server with a certificate configured.
2. Configure the GAX Server to connect when it starts up to the Configuration Server Auto Detect port by setting the GAX Server "-port" property. In the `Start Info` tab of the `GAX_Server Properties` dialog box, enter the following settings:
   - Working Directory: `/path/gax`
   - Command Line: `./startup.sh`
   - Command Line Arguments: `-host <host name> -port <auto detect port number> -app GAX_Server`

Message Server and Solution Control Server

Both Message Server and Solution Control Server are configured the same way.

1. Create a `Secured` port for Message Server and Solution Control Server.
2. Configure the GAX Server to connect to Message Server and Solution Control Server by using the *specific* `Secured` ports that you have created. In the `Properties` dialog box for the server and in the `Connections` tab of the `GAX_Server` dialog box, secured ports are displayed with a key symbol icon.
3. Restart GAX Server to connect over an encrypted session by using the secure ports.

Genesys Deployment Agent

Genesys Deployment Agent (GDA) does not read its configuration from Configuration Server. The TLS for the GDA process is activated by accessing the `security` section of the local `gda.cfg` file and setting the `gda-tls` option to a value of 1.

The annex tab of the related host might or might not have a `security` section that contains the `gda-`

`tls` option.

The `gda-tls` option is not relevant for the GDA runtime; it is read during the installation of LCA and GDA only. GAX reads the value of the `gda-tls` option to determine in what mode GDA is running, and also to determine whether it should connect using TLS or not; therefore, these values must be kept synchronized. If the system administrator changes one of the values in the local file or in the host annex tab, the other option must also be changed to enable GAX to connect correctly.

Disabling Authentication for Certain Connections

The configuring steps outlined above engage authentication for Configuration Server, Message Server, and Solution Control Server. If GAX uses the secure ports to connect to Message Server and Solution Control Server, both server-side certificates will automatically be validated against the trust storage.

In certain rare cases you might want to disable authentication for one of the connections. To do this, add the following line to the `Advanced` tab of the `Properties` dialog box for the connections:

```
"disableAuthentication=1"
```

Do not use white spaces. To separate this option from other options, use a semi-colon.

To disable TLS authentication for Configuration Server, add the following line to the following files:

- (Linux) `setenv.sh`:

```
JAVA_OPTS="$JAVA_OPTS -Dgax.configserver.validate.cert=off"
```

- (Windows) `setenv.bat`:

```
set JAVA_OPTS=%JAVA_OPTS% -Dgax.configserver.validate.cert=off
```

Notes:

- Connections to Message Server and Solution Control Server fail if GAX does not find the received certificate in the trust store, or if Message Server and Solution Control Server do not send a certificate.
- Connections also fail to Configuration Server and databases if they are configured for authentication and the certificate is not in the trust store.

TLS: Configuring the GAX Database

You must configure your Oracle, Microsoft SQL 2008 R2, or PostgreSQL server to use TLS. Refer to the documentation that came with your database for information on how to use TLS security.

Configuring the GAX Database for TLS (Oracle)

Purpose

- To enable TLS support for your GAX Oracle database.

Prerequisites

- Setting up the Genesys Administrator database (for Oracle)

Start

1. Configure Oracle as described in the related database guides, and configure a TCPS listener.
2. Set the level of TLS control on the DAP.
    a. In the GAX section of the DAP, create an option that is named `tls_mode`.
    b. Specify one of the following values for the `tls_mode` option:

- `off`—No TLS will be used.
- `required`—If a server does not support TLS, revoke the connection.
- `authentication`—GAX will validate the server send-certificate with the local trust store.
- `<option not set>`—Same as `off`.

End

Configuring the GAX Database for TLS (Microsoft SQL Server 2008)

Prerequisites

- Setting up the Genesys Administrator database (for Microsoft SQL Server).
- Ensure that you are using the latest JTDS driver (1.2.5 or later).

Start

1. Configure Microsoft SQL Server as described in the related database guides.
2. Set the level of TLS control on the DAP.
    a. In the GAX section of the DAP, create an option that is named `tls_mode`.
    b. Specify one of the following values for the `tls_mode` option:

- `off`—Do not use TLS.
- `request`—If the server supports TLS, it is used.
- `required`—If the server does not support TLS, the connection is revoked.
- `authentication`—GAX validates the server-send certificate against the local trust store.
- `<option not set>`—Same as `off`.

- Verify that the configured port is identical to the TLS listener port of Microsoft SQL Server
- Due to an incompatibility between newer versions of Java and the Microsoft SQL Server driver, disable CBC Protection to enable GAX to connect to a Microsoft SQL Server database.

- For Windows, add the following line to the `setenv.bat` file:

```
set JAVA_OPTS=%JAVA_OPTS% -Djsse.enableCBCProtection=false
```

- For Linux, add the following line to the `setenv.sh` file:

```
JAVA_OPTS="$JAVA_OPTS -Djsse.enableCBCProtection=false"
```

End

Configuring the GAX Database for TLS (PostgreSQL)

	Note: This procedure applies only to GAX 8.1.310 releases or higher.

Prerequisites

- Setting up the Genesys Administrator database (for PostgreSQL).

Start

1. Configure PostgreSQL as described in the related database guides.
2. Set the level of TLS control on the DAP.
    a. In the GAX section of the DAP, create an option that is named `tls_mode`.
    b. Specify one of the following values for the `tls_mode` option:

- `off`—Do not use TLS.
- `required`—If the server does not support TLS, the connection is revoked.

- `authentication`—GAX validates the server-send certificate with the local trust store.
- `<option not set>`—Same as `off`.

End

Cross-site Scripting and Cookies

You can configure your system to improve the protection of Genesys Administrator Extension against Cross-site Scripting (XSS) attacks by configuring the `HttpOnly` and `Secure` flags on your HTTP server to further enhance the existing GAX security. These flags tell browsers how to handle cookies.

Server-side cookies can be tagged with `HttpOnly` and `Secure` flags to tell the browser how to deal with them. To achieve a maximum level of security, administrators must make this configuration on the Application Server.

Securing Server-side Cookies

HttpOnly

Setting the `HttpOnly` flag on cookies forces the browser to prevent (disallow) scripts from accessing the cookies. This prevents JavaScript that might be introduced through an XSS attack into a browser page to access cookie data and send it to a different person. Stolen cookie data can also be used to hijack a browser session.

Secure Flag

With the Secure flag set, cookies are transmitted only from the browser to the server when the connection is secured by using the HTTPS protocol. This setting is applicable to HTTPS connections only. Therefore, you must configure Tomcat to use an HTTPS connector, not an HTTP connector.

Setup

Follow these recommendations to configure the `HttpOnly` and `Secure` flags.

HttpOnly

Open and edit the following file: `$CATALINA_HOME/conf/context.xml`

To set the `HttpOnly` flag, add the following attribute:

```
useHttpOnly="true"
```

The main tag should be:

```
<Context useHttpOnly="true">
```

Instead of: `<Context>`

Secure Flag

Open and edit the following file: `$CATALINA_HOME/conf/server.xml`

To set the Secure flag, add the following attribute to the HTTPS connector:

```
secure="true"
```

The flag must not be applied to any non-HTTPS connectors. If you apply the flag to an HTTP connection, it will become unusable for Genesys Administrator Extension.

The following is an example of a valid Connector:

```
<Connector port="8443" protocol="HTTP/1.1" SSLEnabled="true"
maxThreads="150" scheme="https" secure="true"
keystoreFile="/home/gcti/keystore.key" keystorePass="genesys"
clientAuth="false" sslProtocol="TLS" />
```

Load Balance for Apache Tomcat

The following information applies to Apache Tomcat Configuration for communication by using the AJP protocol with a Web Server in front of it that is acting as a load balancer (for example, Apache Web Server. AJP is a protocol).

You might have to configure Load Balancing to accommodate the use of the `HttpOnly` and `Secure` flags.

In the `$CATALINA_HOME/conf/server.xml` file, make the following adjustment to support the Apache JServ Protocol (AJP) on the default connector:

```
<Connector port="8009" protocol="AJP/1.3" redirectPort="8443" secure="true"/>
```

Inactivity Timeout

For security purposes, GAX can be configured to lock the application if an administrator has not used the keyboard or mouse for a period that you specify. All user input is blocked until the administrator provides login information to unlock the application. This feature ensures that no unauthorized user can access an unattended terminal that is running GAX.

Use the `inactivity_timeout` option to specify the amount of time in minutes of administrator inactivity (no mouse or keyboard usage) that triggers application locking. If the administrator has been inactive longer than the number of minutes that are specified by the `inactivity_timeout` option, the administrator must re-authenticate to be able to use the GAX application. A value of `0` disables this functionality.

GAX employs a keep-alive strategy to prevent Tomcat *session* timeout; this ensures that GAX maintains your session even if the inactivity timeout feature locks the application and requires you to log in.

Configuring the Auditing Feature

To enable the Auditing feature, you must configure GAX, Message Server, and the database. GAX must have a connection to Message Server.

GAX Application

Enable Auditing by setting the value of the `auditing` option in the `general` section of the GAX Server application to `true`.

Next, in the `general` section of the GAX application, make the following configuration option settings:

- Set the number of switch-over attempts before GAX tries to switch-over to the redundant node to 1. To do this, set the value of the `max_switchovers` option in the `general` section of the GAX application to `1`. A value of `0` means switch-overs are disabled. Negative values allow an unlimited numbers of switch-overs, and, therefore, a continuuos reconnect process if the server is unreachable. Setting a negative value is recommended for production systems.
- Set the number of connection attempts before GAX tries to switch over to the redundant node. To do this, set the value of the `attempts` option in the `general` section of the GAX application to `1`.
- Set the timeout interval between connection attempts to Message Server. To do this, set the value of the `warmstandby_timeout` option in the `general` section of the GAX application to any positive integer.
- Set the protocol timeout, which is the time in seconds after which GAX gives up attempting to connect. To do this, set the value of the `timeout` option in the `general` section of the GAX application to `30`.

Message Server

In the Message Server object, set the `db_storage` option in the `messages` section to the value `true`.

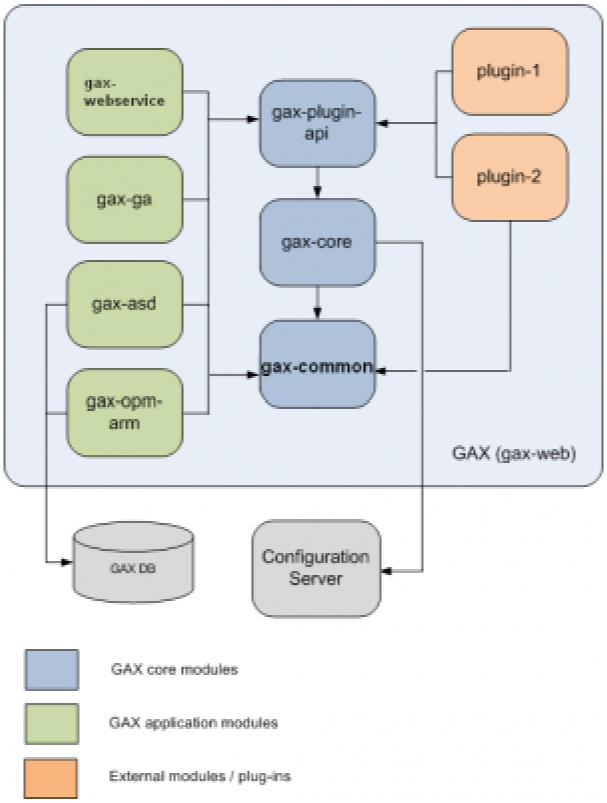
If the `db_storage` option is not set to `true`, Message Server does not save the Audit data to its database.

Database Configuration

To read the Audit data from the Log DB, a DAP must be configured and connected to the GAX Server `Application` object. Configure the DAP in the same way that DAPs were configured for the GAX database and License Usage Reporting. To identify the DAP role, set the value of the `role` option in the `GAX` section of the DAP to `auditing`.

Managing GAX Compatible Plug-ins

Genesys Administrator Extension 8.1.3 is deployed as a set of plug-ins into the GAX Core. This enables you to deploy only the functionality that you require, or to restrict the availability of certain functionality to users.



GAX is based on a hierarchical dependency system. The `gax-core` plug-in depends on the `gax-common` plug-in. The `gax-plugin-api` plug-in depends on the `gax-core` and `gax-common` plug-ins. All other GAX plug-ins depend on the `gax-plugin-api` and `gax-common` plug-ins.

The `gax-common` plug-in contains classes, such as error codes, exceptions, static utility classes, and interfaces, that are shared by both the `gax-core` and `gax-plugin-api` plug-ins. Most auditing related interfaces and objects are contained in the `gax-common` plug-in.

The `gax-core` plug-in manages all system-wide resources; therefore, all connections, threads, and stateful classes are contained in the `gax-core` plug-in.

The `gax-plugin-api` plug-in contains GAX functionalities that are used by other plug-ins. This plug-in contains generic configuration APIs, the base class of web access controller (BaseController), and other utility classes.

The `gax-webservice` plug-in contains all core web service interfaces that might be used in GAX.

	Note: If a plug-in contains configuration options, you must have write permissions on the GAX `Application` object for `SYSTEM`.

Using GAX to Manage Plug-ins

The Plug-in Management screen displays all installed plug-ins in your GAX environment. To access the screen, navigate to Configuration > Administrator > Plug-in Management.

You can click on the name of a plug-in to view details about the plug-in, such as which server hosts the plug-in. Click Plug-ins to display more information, which displays in a new panel to the right:

- Name—The name of the plug-in
- Version—The version number of the plug-in.
- Language—The language used by the interface of the plug-in
- Provider—The name of the user or company that provided the plug-in
- State—This field can be set to Enabled or Disabled, depending on the status of the plug-in. See Enabling or disabling a plug-in in GAX for more information.

The following actions can be performed in the Plug-in Management area:

- GAX plug-ins can be installed by using the standard IP installation mechanism. See Installing a GAX compatible plug-in by using the Software Import Wizard for more information.
- Plug-ins can also be installed manually by using the command line. See Installing a GAX compatible plug-in by using the command line for more information.
- Plug-in options can be modified. See Modifying plug-in settings for more information.
- Plug-ins can be enabled or disabled. See Enabling or disabling a plug-in in GAXfor more information.
- Plug-ins can be removed. See Removing a plug-in from GAX for more information.

Installing a GAX compatible plug-in by using the Software Import Wizard

Start

1. In the header, go to Configuration > Solution Deployment > Installation Packages.
2. In the Installation Packages panel, click New. A new panel called Software Import Wizard opens to the right.
3. In the Software Import Wizard panel, select a method for importing the plug-in:
   -

   - Installation Package Upload (includes templates)—Upload a ZIP file that contains an installation package and its associated templates. These files are typically provided by Genesys Technical Support.

   i. In the Software Import Wizard panel, select Installation Package Upload (includes templates) and click Next.

ii. The panel updates. Click `Choose File` to select the file to upload.
iii. Click `Finish`.

-

- `Installation Package Upload (template uploaded separately)`—Upload an installation package and its associated templates.

i. In the `Software Import Wizard` panel, select `Installation Package Upload (template uploaded separately)` and click `Next`.
ii. The panel updates and displays three boxes—`Upload a package`, `Upload an XML template`, and `Upload an APD template`. Click `Choose File` in each field to select the file to upload.

- `Upload a package`—A ZIP file that contains the installation package.
- `Upload an XML template`—The XML template file for this installation package. This is the template that is referenced by the installation package description file. This file must not be modified from the version in the template directory.
- `Upload an APD template`—The APD template file for this installation package. This is the template that is referenced by the installation package description file. This file must not be modified from the version in the template directory.

- Click `Finish`.
  -

- `UNC Path to Mounted CD or Directory`—Upload an installation package that is stored on a mounted CD or network directory.

i. In the `Software Import Wizard` panel, select `UNC Path to Mounted CD or Directory` and click `Next`.
ii. In the text field, enter the path for where the installation package is stored.
iii. Click `Next` to open the path.
iv. The panel updates to display the installation package(s) that is found at the specified location. Click the check box(es) that is beside the installation package(s) to upload.
v. Click `Finish`.

-

- `UNC Path to an Existing Administrator Repository`—Upload an installation package from an existing Genesys Administrator repository.

i. In the `Software Import Wizard` panel, select `UNC Path to an Existing Administrator Repository` and click `Next`.
ii. In the text field, enter the path for the existing Genesys Administrator repository.
iii. Click `Next` to open the path.
iv. The panel updates to display the installation package(s) that is found at the specified location. Click the check box(es) that is beside the installation package(s) to upload.
v. Click `Finish`.

The file(s) upload from your file system to Genesys Administrator Extension and a progress bar displays to show the upload progress. The progress of the upload also displays in the Status column in the `Installation Packages` panel.

	Notes: • A green progress bar represents a successful upload for the installation package. A red progress bar represents a failed upload for the installation package. You can review which step failed in the `Status` field in the `Installation Packages` list. • You cannot upload a plug-in to the repository if a version of the plug-in already exists in the repository. You must have the `Replace IPs and SPDs` privilege enabled to overwrite a plug-in in the repository.

End

Installing a GAX compatible plug-in by using the command line

Start

1. Deploy the following elements of a GAX plug-in:
    a. Copy the plug-in Java `jar` file to the following location:

    ```
    $CATALINA_HOME/webapps/gax/WEB-INF/lib
    ```

    b. Import the template data in the ADP file into Configuration Server as an application template.

    c. Import the XML file that contains role and option metadata into the plug-in application template object.

    d. (Optional) Execute the database schema against the GAX schema.

5. Use Genesys Administrator to prepare Configuration Server by assigning the role privileges that are defined by the plug-in.

6. Prepare the database schema. If the plug-in uses the GAX database or other databases, run the initialization or upgrade script that comes with the plug-in. If the plug-in comes with an SPD file, you can use Solution Deployment to run the script when you install the file.

7. Deploy the IP to the target GAX host.

    a. Stop GAX.

    b. Copy the plug-in`.jar` file to the Tomcat container in the `gax/WEB-INF/lib/` folder by using the Genesys installation package (either executed manually as defined in the deployment guide for the plug-in, or by using an SPD file). The file can now be used by GAX when the application is started.

3. If your plug-in uses configuration options, they are merged with the GAX options and can be configured with the GAX options. Use Genesys Administrator to configure the plug-in options.

	Note: Option description texts and validity rules that are used by Genesys Administrator are not applied when editing options for plug-ins.

4. Use Genesys Administrator to review the roles assigned to the plug-in by the imported XML metadata for your plug-in.
5. Start GAX. Installation of the plug-in occurs during startup.

   GAX registers all available plug-ins and reads out the `tpl` file that is located in the `/META-INF` folder.

   The options for each plug-in might be in the GAX section, or in a section named for the plug-in.

End

Modifying plug-in settings

Purpose

- To provide a consolidated system settings interface for accessing and managing all system settings in GAX. System settings are managed by node.

Start

1. In the header, go to `Configuration > Administrator > Plug-in Management`.
2. Select an application in the `Administrator Applications` list. A new panel opens to the right.
3. Click `Plug-ins` to view which plug-ins are associated with the application. A new panel opens to the right.
4. Select a plug-in in the `Plug-in Info` list. A new panel opens to the right.
5. Click `Plug-in Options`. A new panel opens to the right. The panel displays the options that are associated with the plug-in.
6. Click an option to view more information about the option in a separate panel that opens to the right.
7. When you have finished modifying the option(s), perform one of the following actions:

- Click Save to save your changes.
- Click `Cancel` to discard your changes.

End

Enabling or disabling a plug-in in GAX

Notes:

- It is not possible to disable the `gax-core` plug-in.
- The option to enable or disable a plug-in is available only for the application or node to which the user is currently connected. Other GAX applications or nodes will provide a link to manually login to that instance.

Start

1. In the header, go to `Configuration > Administrator > Plug-in Management`.
2. Select an item in the `Administrator Applications` list. More information about the item displays in a new panel to the right.
3. Click `Plug-ins`. More information about the plug-ins for the item display in a panel to the right.
4. Select a plug-in from the list.
5. Do one of the following:

- If the plug-in is currently enabled, the `Disable` button is displayed. Click `Disable` to disable the plug-in.
- If the plug-in is currently disabled, the `Enable` button is displayed. Click `Enable` to enable the plug-in.

End

Note: To see the changes to the plug-in, refresh the display in your browser.

Removing a plug-in from GAX

Start

1. Stop GAX.
2. Go to `$CATALINA_HOME/webapps/gax/WEB-INF/lib` on the file system (where `$CATALINA_HOME` is your home folder for the Tomcat process).
3. Delete the .jar files for the plug-ins that you wish to disable. For example, to remove the `Solution Deployment` plug-in, delete `gax-asd.jar`.
4. Start GAX.

End

Upgrading GAX

This section describes how to upgrade from previous versions of GAX to the current version.

Upgrading from 8.1.x to 8.1.3

This section contains two procedures. Use the one that applies to your system:

- Upgrading to the latest Genesys Administrator Extension for Management Framework 8.1.1 or higher.
- Procedure: Upgrading to the latest Genesys Administrator Extension for Management Framework 8.1.0 or lower

The following are the generalized steps to upgrade GAX 8.1.x to GAX 8.1.3:

1. Stop GAX.
2. Create the application objects within Configuration Server.
3. Install the GAX IP.
4. Add or remove plug-ins (refer to Managing GAX Compatible Plug-ins or the deployment guide for your plug-in).
5. Upgrade the database for GAX.
6. Start GAX.

Upgrading to the latest Genesys Administrator Extension for Management Framework 8.1.1 or higher

Purpose

- To upgrade from an earlier release of GAX to the latest release for Management Framework 8.1.1 or higher.

Start

1. Stop the instance of GAX that you want to upgrade.
2. Ensure that Management Framework, Configuration Server, and Genesys Administrator are all upgraded to versions that are compatible with the latest version of GAX before proceeding (refer to Prerequisites for Genesys Administrator Extension Modules).
3. This step applies only to instances that use GAX `Application` object of type `Genesys Generic Server`.

   Create and configure the configuration objects that are required for the latest version of GAX by using Genesys Administrator to perform the following steps:

   a. Open your existing GAX `Application` object of type `Genesys Generic Server` in edit mode.

   b. Click the `Options` tab.

   c. Click `Export` to save your configured GAX options to a file on your local file system of type `CONF/CFG`.

   d. Create and configure a new Server Application object for Genesys Administrator Extension of type `Genesys Administrator Server` by following Step 4 of Creating the necessary configuration objects for Genesys Administrator Extension.

      i. Ensure that you follow the steps that pertain to the use of Management Framework Configuration Server 8.1.1, or higher.

      ii. Replicate any configuration that you wish to add to your newly created `Application` object by referring to the GAX `Application` object of your previous version.

      iii. Click the `Options` tab.

      iv. Click `Import` and specify the `CONF/CFG` file that you previously created. Select `No` to not overwrite any existing options.

      v. (Optional) Create a DAP that points to the Log Database (refer to Step 3 of Creating the necessary configuration objects for Genesys Administrator Extension. Set the role of the DAP to auditing. Enable auditing by setting the value of the `general/auditing` option to `true`. Add the DAP to your GAX connections. On the `Options` tab of the DAP, in the `GAX` section, configure the `role` option with the value `auditing`.

5. On the target machine, run the GAX installer for the release to which you want to upgrade. The installer copies the binary file to the Tomcat instance that was defined during installation and copies all of the

required files to the target directory. For more details, refer to the Installing Genesys Administrator Extension server on a Linux host or Installing Genesys Administrator Extension server on a Windows Server 2008 host.

6. Remove or deactivate all old GAX objects. You can use only one GAX `Application` object to point to one physical GAX instance. If you want more than one GAX `Application` object to point to a single machine, you must install separate physical GAX instances on the same machine, each with a separate, independent, Tomcat installation.
7. Execute all of the applicable database upgrade scripts, if necessary. To determine if you have to apply any database scripts, check the `resources/sql_scripts` folder in the target directory of the installation.

	Note: GAX database schema version numbers are not necessarily synchronized with the version numbers of plug-ins, nor will they necessarily match the GAX release number. For example, your version of GAX might be 8.1.201.54 and your database schema version might be 8.1.201.25.

Perform one of the following steps, depending on whether you are using Oracle or Microsoft SQL:

- (Oracle only) Run all of the database upgrade scripts from the previous version. To determine if you have to apply any database scripts, check the `resources/sql_scripts` folder in the target directory of the installation.

  For example, if you have release 8.1.201.25 running and you intend to upgrade to release 8.1.300.XX, you must execute the following SQL scripts:

  - `gax_core_upgrade_db_8.1.201.25_to_8.1.300.XX_ora.sql`
  - (For Solution Deployment only) `gax_asd_upgrade_db_8.1.201.15_to_8.1.300.XX_ora.sql`
  - (For Operational Parameter Management only) `gax_opm_upgrade_db_8.1.201.15_to_8.1.300.XX_ora.sql`
- (Microsoft SQL only) Run all of the database upgrade scripts from the previous version. To determine if you have to apply any database scripts, check the `resources/sql_scripts` folder in the target directory of the installation.

  For example, if you have release 8.1.201.25 running and you intend to upgrade to release 8.1.300.XX, you must execute the following SQL scripts:

  - `gax_core_upgrade_db_8.1.201.25_to_8.1.300.XX_mssql.sql`
  - (For Solution Deployment only) `gax_asd_upgrade_db_8.1.201.15_to_8.1.300.XX_mssql.sql`
  - (For Operational Parameter Management only) `gax_opm_upgrade_db_8.1.201.15_to_8.1.300.XX_mssql.sql`

	Notes: • Files that have version numbers prior to the ones from which you upgraded do not have to be executed. • You must log in to the database schema as a

	GAX user and run the commands inside the SQL scripts as commands for the database. • If you are installing GAX for the first time or upgrading from release 8.1.x to 8.1.3, when you execute the SQL upgrade scripts, make sure that the scripts are properly committed. If your client application has auto-commit switched off, you might have to add the following line(s) to the scripts. • For Oracle: `commit;` • For MS SQL: `BEGIN TRANSACTION;COMMIT TRANSACTION;`

- (Optional) You can delete the previous GAX `Application` object after you have verified that the new release is working correctly.

End

Install GAX

Refer to the Installing Genesys Administrator Extension server on a Linux host or the Installing Genesys Administrator Extension server on a Windows Server 2008 host.

Upgrading to the latest Genesys Administrator Extension for Management Framework 8.1.0 or lower

Purpose

- To upgrade from an earlier release of GAX to the latest release of GAX for Management Framework 8.1.0 or lower.

Start

1. Stop the instance of GAX that you intend to upgrade.
2. On the target machine, run the GAX installer for the release to which you want to upgrade. The installer copies the binary file to the Tomcat instance that was defined during installation, and also copies all of the required files to the target directory.
3. Perform one of the following actions, depending on your server installation:

	Note: GAX database schema version numbers are not necessarily synchronized with the version numbers of plug-ins, nor will they necessarily match the GAX release number. For example, your version of GAX might be 8.1.201.54 and your database schema version might be 8.1.201.25.

- (Oracle only) Run all of the database upgrade scripts from the previous version. To determine if you have to apply any database scripts, check the `resources/sql_scripts` folder in the target directory of the installation.

  For example, if you have release 8.1.201.25 running and you intend to upgrade to release 8.1.300.XX, you must execute the following SQL scripts:

  - `gax_core_upgrade_db_8.1.201.25_to_8.1.300.XX_ora.sql`
  - (For Solution Deployment only) `gax_asd_upgrade_db_8.1.201.15_to_8.1.300.XX_ora.sql`
  - (For Operational Parameter Management only) `gax_opm_upgrade_db_8.1.201.15_to_8.1.300.XX_ora.sql`

- (Microsoft SQL only) Run all of the database upgrade scripts from the previous version. To determine if you have to apply any database scripts, check the `resources/sql_scripts` folder in the target directory of the installation.

  For example, if you have release 8.1.201.25 running and you intend to upgrade to release 8.1.300.XX, you must execute the following SQL scripts:

  - `gax_core_upgrade_db_8.1.201.25_to_8.1.300.XX_mssql.sql`
  - (For Solution Deployment only) `gax_asd_upgrade_db_8.1.201.15_to_8.1.300.XX_mssql.sql`
  - (For Operational Parameter Management only)

```
gax_opm_upgrade_db_8.1.201.15_to_8.1.300.XX_mssql.sql
```

Notes:

- Role privileges must be renewed if the application type is changed. Genesys stores role privileges that are associated with the application type to which they apply, but since GAX is associated with `Genesys Administrator Server` in 8.1.1 releases of Management Framework (for GAX 8.1.2 and higher), not `Genesys Generic Server`, the role privileges must be set using the correct type.
- Files that have version numbers prior to the ones from which you upgraded do not have to be executed. You must log in to the database schema as a GAX user and run the commands inside the SQL scripts as commands for the database.
- If you are installing GAX for the first time or upgrading from release 8.1.x to 8.1.2, when you execute the SQL upgrade scripts, make sure that the scripts are properly committed. If your client application has auto-commit switched off, you might have to add the following line(s) to the scripts:
  - For Oracle: `commit;`
  - For MS SQL: `BEGIN TRANSACTION;COMMIT TRANSACTION;`

- (Optional) You can delete the previous GAX `Application` object after you have verified that the new release is working correctly; however, you can choose to maintain both versions simultaneously.

End

Install GAX

Refer to the Installing Genesys Administrator Extension server on a Linux host or the Installing Genesys Administrator Extension server on a Windows Server 2008 host.

Customizing the GAX Homepage

When Genesys Administrator Extension is launched, it opens to the default homepage view. The default view is a placeholder that you can customize to suit your business needs.

The homepage is an HTML document (`home.html`) and a style sheet (`home.css`) that are stored in the following location after you install GAX: `<tomcat-installation-dir>\webapps\gax\plugins\gax-core\home\`

The file `home.html` is a document fragment. It does not contain all of the standard HTML tags. The default, temporary content is the following:

```
<div class="home-container">
    <h1>Welcome to ${GENESYS_ADMINISTRATOR_EXTENSION}</h1>
    <p>This is a placholder page for the Home module. You can customize it by editing
home.html and home.css.</p>
</div>
```

You can change the contents of this page to completely to suit your requirements.

The style sheet file can also be modified to suit your requirements. The default contents are as follows:

```
.home-container {
    padding: 16px;
    height: 400px;
    background-image: url(i/background.jpg);
    background-repeat: no-repeat;
}Genesys recommends that you use a class prefix like "<tt>home-</tt>" to prevent clashes with
class names that are used elsewhere within GAX.
```

The images that are referenced by the CSS file are in the folder that is named "`i`" in the same folder as `home.css`. You can store as many image files as you require in this folder. Reference your images in the CSS file.

After you edit the `home.html` file, click `Refresh` in the GAX interface to display your updates.

Cleaning the GAX Database After a Tenant is Deleted

If a tenant has been deleted from your environment, some data from that tenant might not be removed from the GAX database.

For more information on how to clean the GAX database after a tenant is deleted, please contact Genesys Technical Support.

Accessing Genesys Administrator Extension

This chapter describes how to log in to, and out of, Genesys Administrator Extension. It also explains how to set User Preferences and System Preferences.

This chapter contains the following sections:

- Logging In
- Logging In to Genesys Administrator from GAX
- Logging Out
- Preferences

Logging In

The Genesys Administrator Extension web-based interface runs on a web application server. It is loaded into your browser each time that you open the website where you installed Genesys Administrator Extension. You then log in.

> **Important**
>
> Genesys Administrator Extension supports the use of blank passwords only if Configuration Server is configured to allow blank passwords. Refer to the Genesys 8.0 Security Deployment Guide for information about using blank passwords.

Logging in to Genesys Administrator Extension

Prerequisites

- Configuration DB Server and Configuration Server are installed and running.
- An instance of a Genesys Administrator Extension `Application` object, configured as a server in Step 4 of Creating the necessary configuration objects for Genesys Administrator Extension, is connected to Configuration Server and running.
- Your browser and its windows are set to a resolution of 1024x768 or greater. If you are working in 1024x768, maximize the browser.
- The user logging in must have Read permission to their own `User` object and Read and Execute permissions on the Genesys Administrator Extension client object. Refer to the Genesys 8.0 Security Deployment Guide for information about permissions. Genesys Administrator Extension respects read-write permissions that are set for Environments and Tenants. You can only access those objects that you have permission to see.

Start

1. Start GAX by using Genesys Administrator.
2. Navigate to the `Application` object for the GAX instance that you intend to start/log in to.
3. Start the application by using the `Start` button in the icon bar.
4. Open a web browser.
5. Enter the following URL in the address bar of the browser:

   ```
   http://<Host name>:8080/gax/
   ```

   where <Host name> is the name of the computer on which you installed Genesys Administrator Extension. The port number is the port that was defined when setting up Tomcat in Installing Tomcat.
6. Log in to Genesys Administrator Extension with your assigned user name and password, and click `Log in`.

> **Important**
>
> Each instance of Genesys Administrator Extension is associated with a single instance of Management Framework; Configuration Server and Port selection is not required during login, nor is it possible to select it.

If you get a permissions error, refer to Required Permissions for instructions.

Your login name and the tenant to which you are logged in is displayed in the top Header Bar of the Genesys Administrator Extension window. The time of your last login is displayed in the Preferences menu. See Preferences for more information.

> **Important**
>
> The date and time of the local machine and the Management Framework machine must be synchronized for the last login time to be accurate.

7. Your account might be configured to set a new password the first time that you log in, or after a system administrator has reset your password. The `Change Password` dialog box is displayed:
   a. Enter a new password in the `New Password` field.
   b. Enter the same password in the `Confirm Password` field.
   c. Click `OK`.

   > **Important**
   >
   > Please see the Genesys 8.0 Security Deployment Guide for more information about resetting passwords.

End

Logging In to Genesys Administrator from GAX

Note: This functionality is only available in releases 8.1.201.30 and higher.

You can access Genesys Administrator from GAX by using the gax-ga plug-in that is part of the core plug-ins that are installed when you install GAX. Your role, and the credentials that you use to log in to GAX, must enable you to access Genesys Administrator.

Login to Genesys Administrator occurs during your login to GAX (refer to Procedure: Logging in to Genesys Administrator Extension). When you enter your GAX login credentials and click Log in, the following process is started:

1. The Log In button becomes disabled.
2. A progress indicator is displayed.
3. The credentials and role that enable you to log in to GAX and Genesys Administrator are verified by GAX.
4. The GAX interface is displayed.
5. If you are permitted to use Genesys Administrator, a menu that enables you to link to Genesys Administrator is displayed.
6. If you click the Genesys Administrator menu, you can select one of the following views in Genesys Administrator:
   - MONITORING > Environment > Dashboard
   - PROVISIONING > Environment > Applications > New Application
   - DEPLOYMENT > Repository > Installation Packages
   - OPERATIONS > Outbound Contact > Dialing Sessions

   Genesys Administrator is launched in a new browser tab or window. The content that is displayed depends on your privileges and access.
7. If you log out of Genesys Administrator, you can continue to use GAX. If you log out of GAX, you are also logged out of Genesys Administrator.

Corporate Login to Genesys Administrator and GAX

Your account might be configured to log in to GAX from a corporate page instead of directly through the GAX login page. If the gax-ga plug-in is installed in your environment, and you are provisioned to log in to both Genesys Administrator and GAX with the same credentials, when you log in by using a corporate interface, GAX is launched and the Genesys Administrator menu is displayed in the GAX interface.

Logging Out

To log out of Genesys Administrator Extension, use the procedure below.

Logging out of Genesys Administrator Extension

Start

1. Click the `Log Out` button in the Header Bar.

End

Preferences

Genesys Administrator Extension enables you to customize the interface to suit your personal preferences. These preferences take effect each time that you, or anyone using your login credentials, logs in to Genesys Administrator Extension from any browser.

To open the Preferences menu, click `Preferences` (the gear icon) in the top header of the main Genesys Administrator Extension screen. The menu displays the last time that this user account was logged into Genesys Administrator Extension.

Note: The date and time of the local machine and the Management Framework machine must be synchronized for the last login time to be accurate.

The Preferences menu contains three options:

- User Preferences
- System Preferences
- Genesys Administrator

Notes:

- Some settings in the Preferences menu are available only in releases 8.1.3 or higher.
- Settings in the User Preferences menu take precedence over settings in the System Preferences menu. For example, if the System Preferences language setting is `English (US)` and the User Preferences language setting is different, Genesys Administrator Extension will use the User Preferences language setting.

User Preferences

Advanced

On the Advanced window, you can specify the logging level for Genesys Administrator Extension JavaScript logging. Set this only if instructed to do so by support personnel. Use the drop-down list to set the level to one of the following:

- `Debug`—All (error, warning, info, and debug) logs are generated.
- `Info`—Error, warning, and info logs are generated.
- `Warning`—Only error and warning logs are generated.

- `Error`—Only error logs are generated.
- `Off`—Logging is disabled.

Note: These logs can be viewed in the browser console. Do not confuse them with Tomcat logs.

Locale

In the Locale window, you can set the following preferences by selecting the appropriate radio button:

- Language—The language to use in the GAX user interface. The default is `English (US)`. You can add more language options by installing localization kit plug-ins. **Note**: A browser refresh is required for the changes to take effect.
- `Date Format`—The format in which dates are to be displayed in Genesys Administrator Extension.
- `Start of Week`—The day on which you consider the week to start, either Sunday or Monday.
- `Number Format`—The format in which numbers are to be displayed.
- `Time Zone`—The time zone in which times are displayed in GAX.

Notes:

- The Time Zone setting in the User Preferences menu applies only for this user account. To set the time zone used by the system, see Locale in the System Preferences menu.
- If no time zone is set in User Preferences, the time zone is determined by the system setting (see Locale in the System Preferences menu). If no system setting exists for time zone, the local client's time zone setting is used.

Reporting

Note: This section only applies to GAX 8.1.301 releases or lower. In GAX 8.1.310 releases or higher, License Usage Reporting functionality is provided by the License Reporting Manager (LRM) plug-in for GAX.

On the Reporting window, you specify how Genesys Administrator Extension paginates License Usage Reports, in terms of the number of lines of data that are displayed per page. You can use the `Prev` and `Next` buttons to page through a report if the report length is greater than the value set in Page `Size`. You can change this value at any time, and it will apply to the next report that you generate.

System Preferences

Throttling

Genesys Administrator Extension enables you to throttle the number of simultaneous changes that are sent to Configuration Server. You can optimize these settings to help ensure consistent performance across your Genesys environment.

Change the `Bulk Update Batch Size` field to specify the number of bulk updates for configuration objects that can be executed simultaneously. The default value is `300`. A value of `0` indicates that there will be no throttling of changes for configuration objects (all requested operations will be sent to Configuration Server without delay). You can enter `0` or any positive integer in this field.

	Note: The maximum Bulk Update Batch Size for users entering from Genesys Administrator is `300`.

Change the `Bulk Update Batch Timeout` field to specify the time interval (in seconds) that Genesys Administrator Extension should wait between executing bulk update operations. The default value is 1. A value of `0` indicates there will be no delay between bulk update operations. You can enter any value between `0` and `300` in this field.

Locale

In the Locale window, you can set the following preferences by selecting the appropriate radio button:

- `Language`—The language to use in the GAX user interface. The default is `English (US)`. You can add more language options by installing localization kit plug-ins. **Note**: A browser refresh is required for the changes to take effect.
- `Date Format`—The format in which dates are to be displayed in Genesys Administrator Extension.
- `Start of Week`—The day on which you consider the week to start, either Sunday or Monday.
- `Number Format`—The format in which numbers are to be displayed.
- `Time Zone`—The time zone in which times are displayed in GAX.

Genesys Administrator

Click the `Launch Genesys Administrator` link in the Preferences menu to launch the Genesys Administrator application. This link is displayed if you are configured to log in to Genesys Administrator, when you log in to Genesys Administrator Extension.

Troubleshooting

Follow the suggestions in this chapter if your Genesys Administrator Extension installation does not seem to work correctly.

This chapter contains the following sections:

- Required Permissions
- Running Out of Memory
- Tomcat Issues
- Browser Issues
- License Usage Reports Not Available

Required Permissions

Access to Genesys Administrator Extension and its functionality is protected by user permissions and Role-Based Access Control. If you get a permissions error when you try to log in to Genesys Administrator Extension or use any of its functionality, you probably do not have the appropriate permissions or role privileges.

An example of a required permission is this: a Tenant user must have write (`Create`) permission on his or her own `User` object to save his or her User Preferences in Genesys Administrator Extension.

Refer to the Genesys 8.0 Security Deployment Guide for more information about permissions and Role-Based Access Control, including how to set up appropriate permissions and role privileges.

Running Out of Memory

If you are working with a large amount of data, such as deploying large or multiple Solutions with Solution Deployment, the installation process might fail with one or both of the following indicators:

- In the gax.log, the following entry:

  ```
  lang.OutOfMemoryError: Java heap space
  ```

- In the Genesys Administrator Extension interface, on the Solutions Packages screen, there might be an error message similar to:

  ```
  Error while fetching lists of dns. There has been a server error.
  ```

This error is caused when the Java heap space is not large enough to support the current process. The default size of the heap is 64 MB. In the default installation, the heap size is set to 1024 MB (the Tomcat default is only 64 MB). If you still need to increase the memory assigned to Tomcat, do so by editing the `$CATALINA_HOME/bin/setenv.sh` (Linux) file or `$CATALINA_HOME/bin/setenv.bat` (Windows) file and adjusting the memory value.

However, if you still see these errors, increase the size of the heap as necessary.

Tomcat Issues

If you encounter problems with your Tomcat host, you can try the following to determine and resolve the problem:

- From the Tomcat host, ping Configuration Server and Solution Control Server by name and by IP address.
- From Solution Control Server, ping the Tomcat Host by name and by IP address.
- From Solution Control Server, telnet to the Tomcat host on all ports, disabling SELinux or any firewalls if necessary.
- A dedicated Tomcat startup script for Genesys Administrator Extension sets the environment variable `GAX_CMD_LINE_ARGS`. To check if this variable has been created correctly, use `gax_startup.sh` and pass parameters using the command line, or use Solution Control Interface or Genesys Administrator.
- Check that Database Access Points are configured and connected.
- Check that the `ojdbc6.jar` file (for Oracle) or `jtda-<version>.jar` file (Microsoft SQL Server) has been copied into the Tomcat `lib` directory.
- Check that gzip compression is enabled in Tomcat for responses.

> **Note**: The following information applies to Genesys Administrator Extension release 8.1.2 and lower:
>
> When the Tomcat process which hosts the GAX application on Windows runs under the SYSTEM account, UNC paths and mapped drives specified for the asd_repository cannot be accessed due to insufficient permissions. Ensure that Tomcat is run as a user possessing sufficient permissions to access UNC paths and mapped drives. Also note that if Genesys Administrator is used to start or stop GAX, the LCA process should also be run under a non-SYSTEM account.

Ports in Use

The table below shows the typical ports used in a Genesys environment.

Typical Ports Used

Port	Description
22	Remote shell (ssh) connections
80	Webserver; can only be used by Tomcat if it is started from the root.
8080	Web server; any user starting Tomcat may use this
1521	Oracle database connections
1433	Microsoft SQL Server

Port	Description
4999	Local Control Agent
5000	Genesys Deployment Agent (GDA)

Browser Issues

If the download of Audio Resource Files, encoded files, and other GAX downloads are blocked by the Microsoft Internet Explorer 8 or 9 information bar, and, after you confirm the download, you are redirected to the main page and then have to repeat the download request, you can adjust your browser settings to prevent this scenario.

This issue is not GAX-specific; it is related to your Internet Explorer settings. To prevent Internet Explorer from blocking your GAX downloads, you must disable the download information bar for GAX downloads.

There are two approaches that you can take to solve this issue:

- Configuring Internet Explorer to allow all downloads without warnings
- Configuring Internet Explorer to allow GAX downloads without warnings

Configuring Internet Explorer to allow all downloads without warnings

Purpose

- To prevent the information bar from blocking GAX file and software downloads.

This procedure disables the Information bar for all downloads. You will be able to download GAX files without being blocked; however, other content will also now be downloaded without warnings.

Start

1. Launch Internet Explorer.
2. Click `Tools`.
3. Select `Internet Options`.
4. Select the `Security` tab.
5. Click `Custom level`.
6. Scroll to the `Downloads` section of the list.
7. Under `Automatic prompting for file downloads`, click `Enable`.
8. Click `OK`.
9. Click `Yes` to confirm that you want to make the change.
10. Click `OK`.

    The Information bar for file downloads is now turned off. You can download GAX files without being blocked by Internet Explorer.

End

Configuring Internet Explorer to allow GAX downloads without warnings

Purpose

- Adjust the settings of Internet Explorer to make it less restrictive when you want to download GAX files.

This procedure enables you to maintain your security settings when you download files from the internet, while making GAX a trusted site from which all your GAX files are downloaded without warnings in the Internet Explorer information bar. You can choose to run with the security level set to High.

Start

1. Launch Internet Explorer.
2. Open the GAX site URL.
3. Click `Tools`.
4. Select `Internet Options`.
5. Click the `Security` tab.
6. Click `Trusted sites`.
7. Click `Sites`.
8. In the `Add this website to the zone` field, verify that the GAX URL is displayed. If not, enter the website in the field. It is not necessary to include the port number.
9. Click `Add`.
10. De-select `Require server verification (https:) for all sites in this zone`.
11. Click `Close`.
12. Click `Custom level`.
13. Scroll to the `Downloads` section of the `Settings` list.
14. Under `Automatic prompting for file downloads`, click `Enable`.
15. Scroll to the `Scripting` section of the `Settings` list.
16. Under `Active Scripting`, click `Enable`.
17. Click `OK`.
18. Click Yes to confirm that you want to make the change.
19. In the `Internet Options` window, click `OK`.

    The Information bar warnings for file downloads is now turned off for trusted sites only, and GAX is set as a trusted site.

End

License Usage Reporting Report Download Issues with Internet Explorer 9

Note: This section only applies to GAX 8.1.301

	releases or lower. In GAX 8.1.310 releases or higher, License Usage Reporting functionality is provided by the License Reporting Manager (LRM) plug-in for GAX.

A specific configuration option setting in Microsoft Internet Explorer 9 on Windows Server systems might prevent the exporting of License Usage Reporting reports.

The advanced option `Do not save encrypted pages to disk` , which is set by default, prevents the saving of encrypted pages from HTTPS connections.

Use the following procedure to enable the download of encrypted files by using Internet Explorer 9:

Enabling download of License Usage Reporting reports from Internet Explorer 9

Purpose

- To enable the downloading of License Usage Reporting reports by using Microsoft Internet Explorer 9 in a Windows Server system.

Start

1. Launch Microsoft Internet Explorer 9.
2. Click `Tools`.
3. Select `Internet Options`.
4. Select the `Advanced` tab.
5. Under the `Security` heading, uncheck `Do not save encrypted pages to disk`.
6. Click `OK`.
7. Close all open Internet Explorer 9 windows.
8. Restart Internet Explorer 9.

End

Audio Resource File Playback Issue in Internet Explorer 8.x

Users of Internet Explorer 8 cannot play Audio Resource Files in Genesys Administrator Extension. This is due to the lack of support for HTML5 in Internet Explorer 8.

To play Audio Resource Files, use a newer version of Internet Explorer or another supported browser. See Browser Requirements for more information.

License Usage Reports Not Available

	Note: This section only applies to GAX 8.1.301 releases or lower. In GAX 8.1.310 releases or higher, License Usage Reporting functionality is provided by the License Reporting Manager (LRM) plug-in for GAX.

If there are no on-demand License Usage Reporting reports available, it might be because there is no data available from which the reports are generated. Assuming that the systems are otherwise operating normally, it is probably because the automatic process for collecting the data from local LRM databases did not run as scheduled. In this case, ask the LRM Administrator to restart LRM.

You might experience an issue when you try to export a License Usage Reporting report by using Microsoft Internet Explorer 9 on a Windows Server system. Refer to Enabling download of License Usage Reporting reports from Internet Explorer 9 to fix this issue.

Role Privileges

This Appendix describes the role privileges that are available and enforced by Genesys Administrator Extension. The privileges are in a hierarchy based on the modules in Genesys Administrator Extension. They are organized in this Appendix as follows:

- General
- GA Direct Login Integration
- Operational Parameter Management
- Solution Deployment
- Account Management
- Audio Resources Management-Tenant
- Audio Resources Management-System
- License Usage Reporting

To view these privileges on the `Role Privileges` tab of a `Role` object in Genesys Administrator, make sure that `Genesys Administrator Extension` is checked in the `Add/Remove Products` section on the tab.

	Note: You must have Genesys Administrator 8.1.2, or later, installed when you upload the Genesys Administrator templates to Configuration Server and assign roles. The new template type, 184 (Genesys Administrator Server), is not recognized by earlier versions of Genesys Administrator; therefore, role assignments will not be functional.

For more information about role privileges specifically, and Role-Based Access Control in general, refer to the Genesys 8.0 Security Deployment Guide.

General

The following privileges apply to Genesys Administrator Extension.

Prerequisites

None

Role Privileges

`View Audit History Data`	Enables users to read privilege auditing history information.
`Read Plug-ins`	Enables users to read nodes and plug-ins.
`Write Plug-ins`	Enables users to enable or disable plug-ins, and also enables users to modify plug-in options.

GA Direct Login Integration

The following privileges apply to Genesys Administrator Extension.

Prerequisites

None

Role Privileges

`GA Direct Login Integration`	User privilege to access Genesys Administrator directly from GAX without re-entering credentials. Prerequisites: None.

Operational Parameter Management

Operational Parameter Management role privileges control what tasks a user can do in the Operational Parameter Management module of Genesys Administrator Extension.

Prerequisites

None

Role Privileges

`Read Parameters`	Allows a user to view Operational Parameters for OPM. Prerequisites: None.
`Write Parameters`	Allows a user to create, update, and delete Operational Parameters for OPM. Prerequisites: `Read Parameters`.
`Read Group Templates`	Allows a user to view Parameter Group Templates. Prerequisites: `Read Parameters`.
`Write Group Templates`	Allows a user to create, update, and delete Parameter Group Templates. Prerequisites: `Read Group Templates`.
`Read Parameter Groups`	Allows a user to view Parameter Groups. Prerequisites: None.
`Update and Delete Parameter Groups`	Allows a user to update or delete Parameter Groups. Prerequisites: `Read Parameter Groups`.
`Deploy and Re-associate Parameter Groups`	Allows a user to deploy or re-associate Parameter Groups. Prerequisites: `Read Group Templates` and `Read Parameter Groups`.

Solution Deployment

Solution Deployment role privileges control what tasks a user can perform in the Solution Deployment module of Genesys Administrator Extension.

Prerequisites

None

Role Privileges

`Delete IPs and SPDs`	Delete privilege for IPs and SPDs of ASDs. Prerequisite: `Read Deployable IPs and SPDs`.
`Deploy IPs`	Deploy privilege for IPs of ASDs. Prerequisite: Read `Deployable IPs and SPDs`.
`Deploy SPDs`	Deploy privilege for SPDs of ASDs. Prerequisite: `Read Deployable IPs and SPDs`.
`Read Deployable IPs and SPDs`	Read privilege for marked IPs and SPDs of ASDs.
`Read Deployed IPs and SPDs`	Read privilege for deployed IPs, SPDs, and audit logs of ASDs. Prerequisites: None.
`Replace IPs and SPDs`	Enables a user to upload another version of an IP or SPD and replace the version that is already in the database.
`Upload IPs and SPDs`	Create privilege for IPs and SPDs of ASDs. Prerequisite: `Read Deployable IPs and SPDs`.
`Write IPs and SPDs`	Write privilege for IPs and SPDs of ASDs. Enables the copy and move operations. Prerequisite: Read `Deployable IPs and SPDs`.

Account Management

Account Management role privileges control what tasks a user can perform in the Account Management module of Genesys Administrator Extension.

Prerequisites

None

Role Privileges

`Administer Users`	Allows a user to read and update the Force Password Reset on Next Login option in the User Accounts section. It also allows access to the User Options, Access Control, and Accessible Objects panels. Prerequisite: `Write Users`. **Notes:** • The `Force Password Reset on Next Login` option only displays if Genesys Administrator Extension connects to Management Framework version 8.1.1 and above. • Please see the *Genesys 8.1 Security Deployment Guide* for more information about resetting passwords.
`Read Agent Information`	Allows a user to access the `Agent Information` function and to view agent information in the User Accounts section. Prerequisites: None.
`Read Users`	Allows a user to access the User Accounts details pane, except for `Force Password Reset on Next Login`, `User Options`, `Access Control`, `Accessible Objects`, and `Agent Information`. Prerequisites: None.
`Write Agent Information`	Allows a user to create and update all values on the User Accounts details pane for agents. Prerequisite: `Read Agent Information`.
`Write Users`	Allows a user to create and update all values on the User Accounts details pane except for Force `Password Reset on Next Login`, `User Options`, `Access Control`, `Accessible Objects`, and `Agent Information`. Prerequisite: Read `Users`.
`Administer Roles`	Allows a user to access the `User Options` and `Access Control` buttons. Prerequisite: `Write`

	Roles.
Read Roles	Allows a user only to read Roles. The User Options and Access Control buttons are not displayed. Prerequisite: None.
Write Roles	Allows a user to create, update, and delete Roles. The User Options and Access Control buttons are not displayed. Prerequisite: Read Roles.
Administer Skills	Allows a user to access the User Options and Access Control buttons. Prerequisite: Write Skills.
Read Skills	Allows a user only to read Skills. The User Options and Access Control buttons are not displayed. Prerequisite: None.
Write Skills	Allows a user to create, update, and delete Skills. The User Options and Access Control buttons are not displayed. Prerequisite: Read Skills.
Administer Agent Groups	Allows a user to access the User Options and Access Control buttons. Prerequisite: Write Agent Groups.
Read Agent Groups	Allows a user only to read Agent Groups. The User Options and Access Control buttons are not displayed. Prerequisite: None.
Write Agent Groups	Allows a user to create, update, and delete Agent Groups. The User Options and Access Control buttons are not displayed. Prerequisite: Read Agent Groups.
Administer Access Groups	Allows a user to access the User Options and Access Control buttons. Prerequisite: Write Access Groups.
Read Access Groups	Allows a user only to read Access Groups. The User Options and Access Control buttons are not displayed. Prerequisite: None.
Write Access Groups	Allows a user to create, update, and delete Access Groups. The User Options and Access Control buttons are not displayed. Prerequisite: Read Access Groups.
Administer Capacity Rules	Allows a user to access the User Options and Access Control buttons. Prerequisite: Write Capacity Rules.
Read Capacity Rules	Allows a user only to read Capacity Rules. The User Options and Access Control buttons are not displayed. Prerequisite: None.
Write Capacity Rules	Allows a user to create, update, and delete Capacity Rules. The User Options and Access Control buttons are not displayed. Prerequisite: Read Capacity Rules.

Audio Resources Management—Tenant

Audio Resource Management—Tenant role privileges control what tasks a user can perform at the Tenant level in the Audio Resource Management module of Genesys Administrator Extension.

Prerequisites

None

Role Privileges

`Write Audio Resources`	Allows a user to create, update, and delete Audio Resources and the Audio Resource Files that they contain. Prerequisites: `Read Audio Resources` and `Read Personalities`.
`Write Personalities`	Allows a user to create, update, and delete Personalities for Audio Resources and their files. Prerequisite: `Read Personalities`.
`Process Audio Resources`	Allows a user to initiate re-encoding of Audio Resources and re-transferring them to target storage. Prerequisites: `Read Audio Resources` and `Read Personalities`.
`Read Audio Resources`	Allows a user to view Audio Resources and the Audio Resource Files that they contain. Prerequisite: None.
`Read Personalities`	Allows a user to view Personalities for Audio Resources and their files. Prerequisite: None.

Audio Resources Management—System

Audio Resource Management—System role privileges control what tasks a user can perform at the Solution Provider level in the Audio Resource Management module of Genesys Administrator Extension.

Prerequisites

None

Role Privileges

`Deploy Audio Resources`	Allows a user to deploy Audio Resources and the Audio Resource Files that they contain from the System Provider to Tenants. Prerequisites: Read `Audio Resources` and `Read Personalities`. This privilege is effective only if it is granted to a user in the Environment Tenant. Users in other Tenants are unable to deploy Audio Resources even if they are granted this privilege.

License Usage Reporting

	Note: This section only applies to GAX 8.1.301 releases or lower. In GAX 8.1.310 releases or higher, License Usage Reporting functionality is provided by the License Reporting Manager (LRM) plug-in for GAX.

License Usage Reporting role privileges control what tasks a user can perform in the License Usage Reporting module of Genesys Administrator Extension.

Prerequisites

None

Role Privileges

`Generate Tenant Report`	Allows a user to access the License Usage Reporting module and generate a System report.
`Generate System-wide Report`	Allows a user to access the License Usage Reporting module and generate a Tenant report.
`Manage Tenant Provisioned Count`	Privilege to manage the tenant provisioned count for LUR. Prerequisites: None.

Configuration Options

This appendix describes the configuration options for Genesys Administrator Extension, and contains the following sections:

- Mandatory Options
- general Section
- asd Section
- arm Section
- ga Section
- log Section
- opm Section

Note: There are no configuration options required for the Operational Parameters Management and License Usage Reporting modules in Genesys Administrator Extension.

Setting Configuration Options

Use Genesys Administrator to set Genesys Administrator Extension configuration options. Unless specified otherwise, set Genesys Administrator configuration options in the `Options` of the Genesys Administrator Extension `Application` object with which Genesys Administrator was deployed (refer to the Framework 8.1 Genesys Administrator Deployment Guide). Use one of the following navigation paths:

- Genesys Administrator Extension server—Genesys Administrator Extension `Application` object of type `Generic Genesys Server` > `Options` tab > `Advanced View (Options)`
- Genesys Administrator Extension client—Genesys Administrator Extension `Application` object of type `Configuration Manager` > `Options` tab > `Advanced View (Options)`

Warning! Configuration section names, configuration option names, and predefined option values are case-sensitive. Type them in Genesys Administrator exactly as they are documented in this appendix.

Mandatory Options

You do not have to configure any options to start Genesys Administrator Extension.

general Section

This section must be called `general`, and is configured in the Genesys Administrator Extension Server `Application` object of type `Generic Genesys Server`.

The options in this section are required for the general behavior of Genesys Administrator Extension.

auditing

- Default Value: `true`
- Valid Values: `true`, `false`
- Changes Take Effect: After Genesys Administrator Extension is restarted.
- By default GAX is set to audit all actions that are performed by users. Set to `false` if auditing is not required.

client_app_name

- Default Value: `default`
- Valid Values: The valid name of an application object of type Configuration Manager.
- Changes Take Effect: After Genesys Administrator Extension is restarted.
- Specifies the name of the client application. GAX requires a client application object to enable access control of the browser-based interface.

confserv_timeout

- Default Value: `30`
- Valid Values: The value of the timeout protocol.
- Changes Take Effect: Immediately.

Protocol timeout value for connections to Configuration Server.

default_account_dbid

- Default Value: `100`
- Valid Values: The database ID of the default account. A valid DBID that represents the person object that should be used as the default account (refer to Default Account Support).
- Changes Take Effect: After Genesys Administrator Extension is restarted.
- The DBID that is assigned to default account. This DBID can be set to any DBID of any valid user. The user with the specified DBID will have all role privileges.
- If this option is not set, GAX uses the value `100` for the DBID. The default account is identified by DBID. The default value for the DBID is `100`. If the default account is deleted and recreated, it will be assigned a new DBID. Use the `default_account_dbid` option to specify the DBID of the default account if the

value is not 100.

inactivity_timeout

- Default Value: 600
- Valid Values: Any integer value.
- Changes Take Effect: After Genesys Administrator Extension is restarted.
- Specifies the value of the inactivity timeout in seconds. A negative value deactivates this timer.

msgsrv_attempts

- Default Value: 1
- Valid Values: Any positive integer value greater than 0.
- Changes Take Effect: After Genesys Administrator Extension is restarted.
- Specifies the number of connection attempts that will be made until GAX tries to connect to the backup Message Server.

msgsrv_max_switchovers

- Default Value: -1
- Valid Values: Any integer value.
- Changes Take Effect: After Genesys Administrator Extension is restarted.
- Specifies the number of switch-overs between Message Servers before GAX gives up trying to reconnect. 0 specifies no reconnection attempts. A negative values specifies unlimited reconnection attempts.

msgsrv_timeout

- Default Value: 10
- Valid Values: Any positive integer value.
- Changes Take Effect: After Genesys Administrator Extension is restarted.
- Specifies the protocol timeout value for connections to Message Server.

msgsrv_warmstandby_timeout

- Default Value: 60
- Valid Values: Any integer value.
- Changes Take Effect: After Genesys Administrator Extension is restarted.
- The time in seconds between reconnection attempts to Message Server.

scs_attempts

- Default Value: `1`
- Valid Values: Any positive integer value greater than `0`.
- Changes Take Effect: After Genesys Administrator Extension is restarted.
- Specifies the number of connection attempts that will be made until GAX tries to connect to the backup Solution Control Server.

scs_max_switchovers

- Default Value: `-1`
- Valid Values: Any integer value.
- Changes Take Effect: After Genesys Administrator Extension is restarted.
- Specifies the number of switch-overs between Solution Control Servers before GAX gives up trying to reconnect. 0 specifies no reconnection attempts. A negative values specifies unlimited reconnection attempts.

scs_timeout

- Default Value: `10`
- Valid Values: Any positive integer value.
- Changes Take Effect: After Genesys Administrator Extension is restarted.
- Specifies the protocol timeout value for connections to Solution Control Server.

scs_warmstandby_timeout

- Default Value: `60`
- Valid Values: Any integer value.
- Changes Take Effect: After Genesys Administrator Extension is restarted.
- The time in seconds between reconnection attempts to Solution Control Server.

session_timeout

- Default Value: `900`
- Valid Values: Any positive integer value.
- Changes Take Effect: After Genesys Administrator Extension is restarted.
- The time in seconds after the session timeout that the GAX user session on the server will be destroyed.

asd Section

This section must be called `asd`, and is configured in the Genesys Administrator Extension Server `Application` object of type `Generic Genesys Server`.

The options in this section are required for the Solution Deployment module in Genesys Administrator Extension.

local_ip_cache_dir

- Default Value: None
- Valid Values: Any valid folder
- Changes Take Effect: After Genesys Administrator Extension is restarted.

Specifies the folder where the IP used for the deployment is cached. Caching the IP reduces deployment time if the IP is reused. This option must be set to a UNC path or a local path that points to a directory that can be accessed (with read\write permissions) from the machine that is running the Genesys Administrator Extension server.

local_template_dir

- Default Value: `{CATALINA_HOME}/webapps/gax/WEB-INF/classes/xmltemplates`
- Valid Values: Any valid folder
- Changes Take Effect: After Genesys Administrator Extension is restarted.
- Specifies the local directory where component templates are stored. The default value is sufficient unless the directory has been changed after installing Genesys Administrator Extension.

Note: This option is for users of Genesys Administrator Extension 8.1.2 only.

repository_path

- Default Value: `/opt/gax/I`
- Valid Values: Any valid path
- Changes Take Effect: After Genesys Administrator Extension is restarted.
- Specifies the path where the Genesys Administrator IP Repository can be accessed. It should point to the directory that contains the file `repository.xml`, as defined in Step 3 of Installing Samba.

Note: This option is for users of Genesys Administrator Extension 8.1.2 only.

silent_ini_path

- Default Value: `{CATALINA_HOME}/webapps/gax/WEB-INF/classes/xmltemplates/ ga_default/ genesys_silent_ini.xml`
- Valid Values: Any valid path and XML file name
- Changes Take Effect: After Genesys Administrator Extension is restarted.

Specifies the name of the silent installation folder used by ASD. The default value is sufficient unless the path or file has been changed after installing Genesys Administrator Extension.

arm Section

This section must be called `arm`, and is configured in the Genesys Administrator Extension Server Application object of type `Generic Genesys Server`.

The options in this section are required for the Audio Resource Management module in Genesys Administrator Extension.

delete_from_db_after_processing

- Default Value: `false`
- Valid Values: `false`, `true`
- Changes Take Effect: After Genesys Administrator Extension is restarted.
- Specifies if the original audio file is to be deleted from the database after all required processing (including any format conversion and transfer to target storage) is complete. If set to `true`, the original file located in the target storage is used for any subsequent reprocessing, and if required, is downloaded from the target storage rather than from the database (from which it was removed).

This option enables the user to decide if he or she wants the system to delete the binary audio information in the original audio file from the database after processing is done. The advantage of deleting the information is that less database space is used. The disadvantage is that reprocessing is possible on the files located in target storage. These files could be subject to corruption, loss, or a problem with the target storage itself, thereby losing the original information. In this case, the database just offers redundancy and robustness of the data.

local_announcement_folder

- Default Value: `announcement`
- Valid Values: Any valid folder
- Changes Take Effect: After Genesys Administrator Extension is restarted.
- Specifies the name of the folder where the audio data for audio resources of type Announcement is stored while the audio resource is stored in the database, encoded, and moved to target storage. This folder is specified relative to the path specified by the option local_path.

local_music_folder

- Default Value: `music`
- Valid Values: Any valid folder
- Changes Take Effect: After Genesys Administrator Extension is restarted.
- Specifies the name of the folder where the audio data for audio resources of type Music is stored while the audio resource is stored in the database, encoded, and moved to target storage. This folder is specified relative to the path specified by the option local_path.

local_os

- Default Value: RHEL5
- Valid Values: `Red Hat Enterprise Linux 5`, `Windows`
- Changes Take Effect: After Genesys Administrator Extension is restarted.
- Specifies the operating system running on the Genesys Administrator Extension server host.

local_path

- Default Value: `/opt/gax/arm`
- Valid Values: Any valid path
- Changes Take Effect: After Genesys Administrator Extension is restarted.
- Specifies the absolute path to the location of local audio storage, that is, to the folders specified by the options target_announcement_folder and local_music_folder.

local_sox_path

- Default Value: `/usr/bin/sox`
- Valid Values: Any valid path
- Changes Take Effect: After Genesys Administrator Extension is restarted.
- Specifies the absolute path to the SoX binary (executable) file.

target_announcement_folder

- Default Value: `announcement`
- Valid Values: Any valid folder name
- Changes Take Effect: After Genesys Administrator Extension is restarted.
- Specifies the folder where all audio files of type Announcement, both original and encoded, are stored. Media Server retrieves the files from this folder and uses them. This folder is specified relative to the path specified by the option target_path.
- If the delete_from_db_after_processing option is set to `true`, the original audio files stored in this folder are used for reprocessing, and are downloaded from this folder instead of from the database. However, the encoded files are always downloaded from this folder, not from the database.

target_music_folder

- Default Value: `music`
- Valid Values: Any valid folder name
- Changes Take Effect: After Genesys Administrator Extension is restarted.
- Specifies the folder where all audio files of type Music, both original and encoded, are stored. Media Server retrieves the files from this folder and uses them.This folder is specified relative to the path specified by the option target_path.

If the delete_from_db_after_processing option is set to `true`, the original audio files stored in this folder are used for reprocessing, and are downloaded from this folder instead of from the database. However, the encoded files are always downloaded from this folder, not from the database.

target_os

- Default Value: RHEL5
- Valid Values: `Red Hat Enterprise Linux 5`, `Windows`
- Changes Take Effect: After Genesys Administrator Extension is restarted.
- Specifies the operating system running on the target storage host.

target_path

- Default Value: `/mnt/arm/target`
- Valid Values: Any valid path
- Changes Take Effect: After Genesys Administrator Extension is restarted.
- Specifies the absolute path to the location of the folders specified by the options target_announcement_folder and target_music_folder. This path must appear local to the Genesys Administrator Extension server, even though target storage is located on a different host. The path specified here must be served by the ARM Web Proxy server (this is typically the root directory from the perspective of the web server).

ga Section

ga_appName

- Default Value: `default`
- Valid Values: The valid name of the Genesys Administrator application object.
- Changes Take Effect: After Genesys Administrator Extension is restarted.
- Specifies the Application name for Genesys Administrator that is to be used to directly log in to Genesys Administrator from GAX.

ga_host

- Default Value: `""`
- Valid Values: The name of a host or an IP address.
- Changes Take Effect: After Genesys Administrator Extension is restarted.
- Specifies the Genesys Administrator host parameter that enables direct login to Genesys Administrator.

ga_port

- Default Value: `80`
- Valid Values: A valid port ID.
- Changes Take Effect: After Genesys Administrator Extension is restarted.
- Specifies the Application port number for Genesys Administrator that is to be used to directly log in to Genesys Administrator from GAX. This option is mandatory if the Genesys Administrator port number is not `80`.

ga_protocol

- Default Value: `http`
- Valid Values: `http`, `https`
- Changes Take Effect: After Genesys Administrator Extension is restarted.
- Specifies the Genesys Administrator protocol that is required to directly log in to Genesys Administrator from GAX.

log Section

all

- Default Value: `stdout`
- Valid Values:

Value	Description
`stdout`	Log events are sent to the Standard output.
`network`	Log events are sent to Message Server, which can reside anywhere on the network. Message Server stores the log events in the Log Database. Setting the all log level option to the network output enables an application to send log events of the Standard, Interaction, and Trace levels to Message Server. Debug-level log events are neither sent to Message Server nor stored in the Log Database.
`[filename]`	Log events are stored in a file with the specified name. If a path and filename are not specified, the file is created in the application's working directory.

- Changes Take Effect: After Genesys Administrator Extension is restarted.

- 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: `stdout, logfile`

expire

- Default Value: `20`
- Valid Values: Any integer value.
- Changes Take Effect: After Genesys Administrator Extension is restarted.

Specifies the maximum number of log files to be kept.

log

- Default Value: `standard`
- Valid Values:

Value	Description
`all`	All log events (that is, log events of the `Standard`, `Trace`, `Interaction`, and `Debug` levels) are generated.
`debug`	The same as `all`.
`trace`	Log events of the `Trace` level and higher (that is, log events of the `Standard`, `Interaction`, and `Trace` levels) are generated, but log events of the `Debug` level are not generated.
`interaction`	Log events of the Interaction level and higher (that is, log events of the `Standard` and `Interaction` levels) are generated, but log events of the `Trace` and `Debug` levels are not generated.
`standard`	Log events of the `Standard` level are generated, but log events of the `Interaction`, `Trace`, and `Debug` levels are not generated.
`none`	No output is produced.

- Changes Take Effect: After Genesys Administrator Extension is restarted.

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`.

log-cache-size

- Default Value: `16000`
- Valid Values: Any integer value.
- Changes Take Effect: After Genesys Administrator Extension is restarted.

Specifies the maximum number of logs in the log message queue.

segment

- Default Value: `10000`

- Valid Values: Any valid file size.
- Changes Take Effect: After Genesys Administrator Extension is restarted.

Specifies the maximum log file size in kilobytes.

standard

- Default Value: ""
- Valid Values:

Value	Description
`stdout`	Log events are sent to the Standard output.
`network`	Log events are sent to Message Server, which can reside anywhere on the network. Message Server stores the log events in the Log Database.
`[filename]`	Log events are stored in a file with the specified name. If a path and filename are not specified, the file is created in the application's working directory.

- Changes Take Effect: After Genesys Administrator Extension is restarted.
- 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: `stderr, network`

trace

- Default Value: ""
- Valid Values:

Value	Description
`stdout`	Log events are sent to the Standard output.
`network`	Log events are sent to Message Server, which can reside anywhere on the network. Message Server stores the log events in the Log Database.
`[filename]`	Log events are stored in a file with the specified name. If a path and filename are not specified, the

Value	Description
	file is created in the application's working directory.

- Changes Take Effect: After Genesys Administrator Extension is restarted.
- 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: `stderr, network`

opm Section

write_json

- Default Value: `false`
- Valid Values: `true`, `false`
- Changes Take Effect: After Genesys Administrator Extension is restarted.
- Defines whether OPM writes `JSON` data directly to transaction objects in binary form (data is written as value for the key `"_json"`).

Document Change History

This section lists topics that are new or that have changed significantly since the previous release of the Framework 8.1 Genesys Administrator Extension Deployment Guide.

Document Version 8.1.302.00

The following topics have been added or have changed significantly since the 8.1.301.00 release:

- Notes have been added throughout this document to indicate that License Usage Reporting functionality is now provided by the License Reporting Manager plug-in for GAX. This change applies to GAX 8.1.310 releases or higher.
- Setting Up Genesys Administrator Extension:
    - Browser Requirements was modified.
    - Required Permissions and Role Privileges was modified.
    - Step 6 of Installing Tomcat was modified to indicate that users of GAX 8.1.3 releases and higher are not required to download ODBC drivers from Oracle. These drivers are now included with the GAX installation package.
    - A warning was added to Setting up the Genesys Administrator database (for Microsoft SQL Server).
    - Setting up the Genesys Administrator database (for PostgreSQL) was added.
    - A note was added to Installing Samba.
    - Configuring the GAX Database for TLS (PostgreSQL) was added.
    - Step 3 of Upgrading to the latest Genesys Administrator Extension for Management Framework 8.1.1 or higher was modified to indicate that this step applies only to instances that use GAX Application object of type Genesys Generic Server.
    - Upgrading to the latest Genesys Administrator Extension for Management Framework 8.1.1 or higher was updated.
    - Upgrading to the latest Genesys Administrator Extension for Management Framework 8.1.0 or lower was updated.

Document Version 8.1.301.00

The following topics have been added or have changed significantly since the 8.1.201.00 release:

- Genesys Administrator Extension Overview:
    - The section Genesys Administrator Extension was modified to include information about language packs.
    - Solution Deployment was modified.

    - Account Management was added.

- Setting Up Genesys Administrator Extension:
    - Browser Requirements was modified.
    - Installing Tomcat was modified.
    - Enabling UTF-8 character encoding (for Oracle) was added.
    - Managing GAX Compatible Plug-ins was added.
    - Upgrading to the latest Genesys Administrator Extension for Management Framework 8.1.1 or higher was updated.
    - Upgrading to the latest Genesys Administrator Extension for Management Framework 8.1.0 or lower was updated.
- Accessing Genesys Administrator Extension:
    - Preferences was added.
- Troubleshooting Genesys Administrator Extension:
    - Tomcat Issues was updated.
    - Browser Issues was updated.
- Appendix A, "Genesys Administrator Extension Role Privileges was updated throughout.
- Appendix B, "Configuration Options:
    - New configuration options were added to the following sections:
        - general Section
        - asd Section
        - log Section

New and Changed Privileges

The following privileges were added or changed in this release (refer to Appendix A, "Genesys Administrator Extension Role Privileges):

Group: Genesys Administrator Extension - General

New privileges

- `Read Plug-ins`—Enables users to read nodes and plug-ins.
- `Write Plug-ins`—Enables users to enable or disable plug-ins, and also enables users to modify plug-in options.

Group: Genesys Administrator Extension - Solution Deployment

Changed privileges

- `Deploy IPs`—Enables the user to deploy an IP or perform any supported profile, such as a rollback or upgrade;.

Group: Genesys Administrator Extension - Account Management

New privileges

- `Administer Users`—Allows a user to read and update the Force Password Reset on Next Login option in the User Accounts section. It also allows access to the User Options, Access Control, and Accessible Objects panels. Prerequisite: Write Users.

Notes

- The Force Password Reset on Next Login option displays only if Genesys Administrator Extension connects to Management Framework version 8.1.1 and higher.
- For more information about resetting passwords, please see the Genesys 8.0 Security Deployment Guide.

- `Read Agent Information`—Allows a user to access the `Agent Information` function and to view agent information in the User Accounts section. Prerequisites: None.
- `Read Users`—Allows a user to access the User Accounts details pane, except for `Force Password Reset on Next Login`, `User Options`, `Access Control`, `Accessible Objects`, and `Agent Information`. Prerequisites: None.
- `Write Agent Information`—Allows a user to create and update all values on the User Accounts details pane for agents. Prerequisite: `Read Agent Information`.
- `Write Users`—Allows a user to create and update all values on the User Accounts details pane except for `Force Password Reset on Next Login`, `User Options`, `Access Control`, `Accessible Objects`, and `Agent Information`. Prerequisite: `Read Users`.
- `Administer Roles`—Allows a user to access the `User Options` and `Access Control` buttons. Prerequisite: `Write Roles`.
- `Read Roles`—Allows a user only to read Roles. The `User Options` and `Access Control` buttons are not displayed. Prerequisite: None.
- `Write Roles`—Allows a user to create, update, and delete Roles. The `User Options` and `Access Control` buttons are not displayed. Prerequisite: `Read Roles`.
- `Administer Skills`—Allows a user to access the `User Options` and `Access Control` buttons. Prerequisite: `Write Skills`.
- `Read Skills`—Allows a user only to read Skills. The `User Options` and `Access Control` buttons are not displayed. Prerequisite: None.

- Write Skills—Allows a user to create, update, and delete Skills. The User Options and Access Control buttons are not displayed. Prerequisite: Read Skills.
- Administer Agent Groups—Allows a user to access the User Options and Access Control buttons. Prerequisite: Write Agent Groups.
- Read Agent Groups—Allows a user only to read Agent Groups. The User Options and Access Control buttons are not displayed. Prerequisite: None.
- Write Agent Groups—Allows a user to create, update, and delete Agent Groups. The User Options and Access Control buttons are not displayed. Prerequisite: Read Agent Groups.
- Administer Access Groups—Allows a user to access the User Options and Access Control buttons. Prerequisite: Write Access Groups.
- Read Access Groups—Allows a user only to read Access Groups. The User Options and Access Control buttons are not displayed. Prerequisite: None.
- Write Access Groups—Allows a user to create, update, and delete Access Groups. The User Options and Access Control buttons are not displayed. Prerequisite: Read Access Groups.
- Administer Capacity Rules—Allows a user to access the User Options and Access Control buttons. Prerequisite: Write Capacity Rules.
- Read Capacity Rules—Allows a user only to read Capacity Rules. The User Options and Access Control buttons are not displayed. Prerequisite: None.
- Write Capacity Rules—Allows a user to create, update, and delete Capacity Rules. The User Options and Access Control buttons are not displayed. Prerequisite: Read Capacity Rules.

New and Changed Configuration Options

The following configuration options were added or changed in this release (refer to Appendix B, "Configuration Options):

Section: general

This section contains general options for GAX (refer to general Section).

New option

- confserv_timeout—Specifies the timeout value for connections to Configuration Server.

Section: asd

The following option has been added in the asd section (refer to asd Section).

New option

- local_ip_cache_dir—Specifies the local directory where the IP used for the deployment is cached. Caching the IP reduces deployment time if the IP is reused This option must be set to a UNC path or a local path that points to a directory that can be accessed (with read\write permissions) from the machine that is running the Genesys Administrator Extension server.

Section: log

The following options have been added in the `log` section (refer to log Section).

New options

- expire—Specifies the maximum number of log files to be kept.
- log—Determines whether a log output is created.
- log-cache-size—Specifies the maximum number of logs in the log message queue.
- segment—Specifies the max log file size in kilobytes.