Framework Deployment Guide

Management Framework Current

12/21/2023

Table of Contents

Management Framework Deployment Guide

Published: June 8, 2018

9.x This 9.x version of the Management Framework Deployment Guide can apply to component releases from the 9.1.x, 9.0.x, and 8.5.x code streams. For releases of Management Framework prior to August 22, 2017, see the Management Framework home page.

Use this guide to introduce you to the concepts and terminology relevant to the Genesys Framework, and procedures to install, configure, and run Management Framework.

Sharing the Load Systems

Distributed components:

Configuration Server Proxies

Distributed Solution Control Servers

Disaster Recovery/Business Continuity

Disaster Recovery / Business Continuity

Framework Database Replication for DR Using:

Oracle GoldenGate

Microsoft AlwaysOn

Starting and Stopping Framework Components

Using Start Files

Using the Management Layer

Starting Manually

Using Windows Service Manager

Additional Information

Generic Deployment and Login Procedures

Configuration Server History Log

Monitoring Performance of Configuration Layer

Automatic Disconnect Detect Protocol (ADDP)

Internet Protocol version 6 (IPv6)

and more ...

About Genesys Framework

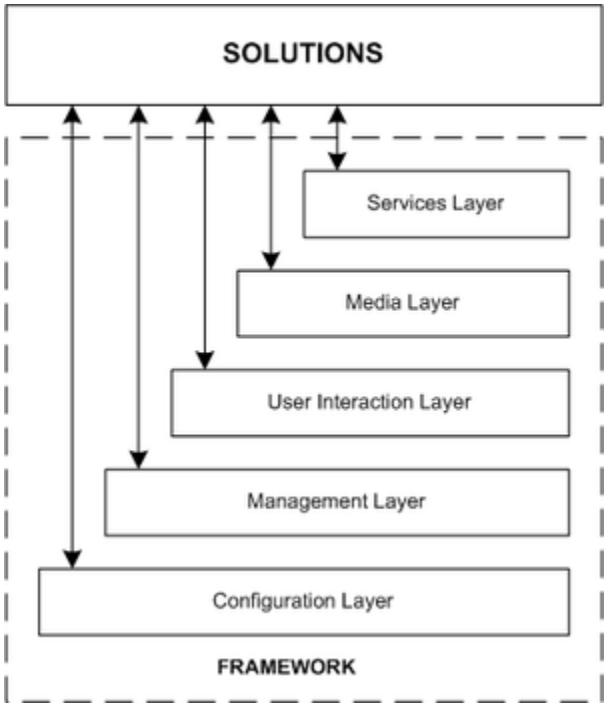
The Genesys Framework is a mandatory part of any Genesys-based interaction management system. It provides functions required for the normal operation of any Genesys solution:

- Configuration centralizes processing and storage of all the data required for Genesys solutions to work within a particular environment.
- Access Control sets and verifies users' permissions for access to, and manipulation of, solution functions and data.
- Solution Control starts and stops solutions and monitors their status.
- Alarm Processing defines and manages conditions critical to the operation of solutions.
- Troubleshooting hosts a user-oriented, unified logging system with advanced storage, sorting, and viewing capabilities.
- Fault Management automatically detects and corrects situations that might cause operational problems in solutions.
- External Interfaces enable communication with a variety of telephony systems and database management systems (DBMS).
- Attached Data Distribution supports the distribution of business data attached to interactions, within and across solutions.

Architecture

The Genesys Framework consists of five layers (see the figure below). In sophisticated configurations using Management Layer functionality, each layer depends on the layers below it to work properly.

- The Configuration Layer processes and stores all the data required for running Genesys solutions in a particular environment; it notifies clients of any configuration changes. The Configuration Layer also controls user access to a solution's functions and data.
- The Management Layer controls the startup and status of solutions, logging of maintenance events, generation and processing of alarms, and management of application failures.
- The User Interaction Layer provides a comprehensive user interface to configure, monitor, and control the management environment.
- The Media Layer enables Genesys solutions to communicate across media, including traditional telephony systems, Voice over IP (VOIP), email, and the Web. This layer also provides the mechanism to distribute interaction-related business data within and across solutions.
- The Services Layer generates the statistical data used for interaction processing and contact center reporting.



Genesys Framework Architecture

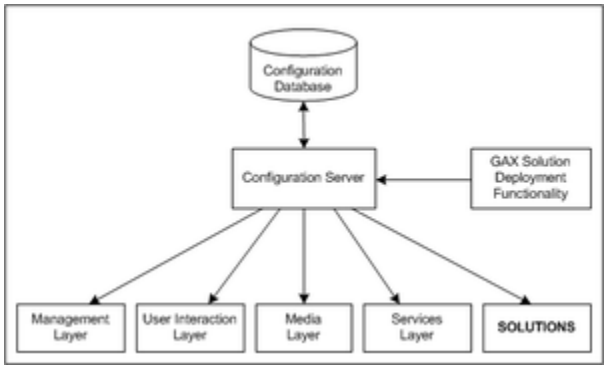
Configuration Layer

Functionality

The Configuration Layer provides:

- Centralized configuration data processing and storage for one-time entry of any information about contact center entities that any number of applications require to function in a particular business environment. Any number of applications can use this information.
- An advanced, configuration data-distribution mechanism, so applications can read their configuration upon startup and be notified of updates at runtime without service interruptions.
- Comprehensive data-integrity control functions that prevent entry of illogical configuration data that might cause solution malfunction.
- Advanced reconnection management which ensures that applications have up-to-date data after reestablishing connection to Configuration Server.
- Access control functions to regulate user access to solution functions and data, based on the access privileges set for each item.
- Wizards to help users through the automated process of solution deployment.
- Support for geographically distributed environments.
- Integration with external data sources, from which you can import configuration data to the Configuration Database.
- Import and export of configuration data to and from the Configuration Database.
- Secure data transfer between Genesys components using the Transport Layer Security (TLS) protocol.

Architecture



Configuration Layer Architecture

In the Configuration Layer:

- Configuration Server provides centralized access to the Configuration Database, based on permissions that super administrators can set for any user to any configuration object. Configuration Server also maintains the common logical integrity of configuration data and notifies applications of changes made to the data. Optionally, you can run Configuration Server in Proxy mode to support a geographically distributed environment. (The geographically distributed architecture is more complex than shown in the diagram.)
- Genesys Administrator, part of the User Interface Layer, provides a user-friendly interface for manipulating the contact center configuration data that solutions use and for setting user permissions for solution functions and data.
- The Configuration Database stores all configuration data.

> ### Warning
> Never add, delete, or modify any data in the Configuration Database, except through applications developed by Genesys, or through applications instrumented with the Genesys Configuration Server application programming interface (API). If you have compelling reasons for accessing the database directly, consult Genesys Customer Care before you do so.

- Genesys Administrator Extension solution deployment functionality automates deployment and upgrade. This functionality also handles solution-specific data integrity.
- Configuration Conversion Wizard (CCW) (not shown in the diagram) provides a user-friendly interface for migrating Genesys configuration data to the most recent data format. Database migration is optional, but required if you want to take advantage of the most recent features of Management Framework. Starting in release 8.1.3, CCW also enables you to migrate the Configuration Database to a multi-language format using UTF-8, and to migrate a pre-8.5.1 single-tenant Configuration Database to an 8.5.1 Configuration Database. Refer to the *Framework Migration Guide* for more information about CCW.

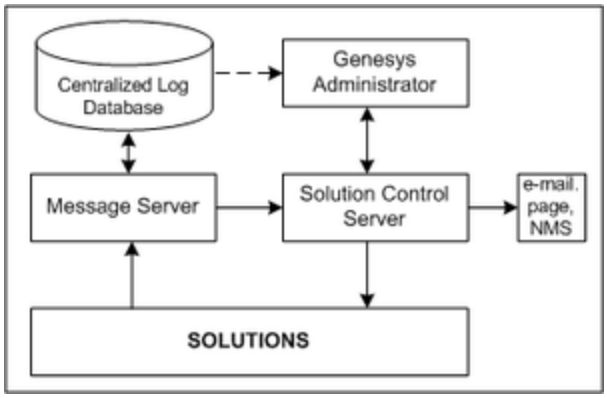
Management Layer

Functionality

The Management Layer provides:

- Centralized solution control and monitoring, displaying the real-time status of every configured Solution object, and activating and deactivating solutions and single applications, including user-defined solutions.
- Centralized logging that records applications maintenance events. The unified log format enables easy selection of required log records and centralized log storage for convenient access and solution-level troubleshooting. Centralized logging also allows you to track individual interactions, audit activities in your contact center, and store alarm history.
- Flexible alarm signaling that triggers alarms based on application maintenance events, system performance parameters, or Simple Network Management Protocol (SNMP) thresholds. Alarms are communicated to Genesys Administrator and can be written to system logs. You can configure the system to convert alarms into SNMP traps and send them as emails to a specified Internet address. (The latter automatically enables paging notifications.) The Management Layer automatically associates alarms with the solutions they affect and stores alarms as active conditions in the system until they are either removed by another maintenance event or cleared by the user.
- Fault-management functions, consisting of detection, isolation, and correction of application failures. For non-redundant configurations, the Management Layer automatically restarts applications that fail. For redundant configurations, this layer supports a switchover to the standby applications and also automatically restarts applications that fail.
- Individual host monitoring, including CPU and memory usage records and information about running processes and services.
- Support for geographically distributed environments.
- Support for the remote deployment of Genesys components, as performed in Genesys Administrator Extension.
- SNMP support for both alarm processing and SNMP data exchange with an SNMP-compliant network management system (NMS). As a result, you can integrate a third-party NMS with a Genesys system to serve as an end-user interface for control and monitoring functions and for alarm signaling functions.

Architecture



Management Layer Architecture

In the Management Layer:

- Local Control Agent (not shown in the diagram), located on every host that the Management Layer controls and/or monitors, is used to start and stop applications, detect application failures, and communicate application roles in redundancy context.
- Message Server provides centralized processing and storage of every application's maintenance events. Events are stored as log records in the Centralized Log Database where they are available for further centralized processing. Message Server also checks for log events configured to trigger alarms. If it detects a match, it sends the alarm to Solution Control Server for immediate processing.
- Solution Control Server is the processing center of the Management Layer. It uses Local Control Agents to start solution components in the proper order, monitor their status, and provide a restart or switchover in case of application failure. Solution Control Server also includes four utilities that provide the ability to gracefully stop T-Servers, handle T-Server stuck calls, send log messages on behalf of applications, and exchange information with Solution Control Server. These utilities can be installed with or without Solution Control Server.
- Genesys Administrator, a User Interaction Layer component, displays the status of all installed Genesys solutions and information about each active alarm, enables the user to start and stop solutions or single applications (including third-party applications), and enables advanced selection and viewing of maintenance logs.
- The Centralized Log Database (also called the Log Database) stores all application log records, including interaction-related records, alarm history records, and audit records.
- SNMP Master Agent (an optional component not shown in the diagram) provides an interface between the Management Layer and an SNMP-compliant NMS. It is required to support any SNMP-enabled Network Monitoring System, and to support Microsoft Operational Manager (MOM) technology.

  You can use Genesys SNMP Master Agent or any third-party SNMP Master Agent that is capable of using the AgentX protocol. You must configure Solution Control Server to connect to the SNMP Master Agent via the AgentX protocol. Genesys recommends that you use the open source Net-SNMP Master Agent. Starting with release 8.5.1, the Management Layer is capable of controlling and monitoring a Net-SNMP Master Agent in a similar manner as does Genesys SNMP Master Agent. Refer to SNMP Prerequisites and SNMP Master Agent for detailed information and limitations.

User Interaction Layer

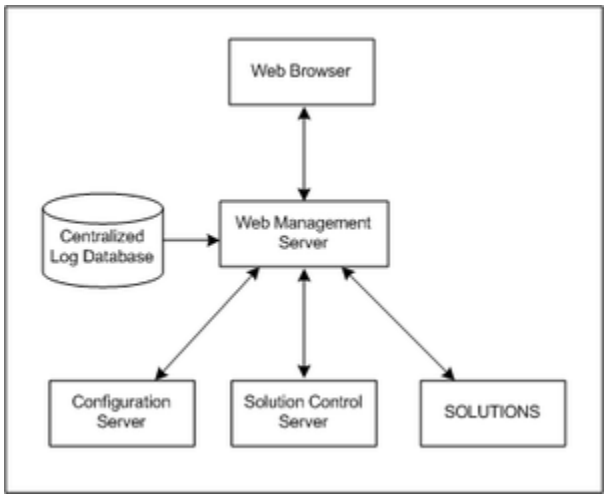
Functionality

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

- Deployment of Genesys components to any computer on the network using the Genesys Deployment Agent (a Management Layer component). Starting in release 8.5, this functionality is part of Genesys Administrator Extension.
- Configuration, monitoring, and control of applications and solutions.

Currently, Genesys Administrator and its extension is the only component in the User Interaction layer.

Architecture



User Interaction Layer Architecture

In the User Interaction Layer:

- The browser-based Genesys Administrator includes a comprehensive user interface to configure, monitor, and control the management environment.
- The Web Management Server:
  - Communicates with Configuration Server (a Configuration Layer component) to exchange configuration information.
  - Communicates with 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 web services for the browser-based Genesys Administrator.

- Depending on the solutions deployed in the system, the Web Management Server may also communicate with other back end servers to retrieve solution-specific information.

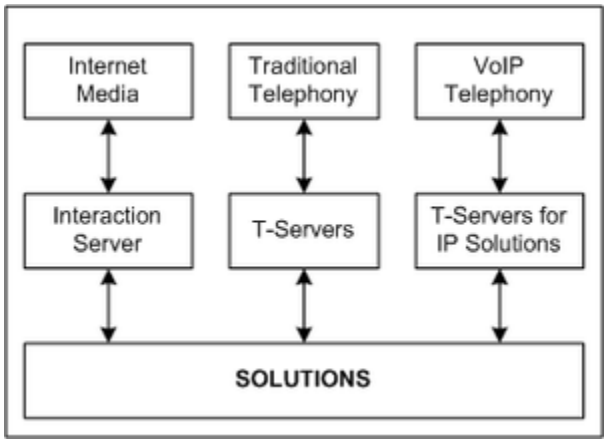
Media Layer

Functionality

The Media Layer provides:

- Interfaces to communication media.
- Distribution of interaction-related business data within and across solutions.

Architecture



Media Layer Architecture

In the Media Layer:

- Interaction Server provides an interface with Internet media like email and web communications. T-Server provides an interface with traditional telephony systems.
- T-Servers provide an interface with traditional telephony systems.
- T-Servers for IP Solutions provide an interface with VoIP telephony systems.

All of these servers communicate interaction-processing requests from the Genesys solutions to the media devices and distribute interaction-processing events in the opposite direction. They also maintain the current state of each interaction and all the business data collected about each interaction during processing stages. These servers distribute attached data to all the applications that participate in processing the interaction. They can also transfer that data across multiple interaction-processing sites.

Another Media Layer component, Load Distribution Server (LDS), not shown in the diagram, increases system scalability and availability.

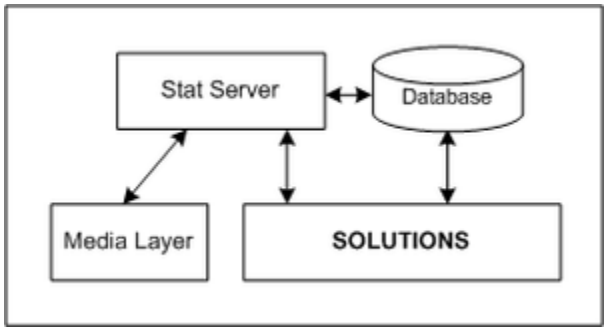
Services Layer

Functionality

The Services Layer provides:

- Conversion of events related to management of single interactions into statistical data, which is then used for interaction processing and contact center reporting.

Architecture



Services Layer Architecture

Stat Server tracks real-time states of interaction management resources and collects statistics about contact center performance. Genesys solutions use the statistical data to more *intelligently* manage real-time interactions. Through Genesys Reporting, you can use the data to generate real-time and historical contact center reports.

Framework Connections

The following diagram shows connections that Framework components establish to each other and to solutions.

Framework Connections

IPv6 Support

All Framework server components support IPv6, except for the following:

- When obtaining technical licenses connecting to the FlexNet license server, all Framework servers support IPv6 on only the RHEL 5 64-bit and Windows 2008 64-bit operating systems.

For more information about IPv6, see Internet Protocol version 6 (IPv6) and IPv6 vs. IPv4 Overview.

Deployment Planning

Achieving optimal performance with your Genesys installation requires comprehensive planning. How well Genesys Framework components function in a particular environment depends on a number of variables, including amount of computer memory, network location of the applications, and the specific tasks the applications perform. The information in this section describes various characteristics of Framework components and looks at how they interact with each other and the applications they serve. It provides basic data and makes recommendations that will help you select the optimal components for your specific needs, choose a computer for each component, and define the optimal location for each component on the network.

Start your deployment planning by identifying the existing telephony resources in your contact center environment. Then follow the deployment recommendations for each architecture layer given in Network Locations for Framework Components.

Consider whether you can benefit from:

- Using the Management Layer.
- Having redundant components.
- Installing additional Configuration Servers in Proxy mode.
- Installing a number of Solution Control Servers in Distributed mode.
- Using Load Distribution Server.

In addition, review Solution Availability and Security Considerations, which are common aspects of any Genesys installation.

Finally, prepare an installation worksheet summarizing your configuration requirements, and fill it in and refer to it as you deploy Framework.

Initial Considerations

How well Genesys Framework components function in a particular environment depends on a number of variables, including amount of computer memory, network location of the applications, and the specific tasks the applications perform. This section provides basic data and makes recommendations that will help you select the optimal components for your specific needs, choose a computer for each component, and define the optimal location for each component on the network.

Telephony Network Description

Certain information is required to deploy Framework, so prepare a description of your telephony and media network as discussed in this section. You will use data from this description when supplying configuration parameters to Deployment Wizards or when configuring objects for your contact center using Genesys Administrator.

You must have the following information available for every switch that you plan to use in your interaction management solution:

- Switch type, which usually corresponds to the switch vendor, brand name, and model number.
- Version of the switch software.
- Type of CTI Link (TCP/IP, X.25, or ISDN).
- Version of the CTI Link software.
- Information required to connect to the CTI Link (for example, for TCP/IP connection, host name and port number), including password, service id, and other parameters required for switch security.
- Types and numbers of telephony devices, also called Directory Numbers or DNs. You may have to configure specific types of DNs (for example, Routing Points) on the switches to support functions of the interaction management solutions.
- Login codes to be assigned to agents for runtime associations between agents and their working places.
- Information about how the switch DNs are arranged into working places.
- Information about how DNs that belong to a particular switch can be reached from other switches in a multi-site installation.

In addition, describe your contact center resources:

- For every user who must access any interaction management application, define the following parameters: a unique employee ID, unique user name, and password. The role of a user in the contact center defines the set of access privileges for this user in the system. For more information, see Security Considerations.
- For agents, define Login codes in every switch at which they might be working.
- For agents, define skills that might be considered as criteria for effective interaction processing.
- Note how agents are arranged into groups.
- Decide how to arrange the working places into groups.

Guidelines for Naming Hosts

To ensure that the operating systems properly interpret host names, follow these guidelines when naming the host computers in your system:

1. If possible, use the host's DNS name.
2. If it is not possible to use the DNS name, use the host's IP address, in the format x.x.x.x. However, verify the availability of that IP address by using the command `ping <IP address>` on the command-line

before starting the installation process.

Configuration Environment

Genesys provides its software to two types of companies:

- Companies that own their telephony equipment and use it for their own needs.
- Companies (such as service providers) that make their telephony equipment available to other companies.

A single Genesys configuration environment can be used to address the needs of both of these types of companies. You establish that configuration environment when you create the Configuration Database structure during the Configuration Layer installation.

Object Hierarchies

In complex configurations, there might be a service provider making its telephony equipment available to other companies. So, this configuration environment also serves the needs of every company using the service. In this environment, configuration information about the resources that are managed exclusively by the service provider is visible on the service provider side only. Only personnel from the service provider company can register the entities that provide the technical foundation for setting up the CTI services, such as switching offices, data network hosts, and CTI applications. These resources may be shared by some or all of the companies using the service ("Tenants"). The resources of the individual companies, such as user accounts, agent groups, outbound campaigns, and so forth, are configured separately by the personnel of these companies. This configuration is visible only to that company's users.

This general structure can extend to an unlimited number of layers. There might be a service provider that provides its services not only to companies that use its services directly but to other companies, such as resellers, who in turn sell those services to other companies. The customers of these resellers can, in turn, be direct users and perhaps other resellers. Tenants that provide services to other tenants are called parent tenants; those that use these services are called child tenants. Therefore, a single Tenant object can be a parent, a child, or both.

This structure can also support the simplest of configurations, where there is only one tenant that uses it own resources.

> **Important**
>
> In release 8.5.1, the *single-tenant* (or *Enterprise*) and *multi-tenant* Configuration Server were consolidated into one Configuration Server that can support one, many, or a hierarchy of tenants. Existing single-tenant configurations are still supported, and you can upgrade existing databases to the latest available format, but single-tenant configurations can no longer be deployed for new installations. For new installations, Genesys recommends that you install Configuration Server and create new databases using the provided multi-tenant database initialization scripts, and then configure an environment with one Tenant (Environment).

Large Configuration Environments

Genesys defines a large configuration environment as one in which the Configuration Database stores 50,000 or more configuration objects. Refer to the "Management Framework" section of the *Genesys Hardware Sizing Guide* for guidelines when operating within a large configuration environment.

Using National Languages

Single-language Environments

The default (legacy) deployment of Genesys software can support only one language in addition to English with which to process data and display messages. Genesys recommends that you select one language for your installation, and use that language across all components and databases.

Multi-language Environments

UTF-8 data encoding enables a system to work with multi-language data that is encoded with UTF-8. This support applies to all string fields of all configuration objects, with some exceptions noted later in this section.

This functionality is optional, and must be enabled to take effect.

> **Warning**
> You must use a separate set of initialization scripts to enable multi-language mode when creating the database, following the setup of the configuration file option for master Configuration Servers.

You can perform an object search of data encoded in UTF-8, using the standard wildcard symbols, with search data supplied by UTF-8. The search parameters, in UTF-8, are compared with the data. All fields that are searchable with non-UTF-8 data are searchable with the UTF-8 data.

To configure your system to support multiple languages, satisfy the database prerequisites, then follow the steps in Deploying the Configuration Layer, taking note of the special requirements to support UTF-8.

Framework Support for UTF-8

Framework supports UTF-8 encoding of the following:

- Most configuration fields (exceptions below)
- Solution Control Server alarm names, messages, and display thereof
- Content of log messages in Message Server and in the centralized Log Database
- UTF-8 initialization of the Configuration Database and the Log Database

Framework does not support UTF-8 encoding of the following items; they must be in ASCII.

- Names of Application objects
- Command-line arguments specified during configuration in the **Start Info** section
- Command-line arguments used by **mlcmd** and **logutility**
- Local configuration file used by Configuration Server
- SNMP traps and scalar data
- Database parameters in Database Access Points
- Host names
- Database table name in Table Access objects
- Log names and log file names specified in the **[log]** configuration option section

DBMS Support for UTF-8 Encoding

The following DBMS can be used with UTF-8 encoding:

- DB2
- MS SQL (uses UCS-2 encoding)
- Oracle
- PostgreSQL

Converting from a non-UTF-8 Database to a UTF-8 Database

Starting in release 8.1.3, you can use the Configuration Conversion Wizard (CCW) to convert a non-UTF-8 compatible Configuration Database into a database that can store and work with encoded data. Refer to the *Framework Migration Guide* for information about using CCW to convert your configuration database.

Except for the conversion of the Configuration Database, Genesys does not otherwise provide any tools to support the migration of an existing database, that cannot work with UTF-8 data, into a database than can store and work with encoded data. Genesys recommends that you use tools and utilities provided by the DBMS you are using to do any such conversion.

> ### Warning
> After you have upgraded to UTF-8, legacy applications will be unable to connect to Configuration Server, unless you use the **allow-mixed-encoding** configuration option. Refer to the *Framework Configuration Options Reference Manual* for more information about this option.

Management Framework and Solution Availability

Think of the *availability* of an interaction management solution as the amount of time that the solution is available to process enterprise interactions. Two major categories of events affect availability: changes in the operating conditions and failures. The first category combines the various operational and maintenance activities that require temporary shutdown and restart of the entire system or of one of its components. The second category deals with the temporary inability of the solution to perform its required functions because of operator errors or software faults.

Given the complexity of the solution architecture, remember that:

- Any interaction management solution relies on functionality provided by a number of components, each performing a specific task. The overall availability of a solution depends on the availability of each of the components involved.
- Interaction management solutions do not operate in isolation. On the contrary, they essentially bring together various business resources, such as telephony switches, call-processing telephony terminations, database management systems, and Internet communication servers. As such, the inability of an interaction management solution to perform its required function may be the result of the unavailability of an external component or system.
- Genesys solutions, which consist of software components only, operate on hardware platforms that require maintenance and that are subject to failures. For example, running redundant processes on the same host may work in the presence of a software failure; however, it offers no protection if the computer itself or a communication link to it fails. The availability of a solution can never be greater than the availability of the underlying hardware platform.

The Genesys Framework is designed to minimize the impact on solution availability associated with operational and maintenance activities. Because the Configuration Layer updates solutions about any configuration changes at runtime, uninterrupted solution operations are guaranteed regardless of the number or frequency of changes made to the contact center environment. Dynamic reconfiguration is a standard feature of every Genesys 7.x and 8.x component and does not require you to make any special adjustments to enable configuration settings.

Solution availability can also be affected by accidental operator errors, unauthorized actions, or actions that are carried out in a less than skillful manner. The data integrity rules implemented in the Configuration Layer greatly reduce errors of the first type. The basic integrity rules common across all solutions are supported by Configuration Server, and therefore enforced regardless of the type of client application through which the data is managed. More advanced integrity rules specific to a particular solution are implemented in the solution wizards. Genesys recommends that you use wizards for the initial deployment of solutions and major configuration updates in the course of solution operation.

Solution availability can also be impacted by the occurrence of a disaster, natural or man-made, that causes an entire site to go down. See Disaster Recovery/Business Continuity for a new Disaster Recovery architecture to prevent permanent failure of Management Framework itself because of the loss of the entire site.

Genesys Framework also provides a comprehensive set of access control functions that help minimize the risk of failures associated with unskilled or unauthorized operator actions. For more information

about these functions, see Security Considerations.

Finally, to reduce the impact on solution operations, schedule all operational and maintenance activities that directly affect system behavior for off-peak hours, when solutions operate at minimum loads.

Faults-accidental and unplanned events causing a system to fail-present the biggest challenge to solution availability. The functions that detect, isolate, and correct various types of faults are partly incorporated into every Genesys component and partly implemented in the Management Layer of the Genesys Framework. Refer to the *Framework Management Layer User's Guide* for more information about the various fault-detection mechanisms implemented in Genesys software.

Communication Session Failures

In a distributed interaction management solution, components must communicate continuously with each other and with some external resources. A communication session with a required resource can fail for any of these reasons:

- Failure of the resource itself
- Problem with the hardware where the resource is located
- Network connectivity problem between the two points
- Forced termination of the connection that has not shown any activity for a specified amount of time

Any time a solution component cannot communicate with a required resource, the solution may not be able to perform its required function.

After a failure is detected, the fault correction procedure normally consists of repeated attempts to regain access to either the resource in question or to a redundant resource, if one is available.

Each underlying communication protocol is typically equipped with functions that monitor open communication sessions. When a failure is detected, the communication software signals an abnormal condition to the interacting processes. This detection mechanism is fully supported in the Genesys solution, whose connection layer translates system messages into appropriate events on the application level.

However, communication protocols do not always provide adequate detection times. The TCP/IP stack, for example, may take several minutes to report a failure associated with a hardware problem (such as when a computer goes down or a cable is disconnected). This delay presents a serious challenge to the availability of any interaction management solution.

Software Exceptions

A *software exception* is an interruption in the normal flow of a program caused by an internal defect. An operating system generates exceptions in response to illegal operations that a software program attempts to perform. After generating an exception, the operating system terminates the process, which may make unavailable all solutions that use the functionality of this component.

Genesys provides an exception-handling function that monitors the exceptions that the operating system generates. The function attempts to prevent application termination by skipping the program block from which the exception originated. In most cases, this action amounts to losing one processing step with respect to a single interaction in favor of preventing an application failure.

Although the function attempts to prevent application termination, it still reports the exception with the highest priority marking. This ensures that operators know about the exception and can take appropriate measures.

You can configure the number of times during which the function tries to prevent an application from failing if it continues to generate the same exception. If this threshold is exceeded, the exception-

handling function abandons the recovery procedure, allowing the operating system to terminate the application. This termination can then be detected and corrected by external fault-management functions.

By default, the exception-handling function is enabled in any daemon application; six exceptions occurring in 10 seconds will not cause an application to terminate. To change these parameters or disable the exception handling, use a corresponding command-line parameter when starting an application.

Application Failures

A complete application failure may be a result of either an internal defect (for example, an infinite loop) or an external event (for example, a power failure). It may manifest itself as either a process nonresponse or termination. Typically, if a solution component stops working, the solution is no longer available to process customer interactions.

Because the application that fails cannot perform any functions, you must use an external mechanism for both detection and correction of faults of this type. In Framework, the Management Layer is this mechanism. For information about the architecture and components in the Management Layer, see the *Framework Management Layer User's Guide*.

Configuration Server Failure Because of Memory Starvation

When Configuration Server responds to client requests with data, the responses are stored in Configuration Server memory until they are sent. The rate at which they are send depends on several factors, such as:

- load on Configuration Server
- network throughput
- ability of the client to receive and process the data

In some cases, the unsent messages might accumulate in memory. In severe cases, they could accumulate to the point where Configuration has to terminate unexpectedly because it has used 100% of memory.

To resolve this, you can impose flow control by limiting how much memory is used by unsent mail. When this limit is reached, Configuration Server stops processing client requests. When the backlog of unsent requests starts to clear and its memory usage drops below the imposed limit, Configuration Server starts process client requests again, in the order in which they were received.

Flow control is activated by two configuration options. **max-client-output-queue-size** provides flow control for communications for a single client. **max-output-queue-size** defines flow control for all clients.

> **Warning**
> Be very careful when using this option, as it effectively stops Configuration Server

> until all of its output buffers drop below the specified limit. Use this option only as a last resort.

Refer to the *Framework Configuration Options Reference Manual* for detailed descriptions about these options.

Remote Site Failures

Starting in release 8.0, each Solution Control Server in a Distributed Solution Control Server environment can detect the failure of a remote site controlled by another Solution Control Server. Refer to the *Framework Management Layer User's Guide* for more information.

Network Connectivity

Network connectivity handles network communication in two modes:

- *single-threaded mode*—on the same thread as other processes
- *multi-threaded mode*—on a separate thread

The purpose of using a separate thread is to increase the robustness of detecting and handling network disconnections without impacting other processes.

By default, Configuration Server, Solution Control Server, and Message Server handle network communications in single-threaded mode. Starting in 8.5.1, you can use an environment variable with Configuration Server, Solution Control Server, and Message Server to indicate how each component is to handle network communications, using either single-threading or multi-threading. The three system environment variables are:

- Configuration Server—**GCTI_CONFSERV_CONN_STARTUP_DEFAULT**
- Solution Control Server—**GCTI_SCS_CONN_STARTUP_DEFAULT**
- Message Server Server—**GCTI_MS_CONN_STARTUP_DEFAULT**

Each of these variables are set on the host on which the corresponding component is running. If not set or set to one (1), the component runs in single-threaded mode. If set to zero (0), the component runs in multi-threaded mode.

> **Important**
>
> - The default threading mode is single-threaded.
> - Genesys strongly recommends that you do not change the threading mode for the current version of the component.

Security Considerations

This section outlines some of the security capabilities provided in Configuration Layer for your data, both from access by unauthorized users and during its transfer between components. For more information about these and other security features, and for full implementation instructions, refer to the *Genesys Security Deployment Guide*.

Access to Hosts File at Start-up

By default, Genesys components try to read from the hosts file at startup to enable them to resolve host names. If an organization has a security policy against this, they can configure the environment variable **GCTI_DNS_USE_HOSTSFILE=0** to disable this access.

User Authentication

User authentication refers to ensuring that the user is actually who he or she claims to be. In Genesys software, this is implemented by the Configuration Server. The data that a Genesys solution requires for operating in a particular environment, as well as the applications and the solutions, is represented as Configuration Database objects. Any person who needs access to this data or these applications must have an account in this database.

Logging In

At startup, every Genesys GUI application opens a Login dialog box for users to supply a User Name and Password, which are used for authentication. The authentication procedure succeeds only if a User with the specified User Name and Password is registered in the Configuration Database. Otherwise, the working session is stopped.

Last Logged In

Starting in release 8.0, you can configure Configuration Server so that some Genesys GUI applications display the date and time of the previous login for the currently logged-in user. Each user can then detect if someone else had accessed the system using their credentials.

Forced Re-Login for Inactivity

You can configure some Genesys GUIs to automatically force a logged-in user to log in again if he or she has not interacted with any element of the interface for a set period of time. In some interfaces, open windows are also minimized, and are restored only when the user logs back in. This functionality is configured in each interface, and is therefore specific to that interface. By default, this functionality is not active, and must be activated on an instance-by-instance basis for those GUI applications that are to use the feature.

> **Important**
>
> This inactivity feature survives reconnection timeouts. In other words, if the interface application becomes disconnected from Configuration Server after the forced re-login timeout has expired but before the user has logged in again, the user must still log in before he or she can access the system.

User Authorization

User authorization refers to ensuring that an authenticated user is entitled to access the system, either all or parts thereof, and defines what the user can do to or with the data that they can access.

The security mechanism implemented in Configuration Server allows the system administrator to define, for each valid user account, a level of access to sets of objects. The access privileges of valid user accounts define what the user can and cannot do within the corresponding set of objects.

Starting in release 8.0, an additional layer of security is available through Genesys Administrator, called Role-Based Access Control. This enables the system administrator (or a designated individual) to define access to objects based on what is to be done (viewed, modified, deleted) to the objects.

This section provides an overview of the various mechanisms in place to ensure data is accessed by only authorized users. For detailed information about how Genesys software implements user authorization, refer to the *Genesys Security Deployment Guide*.

Access Permissions

The level of access to sets of objects granted by the system administrator is defined by a combination of elementary permissions. Each user must be assigned at least one permission; without it, the user has no access to any data.

Access control for daemon applications is different from that for GUI applications. Access permissions for GUI applications are determined by the profile of the person who is currently logged in.

Access Groups

Access Groups are groups of Users who need to have the same set of permissions for Configuration Database objects. By adding individuals to Access Groups-and then setting permissions for those groups-access control is greatly simplified.

Genesys provides preconfigured default Access Groups. You can also create your own Access Groups to customize your own security environment.

Master Account and Super Administrators

The Configuration Database contains a predefined User object, otherwise known as the *Master Account* or *Default User*. The Default User, named `default` and with a password of `password`, is not associated with any Access Group. The Master Account always exists in the system and has a full set

of permissions with respect to all objects in the Configuration Database. You must use this account when you log in to the Configuration Layer for the first time since the Configuration Database initialization. Genesys recommends changing the default name and password of the Master Account, storing them securely, and using this account only for emergency purposes or whenever it is specifically required.

Changing Default Permissions

The default permissions that the Configuration Layer sets provide users with a broad range of access privileges. You can always change those default settings to match the access needs of a particular contact center environment.

> **Important**
>
> Genesys does not recommend changing the default access control setting unless absolutely necessary. Remember, the more complex the security system is, the more difficult it becomes to manage the data and the more it affects the performance of the Configuration Layer software.

Genesys provides two mechanisms to help you manage changes to your permissions-propagation and recursion. Refer to the *Genesys Security Deployment Guide* for details about these mechanisms and how to use them.

New Users

Configuration Server does not assign a new user to an Access Group when the user is created. In effect, the new user has no privileges, and cannot log in to any interface or use a daemon application. The new user must be explicitly added to appropriate Access Groups by an Administrator or by existing users with access rights to modify the user's account. Refer to *Genesys Administrator 8.1 Help* for more information about adding a user to an Access Group.

By default, this behavior applies to all new users added by Configuration Server release 7.6 or later. Users created before release 7.6 keep their existing set of permissions and Access Group assignments. If you want new users to be added automatically to pre-defined Access Groups, as was the behavior prior to release 7.6, you must manually disable this feature by using the Configuration Server configuration option **no-default-access**.

For more information about this feature, including how it works and how to modify it, refer to the *Genesys Security Deployment Guide*.

Login Security Banner

You can create your own security banner to be displayed to a user logging in to Genesys Administrator. You define the content of the banner, typically the terms of use of the application. Users must accept the terms to proceed, or they can reject the terms to close the application without access.

The user-defined security banner is specified during the installation of each instance of a GUI application, such as Genesys Administrator.

Refer to the *Genesys Security Deployment Guide* for more details about the security banner.

Genesys Security Using the TLS Protocol

Genesys supports the optional use of the Transport Layer Security (TLS) protocol to secure data transfer between its components. TLS is supported on Windows and UNIX platforms.

To enable secure data transfer between Genesys components that support this functionality, you must configure additional parameters in the Host objects and Application objects that represent these components. Certificates and corresponding private keys are generated using standard Public Key Infrastructure (PKI) tools, such as OpenSSL and Windows Certification services.

For detailed information about Genesys Security Using the TLS Protocol, refer to the *Genesys Security Deployment Guide*.

Multiple Ports

To provide flexibility in configuring a system with the Genesys Security using the TLS Protocol feature, you can configure multiple ports on a given server with either secure or unsecured connections. You specify the additional ports in the **Server Info** section on the **Configuration** tab of the server's Application object.

Each port can have one of the following listening modes:

- unsecured—The port is not secured by TLS. This is the default status of a port.
- secured—The port is secured by TLS.
- auto-detect—This status applies only to ports on the Configuration Server, and is used only when configuring secure connections to the Configuration Server. If an application that is trying to connect to an auto-detect port has security settings specified in its configuration, Configuration Server checks the validity of those settings. Depending on the results, the client will be connected in secure or unsecured mode.

Refer to the *Genesys Security Deployment Guide* for more information about multiple ports.

Multiple Ports on Configuration Server

When you install Configuration Server, the listening port that you specify during installation is stored in the configuration file as the **port** option. When Configuration Server first starts with an initialized database, it reads the **port** option in the configuration file. The value of the **port** option is also propagated to the Configuration Database, where it is stored as part of the Configuration Server Application object. As additional ports are configured, they are also stored in the Configuration Database as part of the Configuration Server Application object. On subsequent startups of Configuration Server—that is, on all startups after the first—Configuration Server reads the port information from the Configuration Server Application object, ignoring the **port** option in the configuration file.

If necessary, you can specify an additional unsecured listening port in the Configuration Server command line during subsequent startups. This additional port is not written to the Configuration Server Application object, and does not survive a restart of Configuration Server. Use this option only

when regular ports cannot be opened. See **-cfglib_port** for more information about this option.

Dedicated Ports for Client Connections

Starting in release 8.5.1, you can configure separate ports on Configuration Server or Configuration Server Proxy that are restricted for use only by client User Interface type (UI) applications. All other applications would continue to use the assigned listening ports as usual. A firewall is used to direct these applications to the dedicated port, where they are authorized before being allowed access. For more information and instructions, see Configuring a Dedicated Port for Client UI Applications later in this Guide.

The same principle applies in an HA Configuration Server configuration. Configuration Server Proxy has a dedicated port to which the firewall directs the UI applications, while other applications connect through other ports on the proxy server. For instructions on configuring this dedicated port, see Configuring a Dedicated Port for Client UI Applications.

Secure Connections

In addition to configuring secure ports on your server applications, you must configure your client applications, both server and user interface types, to connect to these ports. Use Genesys Administrator to configure these connections.

There are only two exceptions to this standard procedure, as follows:

- Configuring secure connections to the Configuration Server—You must configure a Configuration Server port as an auto-detect port.
- Configuring a secure connection between DB Server and Configuration Server—You must configure the secure connection in the configuration files of the two components.

Refer to the *Genesys Security Deployment Guide* for detailed instructions for configuring secure connections.

European Data Protection Directive Disclaimer

The Genesys suite of products is designed to make up part of a fully functioning contact center solution, which may include certain non-Genesys components and customer systems. Genesys products are intended to provide customers with reasonable flexibility in designing their own contact center solutions. As such, it is possible for a customer to use the Genesys suite of products in a manner that complies with the European Data Protection Directive (EDPD). However, the Genesys products are merely tools to be used by the customer and cannot ensure or enforce compliance with the EDPD. It is solely the customer's responsibility to ensure that any use of the Genesys suite of products complies with the EDPD. Genesys recommends that the customer take steps to ensure compliance with the EDPD as well as any other applicable local security requirements.

Network Locations for Framework Components

This section provides basic data and makes recommendations that will help you select the optimal components for your specific needs, choose a computer for each component, and define the optimal location for each component on the network.

A separate section presents the information for each layer of Framework.

> **Important**
> In release 8.x, Genesys Administrator is the recommended interface for Management Framework, in place of Configuration Manager and Solution Control Interface, both of which are still available for download and use with this release of Management Framework. For this reason, Configuration Manager and Solution Control Interface are not mentioned in this section. For more information, refer to User Interaction Layer (Genesys Administrator), and the *Framework Genesys Administrator Deployment Guide*.

Configuration Layer

The Configuration Layer is a mandatory part of any Genesys CTI installation. You cannot configure and run any other layers of Framework-or any solutions-unless Configuration Layer components are running.

This section provides recommendations for planning and installing the Configuration Layer components.

Configuration Database

The Configuration Database stores all configuration data.

When planning your installation, follow these recommendations for the Configuration Database:

- The size of the Configuration Database depends on the size of the contact center, or—more precisely—on the number of entities in the contact center that you specify as configuration data objects. If data storage capacity is limited, consider allocating 10 KB of space for every object in the contact center as a general guideline. Otherwise, allocating 300 MB accommodates a Configuration Database for a typical installation with one tenant.
- If you want to deploy a Disaster Recovery/Business Continuity architecture, you must set up Configuration Databases across sites. Refer to Disaster Recovery/Business Continuity for more information.

- Treat the Configuration Database as a mission-critical data storage. Ensure that only the properly qualified personnel gain access to the DBMS that contains the Configuration Database itself. Information about access to the database is stored in the configuration file of Configuration Server. To protect this file, place it in a directory that is accessible only to the people directly involved with Configuration Layer maintenance.
- Consider encrypting the database access password via Configuration Server.
- As with any mission-critical data, regularly back up the Configuration Database. Base the frequency of scheduled backups on the rate of modifications in a particular configuration environment. Always back up the database before making any essential modifications, such as the addition of a new site or solution.
- Switch Configuration Server to Read-Only mode before performing any maintenance activities related to the Configuration Database.
- Save the records of all maintenance activities related to the Configuration Database.
- Users of the Configuration Database should have at least the following privileges for all tables in the database:
  - SELECT
  - INSERT
  - UPDATE
  - DELETE

Warning

- Never add, delete, or modify any data in the Configuration Database, except through applications developed by Genesys, or through applications instrumented with the Genesys Configuration Server application programming interface (API). If you have compelling reasons for accessing the database directly, consult Genesys Customer Care before you do so.
- Configuration Server treats its information and checks integrity constraints in a case-sensitive manner. Therefore, your SQL database must be installed and configured in case-sensitive mode. Refer to your SQL Server Administrator documentation for additional information.

Configuration Server

Configuration Server provides centralized access to the Configuration Database, based on permissions that you can set for any user to any configuration object. Configuration Server also maintains the common logical integrity of configuration data and notifies applications of changes made to the data.

When planning your installation, follow these recommendations for Configuration Server:

- Genesys solutions installed in a particular environment can have only one Configuration Database managed though one Configuration Server at a time.
- Because Configuration Server keeps all configuration data in its memory, allocate memory for this

server based on the expected size of the Configuration Database. Refer to the *Management Framework* section of the *Genesys Hardware Sizing Guide* for assistance in determining the amount of memory to allocate for Configuration Server.

- If you want to deploy a Disaster Recovery/Business Continuity architecture, you must set up Configuration Servers across sites. Refer to Disaster Recovery/Business Continuity for more information.
- For client connections:
  - Connect all administrative applications that do WRITE operations to Configuration Server directly.
  - Any other Genesys server applications should be connected to either Configuration Server (if server capacity permits) or Configuration Server Proxy. Server applications that communicate directly with each other, such as URS and T-Server, must be connected to the same Configuration Server or Configuration Server Proxy.
- You can deploy redundant (HA) Configuration Servers.
- Always use SCS to control Configuration Server HA pairs. This SCS must be directly connected to the master Configuration Server.

> **Important**
>
> Configuration Servers in HA Pairs cannot be switched over manually.

Configuration Server Proxy

To support a large number of clients and/or distributed installations, Configuration Server can operate in Proxy mode. In this document, a Configuration Server that operates in Proxy mode is called *Configuration Server Proxy*. For more information about Configuration Server Proxy, see Solution Availability.

When planning your installation, follow these recommendations for Configuration Server Proxy:

- Refer to the *Management Framework* section of the *Genesys Hardware Sizing Guide* for assistance in determining the amount of memory to allocate for Configuration Server Proxy.
- You can install Configuration Server Proxy anywhere on the network because it does not generate heavy traffic.
- If you want to deploy a Disaster Recovery/Business Continuity architecture, you might consider setting up Configuration Server Proxies across sites. Refer to Disaster Recovery/Business Continuity for more information.
- If you are using any agent-facing interfaces, such as Workspace Desktop Edition, or interfaces that will be accessing the Configuration Database on a read-only basis, connect those interfaces to Configuration Server Proxy.
- You can deploy redundant (HA) Configuration Server Proxies.
- Always use SCS to control Configuration Server Proxy.

Genesys Security Pack on UNIX

Genesys Security Pack on UNIX, an optional component of the Configuration Layer, provides the components, such as shared libraries, which are used for generation of certificates and their deployment on UNIX computers on which Genesys components are installed. For more information, refer to the *Genesys Security Deployment Guide*.

Management Layer

The exact configuration of the Management Layer depends on which of the following management functions you would like to use. Genesys recommends that you use all of these capabilities to optimize solution management.

Required Components

If you intend to use one or more of the Management Layer capabilities, plan to install the components required for each capability, as outlined below. Refer to the *Framework Management Layer User's Guide* for descriptions of, and recommendations for, these components.

Solution and application control and monitoring

Install these components to control and monitor solutions and applications:

- Local Control Agent
- Solution Control Server

Centralized Logging

Install these components to use centralized logging:

- Centralized Log Database
- Message Server

> **Important**
> Although Solution Control Server is not required, it is a source of log events vital for solution maintenance. For example, Solution Control Server generates log events related to detection and correction of application failures. As such, it is useful for centralized logging.

Alarm Signaling

Install these components to provide alarm signaling:

- Message Server
- Solution Control Server
- Genesys SNMP Master Agent, if SNMP alarm signaling is required. See also SNMP Support.

Application Failure Management

Install these components to detect and correct application failures:

- Local Control Agent
- Solution Control Server

See Application Failures for information about the application-failure management mechanism.

SNMP Support

Install the following components to integrate Genesys Framework with an SNMP-compliant third-party network management system (NMS):

- Local Control Agent
- Solution Control Server
- Genesys SNMP Master Agent, Net-SNMP, or another third-party SNMP Master Agent compliant with the AgentX protocol
- Message Server if SNMP alarm signaling is required

Note: Starting in release 8.5.1, Net-SNMP can be used to provide the same functionality as the built-in SNMP support, and the two can run in parallel. In this case, Net-SNMP must also be installed.

Management Layer Components

This section provides recommendations for planning and installing the Management Layer components.

Local Control Agent

When planning your installation, follow these recommendations for Local Control Agent:

- Install an instance of LCA on each computer running a monitored application, whether a Genesys daemon or a third-party application. LCA is installed at the port number you specify in the LCA Port property of the corresponding Host object in the Configuration Database. If you do not specify a value for LCA Port, the LCA default port number is 4999. By default, LCA runs automatically on computer startup.

> **Important**
> On Windows operating systems, the installation script always installs LCA as a

Windows Service. If you are changing the LCA port number in the host configuration after the installation, you must also change the port number in the ImagePath in the application folder, which you can find in the Registry Editor. Refer to Notes on Configuring the LCA Port for instructions.

- If you want to deploy a Disaster Recovery/Business Continuity architecture, you must set up an LCA across all sites. Refer to Disaster Recovery/Business Continuity for more information.
- On UNIX platforms, LCA must be added to the **r/c** files during the installation, so that LCA can start automatically on computer startup. In practice, this means that the person installing LCA must have sufficient permissions.

Message Server

When planning your installation, follow these recommendations for Message Server:

- Genesys recommends the use of one Message Server and of one Log Database for all but large installations. If you are working within a large installation and are considering evenly dividing the total log-event traffic among number of Message Servers, each serving any number of clients, keep the following facts in mind:
  - Although any number of Message Servers can store log records in the same Log Database, one Message Server cannot store log records in more than one Log Database.
  - Because any number of Message Servers can send log records to Solution Control Server, Genesys Administrator can display alarms based on log records from a few Message Servers.
- If you want to deploy a Disaster Recovery/Business Continuity architecture, you must set up Message Servers across sites, with one dedicated for communication between all Solution Control Servers at all sites. Refer to Disaster Recovery/Business Continuity for more information.
- If you want an application to generate alarms, you must configure it to send log events to Message Server. Use the same Message Server for both the centralized logging and alarm signaling.
- If you want Message Server to provide alarms, you must connect it to Solution Control Server. This means that you must configure a connection to every Message Server in the SCS Application object.
- As with any other daemon application, you can deploy redundant Message Servers.
- To optimize the performance of the connection to the Log Database, configure the number of messages that the Message Server sends to the database before receiving a response. The smaller the number of messages, the greater the decrease in performance. See the "Message Server" section of the *Framework Configuration Options Reference Manual*, for more information.

Solution Control Server

When planning your installation, follow these recommendations for Solution Control Server:

- Given that you can install and use more than one SCS that is operating in Distributed mode within a given configuration environment, consider deploying a few Solution Control Servers in this mode for large or geographically distributed installations. In these installations, each server controls its own subset of Host, Application, and Solution objects. Distributed Solution Control Servers communicate with each other through a dedicated Message Server.
- If you want to deploy a Disaster Recovery/Business Continuity architecture, you must set up Distributed

Solution Control Servers across sites. Refer to Disaster Recovery/Business Continuity for more information.

- As with any other daemon application, you can deploy redundant Solution Control Servers. Redundancy support for SCS is implemented through direct communication between the backup SCS and the LCA of the host on which the primary SCS runs. Be sure to synchronize the ports between primary and backup Solution Control Servers.

> **Important**
> You cannot perform a manual switchover for Solution Control Server.

Centralized Log Database

As with any historical database, the size of the Centralized Log Database grows with time. When you are planning your installation, keep in mind that:

- The maximum allowable record size is 1 KB.
- The size of the Centralized Log Database depends on:
  - The number of applications in the system.
  - The log level you have set for the network output for each application.
  - The required time the log records should be kept in the database. The following table provides general timing recommendations:

Logging Level	Supported Call Volume	Recommended Storage Time
STANDARD	100 calls/sec	10 days
INTERACTION	10 calls/sec	1 day
TRACE	5 calls/sec	1 day

With these limits in mind, follow these recommendations for the Centralized Log Database:

- For efficient online log viewing, allocate temporary database space of at least 30 percent of the expected Centralized Log Database size.
- Limit permissions to modify the Centralized Log Database content to Message Servers only.
- Define how long the log records are to be kept in the database before they become obsolete. Use the Log Database Maintenance Wizard to delete obsolete records or configure the removal of obsolete records using the DBMS mechanisms.
- Users of the Centralized Log Database should have at least the following privileges for all tables in the database:
  - SELECT
  - INSERT
  - UPDATE
  - DELETE

- Make a trade-off between how long the log records are to be kept and the ability to access them efficiently. If both a considerable period of record storage and quick online access to the log records are important, back up the more dated records in a separate database.
- If you want to deploy the Disaster Recovery/Business Continuity feature, you must set up log databases across sites. Refer to Disaster Recovery/Business Continuity for more information.

SNMP Master Agent

When planning your installation, Genesys recommends that you use SNMP Master Agent only if you want to access the Management Layer functions via an NMS interface; or you have another SNMP-enabled Genesys application and want to access its features via an NMS interface.

User Interaction Layer (Genesys Administrator)

Install the Genesys Administrator web server preferably in close proximity with Configuration Server. You can then install as many web browsers as required, from which you can access and use Genesys Administrator.

Media Layer

For every switch that you plan to make a part of your interaction management solution, install at least one T-Server application.

T-Server

T-Server provides an interface between traditional telephony systems and Genesys applications.

When planning your installation, follow these recommendations for T-Server:

- At the premise level, always associate one switch with one T-Server.
- Allocate memory for T-Server based on the number of interactions you expect to be simultaneously processed at a given site during the busiest hour and the typical amount of business data attached to the interactions. Allocate at least 500 bytes per interaction plus memory space for a "typical" amount of attached data.
- Provide sufficient RAM to run T-Server processes. To ensure adequate performance, do not run T-Server processes in Swap mode.
- Do not install real-time third-party applications on the computer running T-Server.
- Consider using a dedicated subnetwork for T-Server connection to the link.
- Do not enable IP routing between the link subnet and the network when T-Server is installed on a computer with two or more network cards (one of which is used for link connection and the others for connection to the rest of the network).

Services Layer (Stat Server)

Although StatServer is considered an element of Framework, it is logical to install it when you install the solution that it will serve.

Stat Server tracks real-time states of interaction management resources and collects statistics about contact center performance. Genesys solutions use the statistical data to more "intelligently" manage interactions. Use Genesys Reporting to generate real-time and historical contact center reports based on data that Stat Server collects.

For specific recommendations on Stat Server installation, refer to Stat Server documentation.

Installation Worksheet

Use the following tables to help prepare for and perform the installation of Framework components:

- Installation Summary
- Database Information
- Licensing Information
- Application Configuration Parameters
- Windows Application Program Folders (for Windows applications only)

Installation Overview

Installation Summary	
Person responsible	
Start date	
Completion date	
Database information	Refer to Database Information
Licensing information	Refer to Licensing Information
Application configuration	Refer to Application Configuration Parameters
Program folders (for Windows applications only)	Refer to Windows Application Program Folders

Database Information

Parameter	Value		Description
Config Database	**Log Database**		
DBMS Name			The name or alias identifying the SQL server DBMS that handles the database. • For DB2, this value should be set to the name or alias-name of the database specified in the db2 client configuration. • For Microsoft SQL, this value should be

Parameter	Value		Description
			set to the name of the SQL server (usually the same as the host name of the computer on which Microsoft SQL runs). • For Oracle, it is the SID or the net service name as specified in the `tnsnames.ora` file. • For PostgreSQL, this value should be set to the name of the PostgreSQL server (usually the same as the host name of the computer on which PostgreSQL runs).
DBMS Type			The name of the database as it is specified in your DBMS. This value is required for all database types except Oracle. For DB2, Microsoft SQL, and PostgreSQL, this value is the name of the database where the client will connect.
User Name			The user name established to access the database.
Password			The password used for accessing the database.

Licensing Information

Licensing Information	
Parameter	**Value**
License Manager	
host	
port	
License Files	
Full path and filename	

Licensing Information	
Full path and filename	
Full path and filename	

Application Configuration Parameters

When completing this table, remember that:

- All applications must be configured in the Configuration Layer unless otherwise noted.
- Host name or IP address can be specified as the value for the host parameter.
- Application port and working directory are only specified for server applications.
- Working directory is the full path to the directory in which the application is installed and/or is to be running.

Application Type	Application Name	Application Host	Application Port	Working Directory
Configuration Layer Components				
Configuration Server, Primary, for Configuration Database (configured via configuration file)				
Configuration Server, Backup, for Configuration Database (configured via configuration file)				
Management Layer Components				
Local Control Agent	Not applicable		Configured in Host properties	Not applicable
Database Access Point		Not applicable		
Message Server, Primary				
Message Server, Backup				
Solution Control Server, Primary				
Solution Control Server, Backup				
SNMP Master Agent, Primary				

Application Type	Application Name	Application Host	Application Port	Working Directory
SNMP Master Agent, Backup				
User Interaction Layer Components				
Genesys Administrator			Not applicable	
Media Layer Components				
T-Server, Primary, for switch ...				
T-Server, Backup, for switch ...				
T-Server, Primary, for switch ...				
T-Server, Backup, for switch ...				
Services Layer Components				
Stat Server, Primary				
Stat Server, Backup				

Windows Application Program Folders

Application	Application Program Folder
Configuration Layer Components	
Configuration Server, Primary, for Configuration Database (configured via configuration file)	
Configuration Server, Backup, for Configuration Database (configured via configuration file)	
Management Layer Components	
Local Control Agent	
Database Access Point	
Message Server, Primary	
Message Server, Backup	
Solution Control Server, Primary	
Solution Control Server, Backup	
SNMP Master Agent, Primary	
SNMP Master Agent, Backup	
User Interaction Layer Components	
Genesys Administrator	

Application	Application Program Folder
Media Layer Components	
T-Server, Primary, for switch ...	
T-Server, Backup, for switch ...	
T-Server, Primary, for switch ...	
T-Server, Backup, for switch ...	
Services Layer Components	
Stat Server, Primary	
Stat Server, Backup	

Deploying Framework

This section of the Framework Deployment Guide lists the prerequisites for installing the Genesys Framework, and prescribes the deployment order. It then describes how to install and configure the Management Framework components.

> **Tip**
> Use the sample worksheet as you prepare for and perform the Framework installation.

Deployment Overview

The various Framework components are distributed on the following product CDs

- Management Framework
- Genesys Administrator
- Media
- HA Proxy
- Real-Time Metrics Engine

The Framework deployment process involves the configuration and installation of one or more components of the same type within each architecture layer, as outlined here.

Sequence

Deploy Framework components in the following order:

1. Bootstrap components:
    a. Configuration Database
    b. Configuration Server (master/primary instance)
    c. Genesys Administrator/Extension
    d. Local Control Agent (LCA) on the hosts on which Configuration Server is installed
    e. Solution Control Server (SCS) (master/primary instance)
6. Configuration components:
    a. (Optional) High Availability (HA) Configuration Server pair (primary/backup instances)
    b. (Optional) Configuration Server Proxies
3. Management components
    a. LCAs on all of the other hosts on which will be running Genesys server applications and/or monitored third-party server applications
    b. Message Server
    c. Centralized Log Database
    d. (Optional) HA SCS pair (primary/backup instances)
    e. (Optional) Distributed SCS
    f. (Optional) SNMP Master Agent (Simple Network Management Protocol)
7. Media Layer components

a. T-Server

b. HA Proxy for a specific type of T-Server (if required)

> **Important**
>
> Configuration and installation instructions for T-Servers apply to Network T-Servers as well. You can find detailed deployment information about T-Server and HA Proxy in the latest version of the T-Server Deployment Guide for your specific T-Server.

3. Services Layer component: Stat Server

> **Important**
>
> Although Interaction Server, SMCP (Simple Media Control Protocol) T-Server, and Stat Server components are all parts of the Framework architecture, configuring them directly depends on their usage in a Genesys solution. Therefore, you must install them during deployment of a specific solution.

Creation of Configuration Objects

In addition to installed Framework components, the following resources must be registered as Configuration Database objects (or configuration objects) at the time of Framework deployment:

- Hosts
- Switching Offices
- Switches
- Agent Logins
- DNs
- Access Groups
- Skills
- Persons
- Agent Groups
- Places
- Place Groups

To deploy components of the Configuration Layer, you must first configure the objects and then install them, as described later in these pages.

> **Warning**
> Never add, delete, or modify any data in the Configuration Database, except through applications developed by Genesys, or through applications instrumented with the Genesys Configuration Server application programming interface (API). If you have compelling reasons for accessing the database directly, consult Genesys Customer Care before you do so.

Windows Authentication for MS SQL Databases

Prior to release 8.5.1, Management Framework components depended on SQL Server Authentication to authenticate an MS SQL database user. The username and password of a Windows user was passed over the network, and the user identity was confirmed by Windows.

Starting in release 8.5.1, Configuration Server and Message Server can use Windows Authentication to authenticate Windows users and give them access to an MS SQL database, namely the Configuration Database and the Log Database, respectively. Windows Authentication is much more secure than SQL Server Authentication. Windows Authentication uses the Kerberos security protocol, enforces password policies to ensure strong passwords, and supports account lockout and password expiration. A connection made using Windows Authentication is sometimes called a *trusted* connection, because SQL Server trusts the credentials provided by Windows.

Refer to [[Documentation:FR:DBConn:MSSQL|Windows Authentication}} for more information about how to configure applications to use Windows Authentication to access MS SQL databases.

> **Important**
> Currently, only Configuration Server 8.5.1 and Message Server 8.5.1 can use Windows Authentication to access an MS SQL database, namely the Configuration Database and the Log Database, respectively.

Using DB Server

Starting in release 8.5, databases are accessed directly by the servers that need to store and/or retrieve data in them, removing the need to install DB Server. However, you can still use DB Server as in previous releases, if you have legacy components that require DB Server or you are unable to configure Genesys components to access databases from their local hosts.

> **Tip**
> Genesys strongly recommends that you use newer components that support direct

> database access. If you use DB Server, make sure that you install the alternate set of database client processes (available in the DB Server 8.1.3 Installation Package) for use with Framework 8.5. Refer to the DB Server 8.1.x Release Note for more information.

For Configuration Server to access the Configuration Database through DB Server, set the **dbthread** configuration option to `false` in the **[confserv]** section of the primary Configuration Server, and in the appropriately-named section of the backup server, if configured. For Message Server to access the Log Database through DB Server, set the **dbthread** configuration option to `false` in the **[messages]** section of the primary Message Server and the backup Message Server, if configured. Refer to the *Framework Configuration Options Reference Manual* for more information about these options. For other products supporting this approach database access, refer to the product-specific documentation for the option name and instructions.

Then refer to the Framework 8.1 documentation for information about deploying and using DB Server to access the various databases in your environment.

Refer to the *Framework Database Connectivity Reference Guide* for detailed information about setting up and accessing a database.

Prerequisites

Before you deploy Framework, investigate aspects of its size, security, availability and performance, as applied to the specific environment of your contact center. See Deployment Planning for recommendations on these issues. Be sure those applications that require licenses are licensed properly (see the *Genesys Licensing Guide*).

Review the following prerequisites for your Framework installation:

- Databases
- Operating Environment, including hardware, networking, software, and internet browsers
- Licensing
- SNMP
- Minimum Permissions

For prerequisites for Genesys Administrator, refer to the *Framework 8.1 Genesys Administrator Deployment Guide*.

Database Prerequisites

Genesys recommends that you or your database administrator create databases in your database management system (DBMS) before you start a Genesys installation. For Framework, you must create two databases:

- Configuration Database-Mandatory for any Genesys installation.
- Centralized Log Database-Required only if you are using the Management Layer's centralized-logging function.

Genesys also recommends that you or your database administrator back up your Genesys databases on a regular basis.

Refer to Network Locations for Framework Components for recommendations on database sizing. Refer to your DBMS documentation for instructions on how to create a new database.

Refer to the Installation Worksheet for the list of database parameters you must use in your Genesys installation.

Refer to Minimum_Database_Privileges for the minimum database privileges required to deploy and access each of the two databases.

Creating Databases for Multi-language Environments

If your system will be configured in a multi-language environment, or at least be required to handle data encoded in UTF-8 format, you may have to take special steps when creating your database, depending on the DBMS you will be using. Refer to the table in the "Creating Databases for Multi-language Environments" section of the *Framework Database Connectivity Reference Guide* for these additional steps.

Creating Databases for Single-language Environments

If your system will be configured in a single-language environment, you must make sure that the encoding used in their database matches the encoding set up on all hosts. If they have to perform some type of conversion, because they are unable to store data in the same encoding specified at the operation system level, they must depend on the DBMS capabilities of the vendor client software.

Operating Environment Prerequisites

Hardware and Networking

Genesys recommends that you or your IT specialist assign host computers to Genesys software before you start Genesys installation.

If you are considering using IPv6 for some or all connections, make sure that you first review the information in IPv6.

Refer to Network Locations for Framework Components for recommendations on server locations.

Software

Refer to the *Genesys Supported Operating Environment Reference Guide* for the list of operating systems and database systems supported in Genesys releases. Refer to the *Genesys Supported Media Interfaces Reference Guide* for the list of supported switch and PBX versions.

For UNIX operating systems, also review the list of patches Genesys uses for software product builds and upgrade your patch configuration if necessary. A description of patch configuration is linked to the Readme files for the Genesys applications that operate on UNIX.

Refer to Minimum_System_Permissions for the minimum permissions required to deploy and run Management Framework components on Windows and UNIX platforms.

Internet Browsers

To view all elements of Genesys Administrator, you need any combination of Internet Explorer, Mozilla Firefox, and Google Chrome internet browsers.

Refer to the *Framework Genesys Administrator Deployment Guide* for information about supported browser versions, and requirements for the Genesys Administrator web server.

Licensing Prerequisites

Genesys applications require licenses. There are two aspects of licensing for Genesys software:

- Sellable Item licenses, which provide the legal right to deploy and use the components and functionality related to the solutions that you purchased from Genesys.
- Technical license keys, which enable you to use particular functionality. For example, if you are planning to deploy redundant configurations of any Genesys servers, you must have a special high-availability (HA) license. Otherwise, the Management Layer will not perform a switchover between the primary and backup servers. As another example, a technical license is required to deploy Configuration Server Proxy.

Technical licenses are managed and controlled by the FlexNet Publisher (formerly FlexLM) License Manager.

For information about which products require what types of licenses, refer to the *Genesys Licensing Guide*. That Guide also describes how to install FlexNet Publisher License Manager, if required, before you start to deploy Management Framework.

License File

Genesys Technical licensing is based on a valid license file that contains license key information required to operate Genesys components, where required. Request the initial license file from Genesys after you purchase your Genesys system, when you are ready to deploy Management Framework. FlexNet Publisher uses the license keys in the license file to enable certain components to operate.

License Files and the Configuration Database

> **Important**
> Configuration Server no longer requires License Resource Manager (LRM) 8.5 to be present in the environment at any time, and the following step can be skipped when deploying Configuration Server, 8.5.101.00 and later.

Configuration Server stores the license file in the Configuration Database. When you are setting up your new Genesys environment, or when you have just upgraded your Configuration Database in an existing Genesys environment, you must upload the license file into the Database before you start Configuration Server for the first time. To do this, run the following command on the command-line, as appropriate:

- On UNIX: `confserv –license <license file name>`

- On Windows: `confserv.exe –license <filename>`

After that, you can start Configuration Server.

SNMP Prerequisites

Starting in release 8.5.1, you can use Net-SNMP, instead of the Genesys SNMP Master Agent component, to implement SNMP functionality. Net-SNMP supports everything that Genesys SNMP Master Agent does. Genesys strongly recommends that you use the newest version of Net-SNMP Master Agent to make your SNMP configuration as robust as possible.

Net-SNMP and Genesys SNMP Master Agent can run in parallel on the same system, if the following conditions are met:

- Only one SNMP Master Agent can be configured on each Solution Control Server (SCS).
- The ports used by each SNMP Master Agent must be unique, and not used by any other application.

Solution Control Server and Local Control Agent support both implementations.

Installing Net-SNMP

If you choose to use Net-SNMP in addition to, or instead of, Genesys SNMP Master Agent, you must install Net-SNMP on all hosts where you need an SNMP Master Agent to reside. Use the following instructions, according to the operating system running on that host.

Windows 64-bit

To install and configure Net-SNMP on Windows 64-bit Platforms, use the following steps:

1. Download and run Net-SNMP as follows:

a. In a web browser, go to http://sourceforge.net/projects/net-snmp/files/net-snmp%20binaries/ and download **net-snmp-<version>.<version>.x64.exe**, where <version> is the latest stable version of the file available.

b. Run the downloaded file, making the following selections on the **Choose Components** screen of the Net-SNMP <version> Setup Wizard:

- Expand **Net-SNMP Agent Services** and select **With Windows Explorer**
- **Net-SNMP Trap Services**
- **Perl SNMP Modules**

All binaries are created in the installed folder specified in the **Choose Default Location** screen of the setup wizard.

2. Register (or unregister, if required) the Net-SNMP Services by running the appropriate **.bat** file, all of which are located in the Net-SNMP installation folder.:

To Do This ...	To This Service ...	Run this File
Register	Net-SNMP agent	registeragent.bat
	Net-SNMP trap handler	registertrapd.bat
Unregister	Net-SNMP agent	unregisteragent.bat
	Net-SNMP trap handler	unregistertrapd.bat

After registration, the files are located in the following locations: **Note:** The file locations given in the table are by default, based on the default location specified in Step 1.

Files	Location
snmpd.conf	C:\usr\etc\snmp\snmpd.conf
snmpd.conf (persistent)	C:\usr\snmp\persist\snmpd.conf
snmptrapd.conf	C:\usr\etc\snmp\snmptrapd.conf
Log files	C:\usr\log\snmpd.log

3. Modify the Net-SNMP configuration file, **snmpd.conf** to define system-specific parameters, as follows:

a. To open the UDP port through which SCS can communicate with the SNMP Master Agent, add the following line to the configuration file:

```
agentaddress udp:<port_num>
```

where <port_num> is any valid port number. This port must not be used by any other process.

You can check if the port has been opened successfully by running the following command:

```
C:\>netstat –aon | findstr <port_num>
```

If it shows the port, then the port is open.

b. To enable AgentX functionality and open the AgentX port, add the following lines to the configuration file:

```
master agentx
agentxsocket tcp:<host_address>:<port_num>
```

where:

- <host_address> is the IP address of the host on which Net-SNMP is running.
- <port_num> is any valid port number. This port must not be used by any other process.

c. Configure the SNMP version. Do one of the following:

- For SNMP v1 or SNMP v2, add the following lines to the configuration file:

```
rocommunity public
rwcommunity private
```

These parameters are similar to the **read_community** and **write_community** configuration options in an SNMP Master Agent.

- For SNMP v3, add the following lines to the configuration file:

```
# VACM configuration entries
rwuser <username>
```

```
# create and add a user
createUser <username> MD5 <md5_password> <DES des_password>
```

You can check if SNMP v3 has been configured successfully by running the following command:

```
C:\>snmpget -v3 -u <username> -n "" -l authNoPriv -a MD5 -A <md5_password>
<localhost> sysUpTime.0
```

If any output is displayed, then SNMP v3 has been successfully configured.

You can also create another user based on the initial user by running the following command:

```
C:\>snmpusm -v3 <username> -u initial -n "" -l authNoPriv -a MD5 -A <md5_password>
<localhost> create <newuser> initial
```

4. Configure the MIB Browser for the version of SNMP you are using. Do one of the following:

For SNMP v1 or SNMP v2, set the following:

- SNMP version
- Number of port on which **snmpd** is running

For SNMP v3, set the following:

- SNMP version
- User profile name
- Security user name
- Number of SNMP port on which **snmpd** is running
- Authentication protocol
- Privacy protocol

5. Modify the Net-SNMP configuration file, **snmpd.conf**, to define where SNMP traps are to be sent, as follows:

For SNMP v1 traps, add the line:

```
trapsink <transport>:<host_address>:<port>
```

where:

- `<transport>` can be `udp` or `tcp`.
- `<host_address>` is the IP address of the host to where SNMP traps are to be sent.
- `<port>` is the SNMP port on which traps are to be received.

For SNMP v2 traps, add the line:

```
trap2sink <transport>:<host_address>:<port>
```

where:

- `<transport>` can be `udp` or `tcp`.
- `<host_address>` is the IP address of the host to where SNMP traps are to be sent.

- <port> is the SNMP port on which traps are to be received.

For SNMP v3 traps, add the line:

```
trapsess -v 3 –u <username> -l authNoPriv -Ci -a MD5 -A "<authentication_password>"
<host_address>:<port>
```

where:

- <host_address> is the IP address of the host to where SNMP traps are to be sent.
- <port> is the SNMP port on which traps are to be received.

6. Prepare start and stop scripts to start and stop Net-SNMP as a service, as shown below. The path and filename of these scripts are stored in the annex of the SNMP Master Agent Application objects.

Start script, for example **startsnmp.bat**

```
net start "<netsnmp_servicename>"
net start "<netsnmptrap_servicename>"
```

Stop script, for example **stopsnmp.bat**

```
net stop "<netsnmp_servicename>"
net stop "<netsnmptrap_servicename>"
```

Linux 64-bit

To install and configure Net-SNMP on Linux 64-bit Platforms, do the following: Prerequisites:

- You must be logged in as root.
- The Red Hat Package Manager (RPM) files (RPMs) must have execute permission. If you are unsure, give full permission by running the following command:

```
$chmod –R 777 *
```

1. Download and install the RPM files, as follows:

a. Download the following RPM files from http://rpmfind.net:

- **net-snmp-libs-<version>.el5**
- **net-snmp-<version>.el5>**

where <version> is the latest stable version of the files available.

b. Install the rpm files using the command:

```
$rpm –ivh <rpm file name>
```

The installation places the Net-SNMP files in the following locations, unless otherwise specified during the installation:

Files	Location
snmpd.conf	/usr/local/etc/snmp/snmpd.conf
snmpd.conf (persistent)	/var/net-snmp/snmpd.conf
snmptrapd.conf	/var/net-snmp/snmptrapd.conf
Log file	/var/log/snmpd.log
Snmpd.exe (application)	/usr/local/sbin/snmpd
Snmptrapd.exe (application)	/usr/local/sbin/snmptrapd
Init.d script required for running snmpd and snmptrapd	/etc/init.d/snmpd

2. Modify the Net-SNMP configuration file, **snmpd.conf** to define system-specific parameters, as follows:

a. To open the UDP port through which SCS can communicate with the SNMP Master Agent, add the following line to the configuration file:

```
agentaddress <transport>:<port_num>
```

where:

- <transport> is the type of port to be opened, either udp or tcp. A UDP port is preferred, since it is faster than TCP.
- port_num is any valid port number. This port must not be used by any other process.

You can check if the port has been opened successfully by running the following command:

```
$netstat –aon | findstr <port_num>
```

If it shows the port, then the port is open.

b. To enable agentx functionality and open the agentX socket, add the following lines to the configuration file:

```
master agentx
Agentxsocket tcp:<host_address>:<port_num>
```

where:

- <host_address> is the IP address of the host on which Net-SNMP is running.
- <port_num> is any valid port number. This port must not be used by any other process.

c. Configure the SNMP version. Do one of the following:

- For SNMP v1 or SNMP v2, add the following lines to the configuration file:

```
rocommunity public
rwcommunity private
```

These parameters are similar to the **read_community** and **write_community** configuration options in an SNMP Master Agent.

- For SNMP v3, add the following lines to the configuration file:

```
# VACM configuration entries
```

```
rwuser username
# create and add a user
createUser username MD5 md5_password DES des_password
```

You can check if SNMP v3 has been configured successfully by running the following command:

```
$snmpget -v3 -u <username> -n "" -l authNoPriv -a MD5 -A <md5_password>
 <localhost> sysUpTime.0
```

If any output is displayed, then SNMP v3 has been successfully configured.

You can also create another user based on the initial user by running the following command:

```
$snmpusm -v3 <username> -u initial -n "" -l authNoPriv -a MD5 -A <md5_password>
 <localhost> create <newuser> initial
```

3. Configure the MIB Browser for the version of SNMP you are using. Do one of the following:

For SNMP v1 or SNMP v2, set the following:

- SNMP version
- Number of port on which **snmpd** is running

For SNMP v3, set the following:

- SNMP version
- User profile name
- Security user name
- Number of SNMP port on which **snmpd** is running
- Authentication protocol
- Privacy protocol

4. Modify the Net-SNMP configuration file, **snmpd.conf**, to define where SNMP traps are to be sent, as follows:

For SNMP v1 traps, add the line:

`trapsink <transport>:<host_address>:<port>`

where:

- `<transport>` is the type of port through which the traps will be sent, either `udp` or `tcp`. A UDP port is preferred, since it is faster than TCP.
- `<host_address>` is the IP address of the host to where SNMP traps are to be sent.
- `<port>` is the SNMP port on which traps are to be received.

For SNMP v2 traps, add the line:

```
trap2sink <transport>:<host_address>:<port>
```

where:

- <transport> is the type of port through which the traps will be sent, either udp or tcp. A UDP port is preferred, since it is faster than TCP.
- <host_address> is the IP address of the host to where SNMP traps are to be sent.
- <port> is the SNMP port on which traps are to be received.

For SNMP v3 traps, add the line:

```
trapsess -v3 –u <username> -l authNoPriv -Ci -a MD5 -A "<authentication_password>"
<host_address>:<port>
```

where:

- <host_address> is the IP address of the host to where SNMP traps are to be sent.
- <port> is the SNMP port on which traps are to be received.

5. Prepare start and stop scripts, based on the commands provided in Starting and Stopping Net-SNMP. The path to these scripts are stored in the annex of the SNMP Master Agent Application objects.

AIX 64-bit

To install and configure Net-SNMP on AIX 64-bit Platforms, do the following: Prerequisites:

- You must be logged in as root.
- The RPM files must have execute permission. If you are unsure, give full permission by running the following command:

```
$chmod –R 777 *
```

1. Download and install the RPM files, as follows:

a. Download the following RPM files from ftp://www.oss4aix.org/RPMS/net-snmp/:

- **net-snmp-<version>.ppc.rpm**
- **net-snmp-utils-<version>.ppc.rpm**

where <version> is the latest stable version of the files available.

b. Install the rpm files using the command:

```
$rpm –ivh <rpm file name>
```

The installation places the Net-SNMP files in the following locations, unless otherwise specified during the installation:

Files	Location
snmpd.conf	/opt/freeware/etc/snmp/snmpd.conf

Files	Location
snmptrapd.conf	/opt/freeware/etc/snmp/snmptrapd.conf
Log file	/var/log/snmpd.log
Snmpd (application)	opt/freeware/sbin/snmpd
Snmptrapd.exe (application)	opt/freeware/sbin/snmptrapd
Init.d script required for running snmpd	/etc/rc.d/init.d/snmpd
Init.d script required for running snmptrapd	/etc/rc.d/init.d/snmptrapd

2. Add the path of the snmpd binary to the PATH environment variable, as follows:

a. To add the path (**opt/freeware/sbin/** if using the default locations) to the path, enter:

```
PATH=$PATH:opt/freeware/sbin
```

b. To confirm the result, enter:

```
echo $PATH
```

The new path, including the path to the Snmpd binary, is displayed.

3. Modify the Net-SNMP configuration file, **snmpd.conf**, to define system-specific parameters, as follows:

a. To open the port through which SCS communicates with the SNMP Master Agent, add the following line to the configuration file:

```
agentaddress <transport>:<port_num>
```

where:

- <transport> is the type of port to be opened, either udp or tcp . A UDP port is preferred, since it is faster than TCP.
- port_num is any valid port number. This port must not be used by any other process.

You can check if the port has been opened successfully by running the following command:

```
$netstat –aon | findstr <port_num>
```

If it shows the port number, then the port is open.

b. To enable agentx functionality and open the agentX port, add the following lines to the configuration file:

```
master agentx
Agentxsocket tcp:<host_address>:<port_num>
```

where:

- <host_address> is the IP address of the host on which Net-SNMP is running.
- <port_num> is any valid port number. This port must not be used by any other process.

c. Configure the SNMP version. Do one of the following:

- For SNMP v1 or SNMP v2, add the following lines to the configuration file:

```
rocommunity public
rwcommunity private
```

  These parameters are similar to the **read_community** and **write_community** configuration options in an SNMP Master Agent.

- For SNMP v3, add the following lines to the configuration file:

```
# VACM configuration entries
rwuser username
# create and add a user
createUser username MD5 md5_password DES des_password
```

  You can check if SNMP v3 has been configured successfully by running the following command:

```
$snmpget -v 3 -u <username> -n "" -l authNoPriv -a MD5 -A <md5_password>
<localhost> sysUpTime.0
```

  If any output is displayed, then SNMP v3 has been successfully configured.

  You can also create another user based on the initial user by running the following command:

```
$snmpusm -v3 <username> -u initial -n "" -l authNoPriv -a MD5 -A <md5_password>
<localhost> create <newuser> initial
```

4. Configure the MIB Browser for the version of SNMP you are using. Do one of the following:

For SNMP v1 or SNMP v2, set the following:

- SNMP version
- Port number on which **snmpd** is running

For SNMP v3, set the following:

- SNMP version
- User profile name
- Security user name
- SNMP port number, the port on which **snmpd** is running
- Authentication protocol
- Privacy protocol

5. Modify the Net-SNMP configuration file, **snmpd.conf**, to define where SNMP traps are to be sent, as follows:

For SNMP v1 traps, add the line:

```
trapsink <transport>:<host_address>:<port>
```

where:

- <transport> is the type of port through which the traps will be sent, either udp or tcp. A UDP port is preferred, since it is faster than TCP.
- <host_address> is the IP address of the host to where SNMP traps are to be sent.
- <port> is the SNMP port on which traps are to be received.

For SNMP v2 traps, add the line:

```
trap2sink <transport>:<host_address>:<port>
```

where:

- <transport> is the type of port through which the traps will be sent, either udp or tcp. A UDP port is preferred, since it is faster than TCP.
- <host_address> is the IP address of the host to where SNMP traps are to be sent.
- <port> is the SNMP port on which traps are to be received.

For SNMP v3 traps, add the line:

```
trapsess -v 3 –u <username> -l authNoPriv -Ci -a MD5 -A "<authentication_password>"
<host_address>:<port>
```

where:

- <host_address> is the IP address of the host to where SNMP traps are to be sent.
- <port> is the SNMP port on which traps are to be received.

6. Prepare start and stop scripts, based on the commands in Starting and Stopping Net-SNMP. The path to these scripts are stored in the annex of the SNMP Master Agent Application objects.

Solaris 10 64-bit

To install and configure Net-SNMP on Solaris 10 64-bit Platforms, do the following: Prerequisites:

- You must be logged in as root, with execute permission. If you are unsure, give full permission by running the following command:

```
$chmod –R 777 *
```

1. Download and install the Net-SNMP package files, as follows:

a. Download the **netsnmp-5.7-sol10-sparc-local.gz** file from ftp://sunfreeware.saix.net/pub/solaris-freeware/sparc/10/, and extract it with the following command:

```
$gunzip netsnmp-5.7-sol10-sparc-local.gz
```

b. Install the extracted file as follows:

```
$pkgadd –d netsnmp-5.7-sol10-sparc-local
```

If the error message Pkgadd `command not found` appears, update the PATH environment variable, as follows:

```
$PATH=$PATH:/usr/sfw/bin:/usr/ccs/bin:/usr/sbin
```

To view the list of installed packages, enter the following command:

```
$pkginfo
```

To view the list of files installed by the packages, enter the following command:

```
$pkgchk –v <packagename>
```

To determine what file belongs to which package, enter the following command:

```
$ pkgchk -l -p <path_to_file/filename>
```

The installation places the Net-SNMP files in the following locations, unless otherwise specified during the installation:

Files	Location
snmpd.conf	/etc/opt/csw/snmp/snmpd.conf
snmptrapd.conf	/etc/opt/csw/snmp/snmptrapd.conf
Log file	/var/opt/csw/log/snmpd.log
Snmpd (application)	opt/csw/sbin/snmpd
Snmptrapd.exe (application)	opt/csw/sbin/snmptrapd
Init.d script required for running snmpd	/etc/rc.d/init.d/snmpd
Init.d script required for running snmptrapd	/etc/rc.d/init.d/snmptrapd

2. Add the path of the Snmpd binary to the PATH environment variable, as follows:

a. To add the path (**opt/csw/sbin** if using the default locations) to the path, enter:

```
PATH=$PATH:opt/csw/sbin
```

b. To confirm the result, enter:

```
echo $PATH
```

The new path, including the path to the Snmpd binary, is displayed.

Note: If you log in with a new session, you must add the path again.

3. Modify the Net-SNMP configuration file, **snmpd.conf**, to define system-specific parameters, as follows:

a. To open the port through which SCS communicates with the SNMP Master Agent, add the following line to the configuration file:

```
agentaddress <transport>:<port_num>
```

where:

- <transport> is the type of port to be opened, either udp or tcp. A UDP port is preferred, since it is faster than TCP.
- port_num is any valid port number. This port must not be used by any other process.

You can check if the port has been opened successfully by running the following command:

```
$netstat –aon | findstr <port_num>
```

If it shows the port, then the port is open.

b. To enable agentx functionality and open the agentX port, add the following lines to the configuration file:

```
master agentx
Agentxsocket tcp:<host_address>:<port_num>
```

where:

- <host_address> is the IP address of the host on which Net-SNMP is running.
- <port_num> is any valid port number. This port must not be used by any other process.

c. Configure the SNMP version. Do one of the following:

- For SNMP v1 or SNMP v2, add the following lines to the configuration file:

  ```
  rocommunity public
  rwcommunity private
  ```

  These parameters are similar to the **read_community** and **write_community** configuration options in an SNMP Master Agent.

- For SNMP v3, add the following lines to the configuration file:

  ```
  # VACM configuration entries
  rwuser username
  # create and add a user
  createUser username MD5 md5_password DES des_password
  ```

  You can check if SNMP v3 has been configured successfully by running the following command:

  ```
  $snmpget -v 3 -u <username> -n "" -l authNoPriv -a MD5 -A <md5_password>
  <localhost> sysUpTime.0
  ```

  If any output is displayed, then SNMP v3 has been successfully configured.

  You can also create another user based on the initial user by running the following command:

  ```
  $snmpusm -v3 <username> -u initial -n "" -l authNoPriv -a MD5 -A <md5_password>
  ```

```
<localhost> create <newuser> initial
```

4. Configure the MIB Browser for the version of SNMP you are using. Do one of the following:

For SNMP v1 or SNMP v2, set the following:

- SNMP version
- Port number on which **snmpd** is running

For SNMP v3, set the following:

- SNMP version
- User profile name
- Security user name
- SNMP port number, the port on which **snmpd** is running
- Authentication protocol
- Privacy protocol

5. Modify the Net-SNMP configuration file, **snmpd.conf**, to define where SNMP traps are to be sent, as follows:

For SNMP v1 traps, add the line:

```
trapsink <transport>:<host_address>:<port>
```

where:

- <transport> is the type of port through which the traps will be sent, either udp or tcp. A UDP port is preferred, since it is faster than TCP.
- <host_address> is the IP address of the host to where SNMP traps are to be sent.
- <port> is the SNMP port on which traps are to be received.

For SNMP v2 traps, add the line:

```
trap2sink <transport>:<host_address>:<port>
```

where:

- <transport> is the type of port through which the traps will be sent, either udp or tcp. A UDP port is preferred, since it is faster than TCP.
- <host_address> is the IP address of the host to where SNMP traps are to be sent.
- <port> is the SNMP port on which traps are to be received.

For SNMP v3 traps, add the line:

```
trapsess -v 3 –u <username> -l authNoPriv -Ci -a MD5 -A "<authentication_password>"
<host_address>:<port>
```

where:

- <host_address> is the IP address of the host to where SNMP traps are to be sent.
- <port> is the SNMP port on which traps are to be received.

6. Prepare start and stop scripts, based on the commands provided in Starting and Stopping Net-SNMP. The path to these scripts are stored in the annex of the SNMP Master Agent Application objects.

Starting and Stopping Net-SNMP

If you have installed Net-SNMP on a Windows 64-bit platform, start and stop Net-SNMP as a service, using Windows Service Manager.

Otherwise, use the following commands to start or stop Net-SNMP running on a UNIX host:

Action	Command
Start Net-SNMP	`$service snmpd start`
Stop Net-SNMP	`$service snmpd stop`
Check Net-SNMP status	`$service snmpd status`
Restart Net-SNMP	`$service snmpd restart`

Antivirus Guidelines

Antivirus software can affect system performance and call response time, but it prevents viruses from hiding in software. Genesys recommends keeping antivirus software enabled on hosts where Management Framework components are running.

Performance of all applications on a particular host should be analyzed. There might be more vulnerable than Management Framework applications. Some of those vulnerable applications can be considered to be moved to a different host. Genesys does not recommend excluding Management Framework from the antivirus scanning, but in case of a significant overhead, you can consider excluding the following from the scan:

- The folder in which Management Framework components are running.
- Any Management Framework applications' logs folders.
- The Management Framework applications running on Windows/Linux.

The antivirus software must not restrict any ports that are used by the Genesys applications.

Deploying Configuration Layer

The Framework Configuration Layer is a mandatory part of any Genesys installation and is the first step of the Framework deployment.

Important

Before you install Framework components:

- Refer to Network Locations for Framework Components for recommendations on the network locations of these components.
- Create a new database following the instructions in your DBMS documentation.

Warning

During installation on UNIX, all files are copied into a user-specified directory. The installation creates no subdirectories within this directory, so be careful to not install different products into the same directory.

If you are installing the Framework for the first time, follow the instructions for a first time deployment. If you are upgrading your Genesys Framework, refer to the *Framework Migration Guide*. Otherwise, to install new individual components, follow the instructions on the following pages to install the appropriate Configuration Layer component.

After you have a successfully installed and configured Configuration Layer components, you can implement any of the following, as appropriate:

- Enable the Management Layer to control the Configuration Layer (recommended)
- Encrypt a password for the Configuration Database (recommended)
- Configure a user inactivity timeout to disable logged-in users after a period of inactivity. Refer to the *Genesys Security Deployment Guide* for instructions.
- Configure redundant Configuration Servers in HA pairs.
- Configure one or more Configuration Server Proxies.

First-Time Deployment

To install the Framework Configuration Layer components for the first time, follow these steps:

1. Install Configuration Server
2. Initialize the Configuration Database
3. Configure Configuration Server
4. Start Configuration Server
5. Deploy and start Genesys Administrator
6. Create Hosts for each computer in your network

Configuration Database

After you have created a database in your DBMS (see Databases), you can populate the tables of the Configuration Database manually (using your DBMS tools).

Setting Up the Configuration Database

Important

If you install Configuration Server and the Configuration Database separately, you must install and configure an SQL Server client for your database type on the same host where Configuration Server is running. Refer to the *Framework Database Connectivity Reference Guide* for recommendations on environment settings for your database client.

Warning

Configuration Server treats its information and checks integrity constraints in a case-sensitive manner. Therefore, your SQL database must be installed and configured in case-sensitive mode. Refer to your SQL Server Administrator documentation for additional information.

1. In the directory in which Configuration Server is installed, open the **sql_scripts** folder.
2. Open the folder that matches your database type.
3. If you are initializing a database that contains existing or old configuration data, remove that data now, before running the initialization scripts, as follows:
   a. Back up your database if required.

      ### Warning

      All existing configuration data will be lost after running the **drop_tables_<DB_Type>.sql** script.

   b. Load and execute the **drop_tables_<DB_Type>.sql** script that corresponds to your DBMS, as listed in the table below.

DBMS	Script Name
DB2 See note below	drop_tables_db2.sql

DBMS	Script Name
Microsoft SQL	drop_tables_mssql.sql
Oracle	drop_tables_ora.sql
PostgreSQL	drop_tables_postgre.sql
Note: Genesys recommends using the DB2 Command-Line Processor to run Genesys SQL scripts.	

4. Load and execute the initialization script that corresponds to your DBMS, as listed in the table below.

DBMS	Single-language Script Name	Multi-language Script Name
DB2 See note below	init_multi_db2.sql	init_multi_multilang_db2.sql
Microsoft SQL	init_multi_mssql.sql	init_multi_multilang_mssql.sql
Oracle	init_multi_ora.sql	init_multi_multilang_ora.sql
PostgreSQL	init_multi_postgre.sql	init_multi_multilang_postgre.sql
Note: Genesys recommends using the DB2 Command-Line Processor to run Genesys SQL scripts.		

5. Load and execute the script that loads the **CfgLocale** table into the initialized database, depending on your database type, as shown in the table below.

DBMS	Script Name
DB2 See note below	CfgLocale_db2.sql
Microsoft SQL	CfgLocale_mssql.sql
Oracle	CfgLocale_ora.sql
PostgreSQL	CfgLocale_postgre.sql
Note: Genesys recommends using the DB2 Command-Line Processor to run Genesys SQL scripts.	

Important

If you are using a PostgreSQL database, make sure that the value of the PostgreSQL configuration option **standard_conforming_strings** in the **postgresql.conf** file matches that of the Configuration Server Application option **postgre-standard-conforming-strings**, in the **[system]** section. For more information, refer to the *Framework Database Connectivity Reference Guide* and the *Framework Configuration Options Reference Manual*.

Using the DB2 Command-Line Processor

If you are using DB2, Genesys recommends using the DB2 Command-Line Processor to load and execute the script, as follows:

1. Start the Command-Line Processor.
2. Type `quit` at the DB2 prompt to exit the **DB2.exe** process.
3. Specify the database connection parameters by typing the following command line, substituting values in brackets with the actual values:

```
db2 connect to <database name> user <user> using <password>
```

4. Execute the script by typing the following command line, substituting the value in brackets with the actual value:
   `db2 -f <script name including full path>`
   For example, to execute the `CfgLocale` script, type (all on one line):

```
db2 -f C:\GCTI\ConfigurationServer\sql_scripts\db2\CfgLocale_db2.sql
```

About the Initialized Configuration Database

> ### Warning
> Never add, delete, or modify any data in the Configuration Database except through applications developed by Genesys, or through applications instrumented with the Genesys Configuration Server application programming interface (API). If you have compelling reasons for accessing the database directly, consult Genesys Technical Support before you do so.

The Configuration Database contains the following predefined objects, which allow initial access to the database through Genesys Administrator:

- A User object with user name set to `default`, and password set to `password`. Use this Master Account to log in to the Configuration Layer for the first time. A user logged on through this Master Account has all possible privileges with respect to objects in the Configuration Database. The Master Account is not alterable in any way, and you should not use it to perform regular contact center administrative tasks. Rather, it exists as a guarantee that, no matter what happens to the regular accounts, you will always be able to access the Configuration Database.
  Genesys recommends changing the default user name and password of the Master Account during the first session, securing these login parameters, and using the Master Account for emergency purposes only. For regular operations, create a real working account and add it to the access group Super Administrators. (By default, this Access Group has the same privileges as the Master Account.) Use this real working account for any subsequent sessions.

  > ### Important
  > For instructions on creating new configuration objects, and working with existing configuration objects, refer to the Help file with your user interface.

- Four `Application Template` objects, as follows:
  - Configuration Server
  - Configuration Manager
  - Genesys Administrator
  - Genesys Administrator Server

- Five `Application` objects, as follows:
  - `confserv` object of type Configuration Server.
  - `default` object of type Configuration Manager.
  - `Genesys Administrator` object of type Genesys Administrator.

    > **Tip**
    > Consider changing the name of this application during the first session.

  - `Genesys AdministratorServer` object of type Genesys Administrator Server.
  - Installation Configuration Utility Application object with the name set to `ITCUtility`. This utility supports configuration updates during installation processes for Genesys components. No additional configuration is needed..
- The default Access Group objects: Users, Administrators, and Super Administrators. For more information, refer to Security Considerations.
- Folders for all types of objects managed by the Configuration Layer.

The Configuration Database also contains a number of other predefined objects (for example, Alarm Conditions) that help you set up some Genesys functionality as you deploy other Framework and solution components.

Using Language Packs to Localize the Configuration Database

Starting in release 8.5, Language Packs are available to provide localized content in any environment running Configuration Server 8.5 or later. The Language Packs are available for all supported platforms and databases, in both single-language and multi-language environments.

This section describes how to install these Language Packs and apply them to the Configuration Database. For more information about an LP, refer to its Release Note.

Language Packs

A Language Pack (LP) contains a set of SQL scripts for all supported DBMSs and single-language (default) and multi-language (UTF-8) modes of Configuration Server. The LP is associated with a language ID number given in LP documentation.

The scripts in the LP are called **CfgLocale<*dbms*>.sql**, where **<*dbms*>** indicates the target Configuration Database DBMS. All text in the script is in the language for which the LP is intended.

The LP Installation Package (IP) consists of two folders, **multilang** and **singlelang**. Scripts in the **multilang** folder are SQL text files with UTF-8 encoded national characters. Scripts in the **singlelang** subfolder are SQL text files with national characters encoded according to the ANSI code page as specified in the Windows *National Language Support (NLS) API Reference*.

Compatibility

The compatibility of Management Framework Language Packs is determined by the data schema used by Configuration Database, not the version of Configuration Server. Language Packs are supported by database schema 8.5 and later; in other words, any database schema 8.1 or earlier is not compatible with Language Packs. If you did not update the Configuration Database schema to 8.5 when you migrated to Management Framework 8.5.0, you cannot use Language Packs until you migrate your Configuration Database schema. Any compatibility exceptions are documented in the Release Note for the Language Pack.

Installing Language Packs

Install the LP on a host on which DBMS client software is installed and that can access the target Configuration Database.

> **Tip**
>
> When installing the LP, note this number and be prepared to use it when applying the LP. If required, you can also retrieve this number from the info.xml file in the LP installation folder, as the **LocaleId** attribute of the root tag **<language>**. For example, in the following code sample, the language ID is 1041.
>
> ```
> <?xml version="1.0" encoding="UTF-8"?>
> <Language LocaleId="1041">
>   <Version>"confserver_jpn_8.5.000.00"</Version>
>   <Name>"Japanese (Japan)"</Name>
> </Language>
> ```

Once installed, the LP can be applied to a single-language database by loading data directly into the Configuration Database using DBMS tools. The DBMS tools must be capable of dealing with SQL scripts that contain ANSI encoding (Windows) for the LP language.

> **Warning**
>
> You must stop your Configuration Server (and its backup, if configured) when applying Language Packs. This downtime is unavoidable, but if the Configuration Servers are installed in HA pairs, you might be able to minimize downtime.

To apply the LP to the target Configuration Database, use the steps corresponding to your database.

Single-Language Configuration Database

To apply the LP to a single-language Configuration Database, follow these steps. If your Configuration Servers are configured as an HA pair, consider the steps to minimize downtime during this process.

1. Use the command line or the DBMS user interface, if provided by the DBMS vendor, to connect to the target Configuration Database.
2. Execute the **singlelang/CfgLocale_<*dbms*>** script from the LP installation folder, where **<*dbms*>** is the name of the DBMS used by the Configuration Database.
   When the process is complete, the DBMS will display a success message.
3. Update the language ID, as follows:
   a. Open Configuration Server's configuration file in a text editor.
   b. Locate the **langid** option in the **[<*Configuration Server application name*>]** section.
   c. Change its value to the language ID noted when installing the LP.
   d. Save your changes.

   Repeat this step for the backup Configuration Server, if configured.
4. Test the installation, as follows:
   a. Restart the primary Configuration Server. When it is initialized, restart the backup Configuration Server, if configured.
   b. Try to access the primary Configuration Server from a localized Genesys Administrator.
   c. Confirm that messages are delivered in the desired language.

Multi-Language Configuration Database

> **Important**
> The information in this section is for future use. There are currently no Genesys interface applications that can use a multi-language Configuration Server with installed language packs.

To apply the LP to a multi-language Configuration Database, follow these steps. If your Configuration Servers are configured as an HA pair, consider the steps to minimize downtime during this process.

1. Use the command line or the DBMS user interface, if provided by the DBMS vendor, to connect to the target Configuration Database.
2. Execute the **multilang/CfgLocale_<*dbms*>** script from the LP installation folder, where **<*dbms*>** is the name of the DBMS used by the Configuration Database.
   When the process is complete, the DBMS will display a success message.
3. If there is no backup Configuration Server, restart Configuration Server.

Minimize Downtime

If there is a backup Configuration Server, you can minimize downtime when installing Language Packs by doing the following:

1. Stop the backup Configuration Server.
2. Modify the configuration file of the backup Configuration Server to include the **upgrade-mode=**1 option to enable side-by-side startup without contacting the configured peer server.
3. In Solution Control Server (SCS), set **disable-switchover=**`true` in the **[general]** section so that SCS will not automatically perform the switchover.
4. Disconnect the primary Configuration Server from the database (set **force-offline=**`true` in the Configuration Database section), or shut down all DB Servers that the primary server is configured to use.
5. Apply the LP to the database.
6. Start the backup server and let it initialize in primary mode.
7. Stop the original primary server that is running in read-only mode. Clients will fail over to the backup server currently running in primary mode.
8. When the LP is applied, reverse the previous steps, as follows:
    a. In SCS, set **disable-switchover=**`false`, or remove it altogether, to restore automatic switchovers.
    b. In the configuration file of both Configuration Servers, remove the **upgrade-mode=**1 option to re-establish communication between the two servers at startup.
    c. Restart the backup server normally.

Configuration Server

If you want Configuration Server to operate with the Configuration Database, you must install Configuration Server in *Master* mode. This Configuration Server must be configured through a local configuration file.

> **Important**
>
> - The procedures given in this section are for deploying a primary Configuration Server. To deploy a Configuration Server Proxy, refer to Configuration Server Proxy for relevant installation instructions. To install a backup Configuration Server, refer to Redundant Configuration Servers.
> - Refer to the *Framework External Authentication Reference Manual* for information about Configuration Server's External Authentication feature and for relevant deployment instructions.

Deploying Configuration Server

For more information about the Configuration Server configuration file, see Configuration Server Configuration File. For information about Configuration Server configuration options and their values, refer to the *Framework Configuration Options Reference Manual*.

To deploy Configuration Server, do the following:

1. Install Configuration Server.

Installing on UNIX

To install Configuration Server on UNIX, do the following:

a. On the Management Framework 8.5 product CD, locate and open the installation directory **configuration_layer/configserver/<operating_system>**.

b. Type **install.sh** at the command prompt, and press **Enter**.

c. For the installation type, type 1 to select Configuration Server Master Primary, and press **Enter**.

d. For the external authentication option, type the number corresponding to the type of external authentication that will be used (LDAP, Radius, both, or neither), and press **Enter**.

> **Tip**
> If you select LDAP, be prepared with the URL to access the LDAP Server. For more information about LDAP configuration, see the *Framework External Authentication Reference Manual*.

e. Specify the full path of the destination directory, and press **Enter**.

f. If the target installation directory has files in it, do one of the following:

- Type 1 to back up all the files in the directory, and press **Enter**. Specify the path to where you want the files backed up, and press **Enter**.
- Type 2 to overwrite only the files in this installation package, and press **Enter**. Then type y to confirm your selection, and press **Enter**. Use this option only if the application that is already installed operates properly.
- Type 3 to erase all files in this directory before continuing with the installation, and press **Enter**. Then type y to confirm your selection, and press **Enter**.

The list of file names will appear on the screen as the files are copied to the destination directory.

g. For the product version to install, do one of the following:

- Type 32 to select the 32-bit version, and press **Enter**.
- Type 64 to select the 64-bit version, and press **Enter**.

h. To configure the Configuration Server during, or after, installation, do one of the following:

- Type y to configure Configuration Server during installation (now), and press **Enter**. Go to Step 9 to specify values for the configuration file. For information about the Configuration Server configuration options and their values, refer to the *Framework Configuration Options Reference Manual*.
- Type n to not configure Configuration Server during installation. In this case, you have finished installing Configuration Server-do not continue to the next step in this procedure. Before you can start Configuration Server, however, you must create a configuration file and set the configuration options in it.

i. For the **[confserv]** section:

i. Specify a value for the Configuration Server port, and press **Enter**.

ii. Specify a value for the Configuration Server management port, and press **Enter**.

j. For the **[dbserver]** section:

i. Type the number corresponding to the database engine that this Configuration Server uses (**dbengine**), and press **Enter**.

ii. Specify the name or alias of the DBMS that handles the Configuration Database (**dbserver**), and press **Enter**.

iii. To specify the name of the Configuration Database (dbname), do one of the following:

- If you are using an Oracle database engine (that is, you typed 3 in Step i), press **Enter**. This value is not required for Oracle.
- If you are using any other database engine, specify the name of the Configuration Database, and press **Enter**.

> **Important**
>
> If you are using DB Server to access the Configuration Database, you must also specify values for the **host** and **port** fields. Refer to Framework 8.1 documentation in this case.

iv. Specify the Configuration Database `username`, and press `Enter`.

v. To specify the Configuration Database password, do one of the following:

- Specify the password, and press **Enter**.
- Press **Enter** if there is no password; that is, the password is empty, with no spaces.

When the installation process is finished, a message indicates that installation was successful. The process places Configuration Server in the directory specified during the installation process. The installation script also writes a sample configuration file, **confserv.sample**, in the directory in which Configuration Server is installed.

If you chose to configure the Configuration Server during installation, the sample configuration file, **confserv.sample**, is renamed **confserv.conf**, and the parameters specified in Steps 9 through 11 are written to this file.

If you chose to configure the Configuration Server after installation, you must manually rename the sample file **confserv.conf** and modify the configuration options before you start Configuration Server.

Installing on Windows

> **Warning**
>
> Genesys does not recommend installation of its components via a Microsoft Remote Desktop connection. The installation should be performed locally.

To install Configuration Server on Windows, do the following:

a. On the Management Framework 8.5 product CD, locate and open the installation directory **configuration_layer/configserver/windows**.

b. Locate and double-click **setup.exe** to start the Genesys Installation Wizard.

c. Click **About** on the wizard's **Welcome** page to review the **read_me** file. The file also contains a link to the server's Release Notes file.

d. On the **Welcome** page, click **Next**.

e. On the **Configuration Server Run Mode** page, select **Configuration Server Master Primary**.

f. On the **Configuration Server Parameters** page:

i. Specify the **Server Port** and **Management Port** for Configuration Server.

ii. Click **Next**.

g. On the **Database Engine Option** page, select the database engine that the Configuration Server uses, and click **Next**.

h. On the **DB Server Parameters** page:

i. Specify the **Database Server Name** and **Database Name**.

ii. Specify the **Database User Name** and **Password**.

i. On the **Configuration Server External Authentication** page, select the type of external authentication that the Configuration Server uses, or select **None** if Configuration Server is not using external authentication.

j. On the **Choose Destination Location** page, the wizard displays the destination directory specified in the **Working Directory** property of the server's Application object. If the path configured as **Working Directory** is invalid, the wizard generates a path to **c:\Program Files\GCTI\Configuration Server**. If necessary, do one of the following:

- Click **Browse** to select another destination folder. In this case, the wizard will update the Application object's **Working Directory** in the Configuration Database.
- Click **Default** to reinstate the path specified in **Working Directory**. Click **Next** to proceed.

k. On the **Ready to Install** information page, do one of the following:

- Click **Back** to update any installation information.
- Click **Install** to proceed with the installation.

l. On the **Installation Complete** page, click **Finish**.

As a result of the installation, the wizard adds Application icons to the:

- Windows **Start** menu, under **Programs > Genesys Solutions > Framework**.
- Windows **Add or Remove Programs** window, as a Genesys server.
- Windows **Services** list, as a Genesys service, with Automatic startup type.

2. Configure Configuration Server. If you manually installed Configuration Server on Windows, it was configured automatically during the installation process; you can skip this step. If you manually installed Configuration Server on UNIX and chose not to configure it during the installation process, you must configure it now.

Prerequisites:

- You manually installed Configuration Server on UNIX.
- You chose not to configure Configuration Server during the installation process.
- The Configuration Database has been initialized.

Procedure:

a. From the directory in which Configuration Server is installed, open the sample configuration file (**confserv.sample**) in a text editor.

b. Set the configuration options to work with the Configuration Database and DB Server. Consult the relevant chapters in the *Framework Configuration Options Reference Manual* for option descriptions and values. Refer also to Configuration Server Configuration File for a description of the configuration file.

c. Save the configuration file as **confserv.conf**.

d. If required, configure Configuration Server for multi-language environment support. Add the following options to the **[confserv]** (for Configuration Server) or **[csproxy]** (for Configuration Server Proxy) section of the configuration file:

- Set the **locale** option to the value corresponding to English (US). The database against which a

UTF-8 enabled Configuration Server or Configuration Server Proxy is launched must be initialized using English locale scripts.

- Set the **encoding** option to `utf-8`.
- Set the **multi-languages** option to `true`. You must set this option after initializing the database and before you start Configuration Server against the UTF-8 enabled database.

For more information about these options, refer to the *Framework Configuration Options Reference Manual*.

e. If required, configure Windows Authentication with an MS SQL Server by doing the following:

- Ensure that a Configuration Server process is enabled for Windows Authentication.
- Configure access to the MS SQL Configuration Database for Configuration Server.

Refer to "Windows Authentication with MS SQL Server" in the Microsoft SQL Server Databases section of the *Framework Database Connectivity Guide for details.*

f. Start Configuration Server. For descriptions of command-line parameters specific to Configuration Server, refer to Configuration Server.

> **Tip**
>
> - Use the **-c** command line option to point Configuration Server to a configuration file with the name other than the default name (**confserv.conf** on UNIX or **confserv.cfg** on Windows). For example, **confserv -c <configuration file name>**.
> - If you are starting Configuration Server for the first time, and want to start logging during startup and initialization (referred to as *bootstrap logging*), include the **-log-<log option name> <log-type>** parameter in the startup command. At this point, you can also store these logs in a file separate from the operational logs—also include the **-log-<log-type> <filename>** parameters in the startup command.

Prerequisites:

- Configuration Database is initialized.
- DB Server is installed and running.
- Configuration Server is installed.
- The Configuration Server configuration file is configured. Configuration Server uses this file for startup.

Starting on UNIX

Go to the directory in which Configuration Server is installed and do one of the following:

- To use only the required command-line parameters, type the following command line:

```
sh run.sh
```

- To specify the command line yourself, or to use additional command-line parameters, type the following command line:

```
confserv [<additional parameters and arguments as required>]
```

Starting on Windows

Do one of the following:

- Use the **Start > Programs** menu.
- To use only the required command-line parameters, go to the directory in which Configuration Server is installed, and double-click the **startServer.bat** file.
- To specify the command line yourself, or to use additional command-line parameters, open the MS-DOS window, go to the directory in which Configuration Server is installed, and type the following command line:

```
confserv.exe [<additional parameters and arguments as required>]
```

- Use Windows Services Manager. Refer to Starting and Stopping with Windows Services Manager for more information.

Configuration Server Configuration File

At a minimum, the configuration file contains the Configuration Server, Configuration Database, and Log sections.

The Configuration Server section contains the configuration options that define Configuration Server. The name of the section corresponds to the name of the Configuration Server Application object. For the initial installation of Configuration Server, it is called **[confserv]** by default. You can choose to rename this Configuration Server later. In all other cases, or if you rename the initial Configuration Server, the name of this section will be different. The **server** configuration option in this section specifies the name of the Configuration Database section.

By default, the Configuration Database section does not have a name. The section name must be the same as the value of the **server** configuration option that you specified in the Configuration Server section. The Configuration Database section contains information about the Configuration Database.

The name of the Log section is **[log]**. This section contains configuration information about the logging to be done by Configuration Server.

You can find a sample Configuration Server configuration file in the *Framework Configuration Options Reference Manual*.

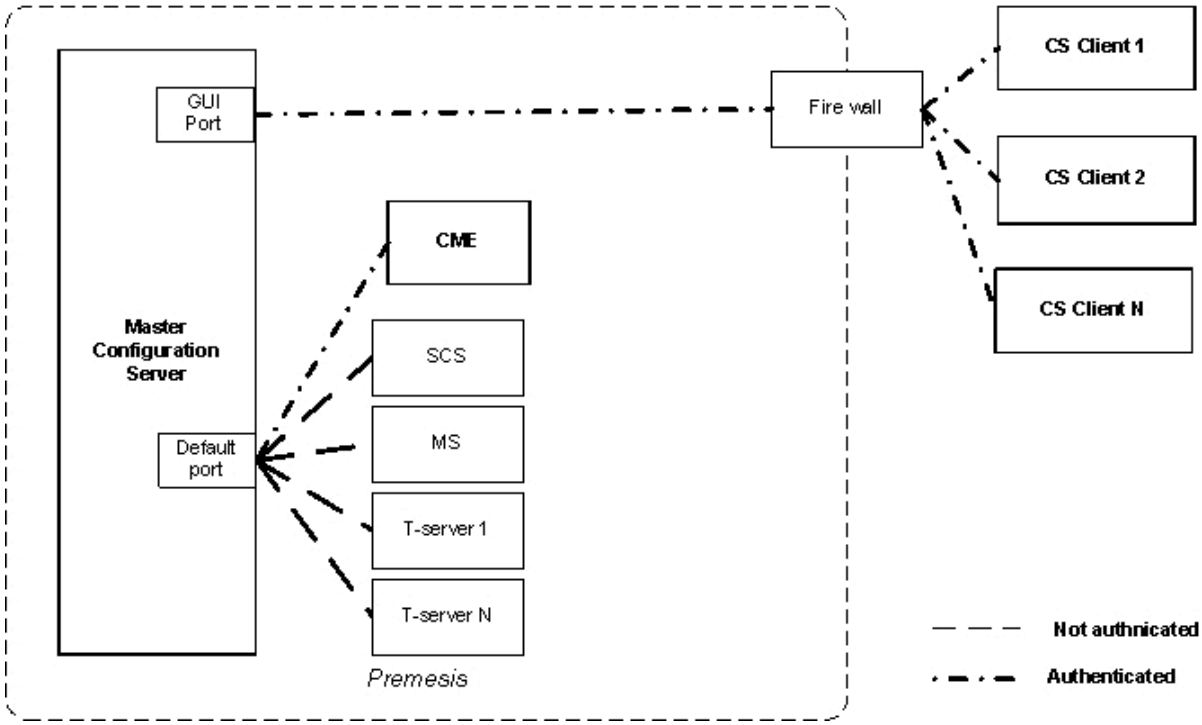
Configuring a Dedicated Port for Client User Interface Applications

> **Warning**

- Genesys strongly recommends that you do not restrict the default port to accept only client UI applications. Because the backup Configuration Server communicates with Configuration Server via the default port, and because many other Genesys Server applications cannot operate properly with being connected to the default port, restricting the default port would disable you from using these additional beneficial components.
- Ports that have been dedicated as **HA sync** (in the **Server Info** section of the port's **Configuration** tab in Genesys Administrator) cannot be provisioned to accept only client UI applications.

Starting in release 8.5.1, you can configure additional ports to which only client UI applications can connect. To configure this port, do the following:

1. Set up a firewall between client UI applications deployed in a less secure area of your network, for whom authorization is required, and applications, including Configuration Server, deployed in a more secured (restricted) area. The firewall directs all "outside" client UI applications to the dedicated port of Configuration Server, where they are authorized. Other "inside" applications continue to use their assigned ports. The following diagram illustrates a dedicated port within the firewall.



2. After you have the firewall in place, configure the port to use as a dedicated port. You can use an existing port (not the default port) or create a new one.
   a. Open the **Server Info** section of the **Configuration** tab of the Configuration Server or Configuration Server Proxy application object.
   b. In the **Listening Ports** section, select an existing port (not the `default` port) or create a new port to be used as the dedicated interface port.

c. In the **Port Info** dialog box, select the **Advanced** Tab.

d. Enter `user=1` in the **Application Parameters** field and click **OK**.

e. Save your changes.

Refer to the *Framework Configuration Options Reference Manual* for more information about the `user` parameter.

Configuring Configuration Server Logging

If you plan to use the centralized logging and auditing functionality of the Management Layer, specify appropriate log options in the Configuration Server configuration file before you start using Configuration Server. Most importantly, enable the network log output (for example, create a new option called **standard** and set its value to `network`). Refer to the *Framework Configuration Options Reference Manual* for more information.

Changing Configuration Server Port Assignments

When you install Configuration Server, you specify values for the listening and management ports in the configuration file. You can change these values at any time.

Changing these port assignments depends on the type of port. To change the value of the management port, you must update the configuration file with the revised information, and restart Configuration Server.

Changing the value of the listening port is more complex. As described in Multiple Ports on Configuration Server, Configuration Server reads its listening port assignment from the configuration file once, at initial startup. For subsequent startups, it reads the port value from the Configuration Database. Therefore, you must change the value in the Configuration Database by modifying the **Port** property of the Configuration Server Application object, as follows:

1. In Genesys Administrator, select the **Provisioning** tab, go to **Environment > Applications**, and double-click the Configuration Server Application object for which you want to change the listening port.
2. On the **Configuration** tab, open the **Server Info** section.
3. In the list of **Listening Ports**, do one of the following:
   - Click the port number that you want to change, enter the new port number, and either click outside of the edit box or press **Enter**.
   - Highlight the port that you want to change and click **Edit**. On the **General** tab of the **Port Info** dialog box, enter the new port number in the **Port** text box. Then click **OK**.
4. Click **Save** or **Save & Close** in the toolbar to save your configuration changes.

Encrypting the Configuration Database Password

You can use Configuration Server to encrypt your password for accessing the Configuration Database so that it does not appear in plain text in Configuration Server logs. This improves the security of your configuration data.

You can encrypt the password at any time, either during installation, or later. However, keep in mind that Configuration Server must be stopped during the encryption process.

In release 8.5.0 and earlier, the password was encrypted using an asymmetric encryption algorithm TEA with a hardcoded encryption/decryption key. For instructions on encrypting the Configuration Database password in release 8.5.0 or earlier, refer to the *Genesys Security Deployment Guide*.

Starting in release 8.5.1, the Configuration Server configuration file optionally supports an asymmetric encryption algorithm using separate encryption and decryption (private) keys that are not hardcoded. In this case, the keys are generated by Configuration Server and stored in separate files. The password is encoded using the key in the encryption file. Upon subsequent restarts of Configuration Server, it uses the key in the decryption file to decrypt and the password.

To encrypt the Configuration Database password in release 8.5.1 or later, do the following:

1. (Optional) Generate encryption keys for encoding and decoding passwords by starting Configuration Server from the command-line with the parameter **-keys [<encryption file name> <decryption file name>]** (and the **-s** and **-c** parameters, if required). The optional subparameters indicate the name and path of the files containing the encryption and decryption keys, respectively. Configuration Server generates the keys and stores them in the file specified by the `-keys` parameter. If no files are specified, the keys are stored in the default files **enc.pem** and **dec.pem**. Configuration Server terminates when this step is complete, generating an error message if key generation was unsuccessful.
2. Encrypt the database password by starting Configuration Server from the command-line with the following parameters:
   - **-p <name of Configuration Database section><password value>** (and **-s** and **-c**, if required)
   - **-keys [<encryption file name> <decryption file name>]**—Required only if you are using asymmetric encryption.

   If the **-keys** parameter is not specified, the hardcoded key is used to encrypt the password, as in previous releases. If **-keys** is specified, Configuration Server creates the encryption and decryption keys and the password is encrypted using the key in the specified encryption file, or **enc.pem** if no file is specified. In both cases, Configuration Server updates its configuration file with the encrypted password in the section specified by the **-p** parameter, and sets the **encryption** option to `true` in the Configuration Server section.

   > **Important**
   > The user launching Configuration Server must have Write permission to the configuration file.

   If the **-keys** parameter is specified, Configuration Server also sets the decryption-key option in the Configuration Server section to point to the specified decryption file, or **dec.pem** if no file is specified. The presence of this option and **encryption=**`true` indicates that the password was

encrypted using the asymmetric algorithm.

3. Start Configuration Server normally. If **encryption=**`true`, it will attempt to decrypt the database password stored in its configuration file using the hard-coded key (if **decryption-key** is not configured or set to an empty string), or use the decryption key stored in the file specified by **decryption-key**.

For Configuration Servers that are part of an HA pair, update each server's configuration file individually. However, they can use the same pair of encryption and decryption keys by specifying the same key file names when configuring encryption for the second server as the first server.

This enhanced encryption capability does not apply to Configuration Server Proxy.

Configuration Server might accept encryption and decryption keys generated by tools or components other than Configuration Server. These keys and their format must be compatible with the cryptography engine used by Configuration, specified in the following table:

Type	RSA asymmetric
Engine	OpenSSL 1.0.1
Key Length	2048 (when keys are generated internally by Configuration Server) **Note:** For Configuration Server 8.5.101.28 and earlier versions, the key length must be 1024.
Embedded Key Generation	default open openssl modulus and RSA_F4 exponent parameters
File Usage	PEM files, that store the RSA key used for encryption or decryption. Both can be produced by Configuration Server.

Install Genesys Administrator

Genesys Administrator is a web-based GUI application that replaces Configuration Manager and Solution Control Interface. You must install it before you can deploy the rest of your system.

Refer to the detailed instructions in the *Framework Genesys Administrator Deployment Guide* to deploy and start Genesys Administrator for your system.

For instructions on deploying Genesys Administrator on Microsoft Server 2012 or 2016, refer to *Genesys Administrator 8.1 Deployment Procedure*.

> **Important**
>
> The latest GA release is available only with MFWK 8.1.3 version. For information on supported GA versions, see the Framework page in the *Genesys Supported Operating Environment Reference Guide*.

Create Hosts

Host objects represent computers in a network. Before you set up the Management Layer, you must configure a Host object for each computer on the data network on which you are going to run the Genesys daemon processes (usually server applications).

To create a Host, do the following:

1. In Genesys Administrator, go to **Provisioning > Environment > Hosts**.
2. Click **New**.
3. On the **Configuration** tab:
   a. Enter the name of the host, exactly as it is defined in the system configuration.

   > **Warning**
   > The host **Name** must be exactly the same as the host name defined in the system configuration.

   b. Enter the IP address of the host.
   c. Select the type of operating system from the **OS Type** drop-down list, and enter its version, if known. For Windows hosts, Genesys recommends that you select **Windows** as a generic type for all hosts running the Windows 2016 and later operating system. For Linux hosts, Genesys recommends that you select **Linux** as a generic type for all hosts running a Linux operating system other than Red Hat Enterprise Linux.
   Note: To use the **Linux** OS type, you must use Configuration Database 8.5 and upgrade to latest locale definitions as described in Updating Configuration Definitions.
   d. Enter the Local Control Agent (LCA) port number or accept the default (4999), to enable the Management Layer to control applications running on this host. This is also the port used by other applications installed on this host to connect to LCA. Refer to Notes on Configuring the LCA Port for additional information about configuring the LCA port value.
4. To customize the Advanced Disconnect Detection Protocol (ADDP) functionality that will be enabled between Solution Control Server (SCS) and LCA, on the Options tab:
   a. In the **View** drop-down list, select `Advanced View (Annex)`.
   b. To specify the ADDP timeout between LCA and SCS, create a section called **[addp]**, add the option **addp-timeout** in this section, and specify a value.
   c. To enable sending LCA polling messages to SCS, in the section **[addp]**, add the option **addp-remote-timeout**, and specify a value.

   Refer to Configuring ADDP Between Solution Control Server and Local Control Agent for more information. For detailed information about the configuration options themselves, refer to the *Framework Configuration Options Reference Manual*.
5. Click **Save and Close**.

For more information about setting configuration options using Genesys Administrator, refer to the

Help file for your user interface. For more information about specific configuration options, refer to the *Framework Configuration Options Reference Manual*.

Enabling Management Layer to Control Configuration Layer

To enable the Management Layer to control (start, stop, and monitor) Configuration Server, you must modify the Configuration Server application to communicate with the Local Control Agent (LCA), as follows:

Prerequisites

- Configuration Server is installed and running, and its Application object is created.
- A Host object exists for the computer on which this Configuration Server will be running. See Creating Hosts.
- You are logged in to Genesys Administrator.

Procedure

1. Go to **Provisioning > Environment > Applications**, and click the Configuration Server Application object (named `confserv`) to open its properties.
2. On the **Configuration** tab, open the **Server Info** section.
3. Select the host on which this Configuration Server runs.
4. Define the **Working Directory** and **Command Line** properties for the primary Configuration Server, if not already defined.
5. Click **Save and Close** to save the changes.

Deploying Management Layer

The Management Layer controls the startup and status of solutions, logging of maintenance events, generation and processing of alarms, and management of application failures.

Deployment Summary

Deploy Management Layer in this order:

1. Local Control Agent (LCA)—LCA must be installed to enable the Management Layer's solution-control and fault-management capabilities. You must install one LCA on each host running a Genesys server application.

   > **Important**
   > An application started by LCA inherits the environment variables from LCA. Therefore, when an application requires that particular environment variables be set, the same environment variables must be set for the account that runs LCA.

2. Database Access Point (DAP) for the Log Database
3. Message Server—You must configure a connection to Message Server for each Genesys server application to enable the Management Layer's centralized-logging and alarm-signaling capabilities.
4. Initialize the Log Database
5. Solution Control Server
6. SNMP Master Agent (optional)

Starting in 8.5, Genesys Administrator replaces Configuration Manager and Solution Control Interface (SCI) as the preferred interface for Management Framework. If you still want to use SCI, deploy it using the deployment instructions in the *Framework 8.1 Deployment Guide*.

Remote Deployment

You can deploy the Management Layer servers (Message Server, Solution Control Server, and Genesys SNMP Master Agent) to any host on your network using Genesys Administrator Extension. Refer to Genesys Administrator Extension documentation for more information and instructions.

Next Steps

After you have successfully installed and configured the Management Layer components, consider whether you would like to configure the following:

- Force logged-in users to log in again after a period of inactivity. See Forced Re-Login for Inactivity.
- Redundant Message Servers, Solution Control Servers, or SNMP Master Agents. See Redundant Configurations.
- Distributed Solution Control Servers.

Continuing the Installation of Your System

Once the Management Layer is set up, you can then deploy the rest of the Framework components, the contact center environment, and other Genesys Voice and Data applications.

Local Control Agent (LCA)

To enable the Management Layer to control the startup and status of applications and solutions, and manage application failures, you must install an instance of Local Control Agent on every computer that is to run either Genesys server applications or third-party server applications you want to control with Management Layer.

Notes on Configuring the LCA Port

- The LCA port must be set to a value of 2000 or greater. When the LCA port is specified within the range of 1-1999, LCA starts on port number 4999 (default value).
- If the LCA port value is changed in the Host configuration while Solution Control Server (SCS) is connected to LCA, SCS does not disconnect from and reconnect to LCA; instead, the new LCA port value takes effect after LCA restarts.
- If you change the LCA port value for the LCA installed as a Windows Service, you must also change the LCA port number in the LCA startup parameters in the Registry Editor. The LCA Registry Key is located at:
  `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\<lca_service_name>\ImagePath`
  The value must have the following format:

  ```
  <full path>\lca.exe <LCA port number> -service <lca_service_name>
  ```

  Change the LCA port number to the current value.

Installing Local Control Agent

> **Important**
> All running LCA processes must be stopped before installing another LCA.

Installing LCA on UNIX

1. Stop all LCA processes that are running. If there are any LCA processes that are running when you begin the installation, the installation process will stop, and not restart until you have stopped those processes (see Step 4).
2. On the Management Framework product CD in the appropriate **management_layer/lca/<operating_system>** directory, locate a shell script called **install.sh**.
3. Type the file name at the command prompt, and press **Enter**.

4. Type **Enter**. This action will have one of the two following results.
   - If there are any LCA processes still running, you must exit from the installation and have to stop these processes before you can restart it.
   - Otherwise, you continue with the installation.
5. To specify the hostname for this LCA, do one of the following:
   - Type the name of the host, and press **Enter**.
   - Press **Enter** to select the current host.
6. Enter the Configuration Server host name, and press **Enter**.
7. Enter the Configuration Server network port, and press **Enter**.
8. Enter the Configuration Server user name, and press **Enter**.
9. Enter the Configuration Server password, and press **Enter**.
10. To specify the destination directory, do one of the following:
    - Press **Enter** to accept the default.
    - Enter the full path of the directory, and press **Enter**.
11. If the target installation directory has files in it, do one of the following:
    - Type 1 to back up all the files in the directory, and press **Enter**. Then specify the path to which you want the files backed up, and press **Enter**.
    - Type 2 to overwrite only the files in this installation package, and press **Enter**. Then type y to confirm your selection, and press **Enter**. Use this option only if the application already installed operates properly.
    - Type 3 to erase all files in this directory before continuing with the installation, and press **Enter**. Then type y to confirm your selection, and press **Enter**.

    The list of file names will appear on the screen as the files are copied to the destination directory.
12. For the product version to install, do one of the following:
    - Type 32 to select the 32-bit version, and press **Enter**.
    - Type 64 to select the 64-bit version, and press **Enter**.
13. If you are authorized to modify startup (RC) files, you are prompted to add LCA to the startup files. Do one of the following:
    - Press **Enter** to add LCA to the startup files.
    - Type n to leave LCA out of the startup files, and press **Enter**.

> **Important**
>
> On UNIX systems, LCA 8.1.0 and earlier is installed with the autostart capability created automatically for run level 3. If you are using another run level, you must modify your operating system startup scripts by adding the startup of LCA.

Installing LCA on Windows

1. Stop all LCA processes that are running.
2. On the Management Framework product CD in the appropriate **management_layer\lca\windows directory**, locate and double-click **setup.exe** to start the Genesys Installation Wizard.
3. Use the **About** button on the wizard's **Welcome** page to view the **read_me** file. The file also contains a link to the server's Release Notes file.
4. Click **Next** to start the installation.
5. On the **Connection Parameters to the Genesys Configuration Server** page, specify the host name, port, user name, and password for Configuration Server, and then click **Next**.
6. On the **Choose Destination Location** page, the wizard displays the default folder **C:\Program Files\GCTI\Local Control Agent**.
   If necessary, do one of the following:
   - Click **Browse** to select another destination folder.
   - Click **Default** to reinstate the default folder, **C:\Program Files\GCTI\Local Control Agent**.
7. On the **Ready to Install** page, do one of the following:
   - Click **Back** to update any installation information.
   - Click **Install** to proceed with the installation.
8. On the **Installation Complete** page, click **Finish**.

As a result of the installation, the wizard adds Application icons to the:

- Windows **Start** menu, under **Programs > Genesys Solutions > Management Layer**.
- Windows **Add or Remove Programs** window, as a Genesys server.
- Windows **Services** list, as a Genesys service, with **Automatic** startup type.

> **Important**
>
> Because the Management Layer functionality requires LCA to be always running while its host computer is up, LCA is installed as a Windows Service with the `autostart` capability.

LCA Log Options

Local Control Agent supports the unified set of log options (common log options) to allow precise configuration of log output. For a complete list of unified log options and their descriptions, see the "Common Log Options" section of the *Framework Configuration Options Reference Manual*.

If you do not specify any log options for LCA, the default values apply. To specify log options for LCA, modify the **lca.cfg** configuration file that was created during LCA deployment, and is located in the

same directory as the LCA executable. The LCA configuration file has the following format:

```
[log]
<log option name> = <log option value>
<log option name> = <log option value>
...
```

A sample LCA configuration file is available in the *Framework Configuration Options Reference Manual*.

Configuring ADDP Between Solution Control Server and Local Control Agent

Advanced Disconnection Detection Protocol (ADDP) is enabled automatically between Solution Control Server (SCS) and LCA. By default, SCS generates polling messages to LCA. If SCS does not receive messages from LCA within this interval, SCS sends a polling message. A lack of response to the polling message from LCA within the same time period is interpreted as a loss of connection.

If you want to change the ADDP timeout between SCS and LCA, configure the **addp-timeout** option. If you also want to enable LCA polling messages to SCS, configure the **addp-remote-timeout** option. Both of these options are set in the annex of the Host object configured for the computer on which LCA runs. For detailed instructions on specifying these options, refer to the *Framework Configuration Options Reference Manual*.

To avoid false disconnect states that might occur because of delays in the data network, Genesys recommends setting the ADDP timeouts to values equal to or greater than ten seconds.

For more information about ADDP, see Advanced Disconnection Detection Protocol.

Database Access Points

To cover the variety of ways the applications in the Genesys installation can be interfaced with databases, the Configuration Layer uses the concept of a Database Access Point.

A Database Access Point (DAP) is a configuration object of the Application type that describes both the parameters required for communication with a particular database, such as JDBC parameters, and the parameters of the database itself. The DAP application you configure for the Management Layer enables Message Server to connect to the Log Database directly. If, according to your configuration, a database can be accessed by multiple applications simultaneously, register one DAP for each possible connection.

For detailed instructions for configuring the Log DAP, and more information about how Management Framework servers connect to databases, refer to the *Framework Database Connectivity Reference Guide*.

Message Server

To deploy Message Server, do the following:

1. Configure a Message Server Application object.

   ### Prerequisites

   - A Database Access Point for the Log Database is configured.
   - You are logged in to Genesys Administrator.

   ### Procedure

   a. In Genesys Administrator, go to **Provisioning > Environment > Applications**, and select **New** in the toolbar. This opens a **Browse** dialog box that lists the available application templates. If a Message Server template file is not listed, do one of the following:
      - Import the **Message_Server_*current-version*.apd** file from the Management Framework 8.1 product CD.
      - Create a new template and repeat this step.

   b. In the **Browse** dialog box, select the Message Server template file. The **Configuration** tab for the new Message Server Application object appears in the Details panel.

   c. In the **General** section:
      - Enter a descriptive name in the **Name** field; for example, `MsgServer`.
      - Add a connection to the Log Database DAP. In the **Connections** field:
        i. Click **Add**.
        ii. Enter the properties of the connection in the **Connection Info** dialog box.
        iii. Click **OK**.

   d. In the **Server Info** section:
      a. In the **Host** field, click the magnifying glass icon to select the host on which this Message Server is running.
      b. For each listening port that an application must use to connect to Message Server:
         1. In the **Listening Ports** field, click **Add**.
         2. Enter the port properties in the **Port Info** dialog box.
         3. Click **OK**.
      c. For the **Working Directory**, **Command Line**, and **Command Line Arguments** fields, do one of the following:
         - Enter the appropriate information in the three text boxes. For information about command-line parameters, see Message Server.
         - Type a period (`.`) in the **Working Directory** and **Command Line** text boxes, and leave the **Command Line Arguments** text box blank. The information will be filled in automatically

when you install Message Server, but only if the Installation Package can connect to Configuration Server.

e. If you want Message Server to direct log events to the Log Database, on the **Options** tab:
   a. In the drop-down list in the top-right corner, select `Options` if not already selected.
   b. In the **[messages]** section, change the value of the **db_storage** option to `true`.
f. Click **Save** or **Apply** in the toolbar to save the new object. The new object will appear in the list of Applications.

2. Install Message Server.

On UNIX

> **Warning**
> During installation on UNIX, all files are copied into the directory you specify. The install process does not create any subdirectories within this directory, so do not install different products into the same directory.

Prerequisite

- A Message Server Application object exists.

Procedure

a. On the Management Framework 8.1 product CD in the appropriate **management_layer/message_server/*operating_system*** directory, locate a shell script called **install.sh**.
b. Type the file name at the command prompt, and press **Enter**.
c. To specify the host name for this Message Server, do one of the following:
   - Type the name of the host, and press **Enter**.
   - Press **Enter** to select the current host.
d. Enter the Configuration Server host name, and press **Enter**.
e. Enter the Configuration Server network port, and press **Enter**.
f. Enter the Configuration Server user name, and press **Enter**.
g. Enter the Configuration Server password, and press **Enter**.
h. The installation displays the list of Application objects of the specified type configured on this Host object. Type the number corresponding to the Message Server Application object you configured above, and press **Enter**.
i. To specify the destination directory, do one of the following:
   - Press **Enter** to accept the default.
   - Enter the full path of the directory, and press **Enter**.

j. If the target installation directory has files in it, do one of the following:
   - Type 1 to back up all the files in the directory, and press **Enter**. Specify the path to which you want the files backed up, and press **Enter**.
   - Type 2 to overwrite only the files in this installation package, and press **Enter**. Then type y to confirm your selection, and press **Enter**.
     Use this option only if the application already installed operates properly.
   - Type 3 to erase all files in this directory before continuing with the installation, and press **Enter**. Then type y to confirm your selection, and press **Enter**.

     The list of file names will appear on the screen as the files are copied to the destination directory.

k. For the product version to install, do one of the following:
   - Type 32 to select the 32-bit version, and press **Enter**.
   - Type 64 to select the 64-bit version, and press **Enter**.

On Windows

> ### Warning
> Genesys does not recommend installation of its components via a Microsoft Remote Desktop connection. The installation should be performed locally.

Prerequisite

- A Message Server Application object exists.

Procedure

a. On the Management Framework 8.1 product CD in the appropriate **management_layer\message_server\windows** directory, locate and double-click **setup.exe** to start the Genesys Installation Wizard.

b. Click **About** on the wizard's **Welcome** page to review the read_me file. The file also contains a link to the server's Release Notes file.

c. Click **Next** to start the installation.

d. On the **Connection Parameters to the Genesys Configuration Server** page, specify the host name, port, user name, and password of Configuration Server, and then click **Next**.

e. On the **Select Application** page, select the name of the Message Server Application object that you configured above, and then click **Next**.

f. On the **Choose Destination Location** page, the wizard displays the destination directory if specified in the **Working Directory** property of the server's Application object during configuration. If you entered a period (.) in this field when configuring the object, or if the path specified in this property is invalid, the wizard generates a path to the destination directory in the **c:\Program Files\GCTI*Product Name*** format.
   If necessary, do one of the following:

- Click **Browse** to select another destination folder. In this case, the wizard will update the Application object's **Working Directory** property in the Configuration Database.
- Click **Default** to reinstate the path specified in the Working Directory property.

Click **Next** to proceed.

g. On the **Ready to Install** page, do one of the following:
   - Click **Back** to update any installation information.
   - Click **Install** to proceed with the installation.

h. On the **Installation Complete** page, click **Finish**.

As a result of the installation, the wizard adds Application icons to the:

- Windows **Start** menu, under **Programs > Genesys Solutions > Management Layer**.
- Windows **Add or Remove Programs** window, as a Genesys server.
- Windows **Services** list, as a Genesys service, with **Automatic** startup type.

3. If required, configure Windows Authentication with an MS SQL Server by doing the following:
   a. Ensure that a Message Server process is enabled for Windows Authentication.
   b. Configure access to the MS SQL Log Database for Message Server. In the **DB Info** section of **Configuration** tab for the Database Access Point of the MS SQL Log Database, configure one of the following as appropriate:
      - If you are using a Trusted user, enter `trusted` in the **User Name** field.
      - If you are using a Data Source Name (DSN), enter the name of the DBMS and Log database in the **DBMS Name** and **Database Name** fields, respectively.

Refer to "Windows Authentication with MS SQL Server" in the Microsoft SQL Server Databases section of the *Framework Database Connectivity Guide for details.*

Initialize the Centralized Log Database

> **Important**
> If you are setting up the Log Database for use in a multi-language environment, refer to the *Framework Database Connectivity Reference Guide* for additional information.

Prerequisites

- A DBMS is installed, and a blank database has been created.
- Message Server is installed and running.

Procedure

1. In your DBMS interface, go to the directory in which Message Server is installed and open the **scripts** folder.
2. Open the folder that matches your database type.
3. Load and execute the scripts that correspond to your DBMS, as provided in the following table.

> **Important**
> Starting in release 8.5.1, the initialization scripts **init_<DBMS>.sql** and **init_multilang_<DBMS>.sql** no longer drop existing tables and procedures as part of initializing the Log Database. If you have to re-run an initialization script against an existing Log Database, make sure that you first run the script **drop_tables_<DBMS>.sql**.

DBMS	Drop Tables Script (Optional)	Initialization Script	Multi-language Initialization Script [a]
DB2	drop_tables_db2.sql	init_db2.sql	init_multilang_db2.sql
Microsoft SQL	drop_tables_mssql.sql	init_mssql.sql	init_multilang_mssql.sql [b]
Oracle	drop_tables_oracle.sql	init_oracle.sql	init_multilang_oracle.sql
PostgreSQL	drop_tables_postgre.sql	init_postgre.sql	Not required

Notes:
a. Use the multi-language scripts if you are setting up your Centralized Log system in multi-language mode. Make sure that the Log Database is created with settings that allow for use in multi-language environments. Refer to the *Framework Database Connectivity Reference Guide* for recommented settings for your database.
b. A multi-language MS SQL database uses UCS-2 encoding instead of UTF-8 encoding. You must set **utf8-ucs2=**`true` in the **[dbclient]** section in the annex of the corresponding Database Access Point.

DBMS	Drop Tables Script (Optional)	Initialization Script	Multi-language Initialization Script [a]
Refer to the *Framework Configuration Options Reference Manual* for more information about this option.			

Solution Control Server

Deploying Solution Control Server

To deploy Solution Control Server, do the following:

1. Configure a Solution Control Application object.

   ### Prerequisite

   - You are logged in to Genesys Administrator.

   ### Procedure

   a. In Genesys Administrator, go to **Provisioning > Environment > Applications**, and select **New** in the toolbar. This opens a **Browse** dialog box that lists available application templates. If a Solution Control Server template file is not listed, do one of the following:

      - Import the **Solution_Control_Server_*current-version*.apd** file from the Management Framework 8.5 product CD.
      - Create a new template and repeat this step.

   b. In the Browse dialog box, select the Solution Control Server template file. The **Configuration** tab for the new Solution Control Server Application object appears in the Details panel.

   c. In the **General** section:

      1. Enter a descriptive name in the **Name** field-for example, SCS.
      2. If you want to enable alarm signaling, add a connection to the Message Server. In the **Connections** field:

         a. Click **Add**.

         b. Enter the properties of the connection in the **Connection Info** dialog box.

         c. Click **OK**.

   d. In the **Server Info** section:

      1. In the **Host** field, click the magnifying glass icon to select the Host object on which this Solution Control Server is running.
      2. For each listening port that an application must use to connect to Solution Control Server:

         a. In the **Listening Ports** field, click **Add**.

         b. Enter the port properties in the **Port Info** dialog box.

         c. Click **OK**.

      3. For the **Working Directory**, **Command Line**, and **Command Line Arguments** fields, do one of the following:

         - Enter the appropriate information in the three text boxes. For information about command-

line parameters, see Solution Control Server.

- Type a period (.) in the **Working Directory** and **Command Line** text boxes, and leave the **Command Line Arguments** text box blank. The information will be filled in automatically when you install Solution Control Server, but only if the Installation Package can connect to Configuration Server.

e. Click **Save** or **Apply** in the toolbar to save the new object. The new object will appear in the list of Applications.

2. Install Solution Control Server.

On UNIX

> **Warning**
> During installation on UNIX, all files are copied into the directory you specify. The install process does not create any subdirectories within this directory, so do not install different products into the same directory.

Prerequisites

- A Solution Control Server Application object exists.

Procedure

a. On the Management Framework 8.1 product CD in the appropriate **management_layer/solution_control_server/*operating_system*** directory, locate a shell script called **install.sh**.

b. Type the file name at the command prompt, and press **Enter**.

c. When prompted to install only the utilities, type n to install SCS and its utilities, and press **Enter**.

d. To specify the host name for this SCS, do one of the following:

- Type the name of the host, and press **Enter**.
- Press **Enter** to select the current host.

e. Enter the Configuration Server host name, and press **Enter**.

f. Enter the Configuration Server network port, and press **Enter**.

g. Enter the Configuration Server user name, and press **Enter**.

h. Enter the Configuration Server password, and press **Enter**.

i. The installation displays the list of Application objects of the specified type configured on this Host object. Type the number corresponding to the SCS Application object you configured above, and press **Enter**.

j. To specify the destination directory, do one of the following:

- Press **Enter** to accept the default.
- Enter the full path of the directory, and press **Enter**.

k. If the target installation directory has files in it, do one of the following:
   - Type 1 to back up all the files in the directory, and press **Enter**. Specify the path to which you want the files backed up, and press **Enter**.
   - Type 2 to overwrite only the files in this installation package, and press **Enter**. Then type y to confirm your selection, and press **Enter**.
     Use this option only if the application already installed operates properly.
   - Type 3 to erase all files in this directory before continuing with the installation, and press **Enter**. Then type y to confirm your selection, and press **Enter**.

     The list of file names will appear on the screen as the files are copied to the destination directory.

l. For the product version to install, do one of the following:
   - Type 32 to select the 32-bit version, and press **Enter**.
   - Type 64 to select the 64-bit version, and press **Enter**.

m. To decide whether you require a license, refer to the *Genesys Licensing Guide* for information about licensing requirements. Then, do one of the following:
   - Type y if you require a license, and press **Enter**.
   - Type n if you do not require a license, and press **Enter**.

n. If you typed y in the previous step, enter the license location format, press **Enter**, and enter the required parameters.

On Windows

> **Warning**
> Genesys does not recommend installation of its components via a Microsoft Remote Desktop connection. The installation should be performed locally.

Prerequisite

- A Solution Control Server Application object exists.

Procedure

a. On the Management Framework 8.1 product CD in the appropriate **management_layer\solution_control_server\windows** directory, locate and double-click **setup.exe** to start the Genesys Installation Wizard.

b. Click **About** on the wizard's **Welcome** page to review the read_me file. The file also contains a link to the server's Release Notes file.

c. Click **Next** to start the installation.

d. On the **Solution Control Server Installation Mode** page, select **Solution Control Server and Utilities**, and then click **Next**.

e. On the **Connection Parameters to the Genesys Configuration Server** page, specify the host name, port, user name, and password of Configuration Server, and then click **Next**.

f. On the **Select Application** page, select the name of the SCS Application object that you configured above, and then click **Next**.

g. On the **Run-time License Configuration** page, select whether you are using a license. Refer to the *Genesys Licensing Guide* for information about licensing requirements, and then click **Next**.

h. If you selected **Use License** in the previous step, on the **Access to License** page, enter the license access type and required parameters.

i. On the **Choose Destination Location** page, the wizard displays the destination directory if specified in the **Working Directory** property of the server's Application object during configuration. If you entered a period (**.**) in this field when configuring the object, or if the path specified in this property is invalid, the wizard generates a path to the destination directory in the **c:\Program Files\GCTI*Product Name*** format.
If necessary, do one of the following:

- Click **Browse** to select another destination folder. In this case, the wizard will update the Application object's **Working Directory** property in the Configuration Database.
- Click **Default** to reinstate the path specified in the **Working Directory** property.

Click **Next** to proceed.

j. On the **Ready to Install** page, click:

- **Back** to update any installation information.
- **Install** to proceed with the installation.

k. On the **Installation Complete** page, click **Finish**.

As a result of the installation, the wizard adds Application icons to the:

- Windows **Start** menu, under **Programs > Genesys Solutions > Management Layer**.
- Windows **Add or Remove Programs** window, as a Genesys server.
- Windows **Services** list, as a Genesys service, with Automatic startup type.

Solution Control Server Utilities

Solution Control Server includes four utilities:

- **ccgs.pl**-Graceful Call Center T-Servers stop script.
- **gstuckcalls** utility and Stuck Calls detection and deletion scripts-To handle T-Server stuck calls and raise alarms.
- **logmsg** utility-To send log messages on behalf of applications.
- **mlcmd** utility-To send and receive information to and from Solution Control Server.

Staring with SCS 8.5.100.42 release, the following utilities are not installed by default with the SCS installation package: **mlcmd**, **logmsg**, **gstuckcalls**, scripts for **gstuckcalls**, and SNMP utilities and scripts. You can install them separately by following the instructions below.

Installing SCS Utilities On UNIX

1. On the Management Framework 8.1 product CD in the appropriate directory under **management_layer/solution_control_server/*operating_system*** locate a shell script called **install.sh**.
2. Type the file name at the command prompt, and press **Enter**.
3. Type y to specify that you want to install only the utilities, and press **Enter**.
4. Enter the full path of the directory in which you want to install the utilities, for example, **/opt/genesys/scsutil**, and press **Enter**. The installation displays the list of files being extracted and copied to the destination directory.
5. Enter the bit version of the system on which you are installing the utilities, either 32 or 64, and press **Enter**.

Installing SCS Utilities On Windows

1. On the Management Framework 8.1 product CD in the appropriate directory in: **management_layer\solution_control_server\windows** locate and double-click **setup.exe** to start the Genesys Installation Wizard.
2. On the wizard's **Welcome** page, click **Next** to start the installation.
3. On the **Solution Control Server Installation Mode** page, select **Solution Control Server Utilities**, and then click **Next**.
4. On the **Choose Destination Location** page, do one of the following to specify the directory where the utilities will be installed:
   - Click **Next** to accept the default directory.
   - Specify a different path and directory by entering it in the text box or using the **Browse** button. If necessary, use the **Default** button to reinstate the original default. Click **Next** to proceed.
5. On the **Ready to Install** page, click:
   - **Back** to update any installation information.
   - **Install** to proceed with the installation.
6. On the **Installation Complete** page, click **Finish**.

Configuring ADDP between primary and backup Solution Control Server

Advanced Disconnection Detection Protocol (ADDP) is enabled automatically between the primary and backup Solution Control Servers (SCS). By default, the backup SCS generates polling message to the primary SCS. If the backup SCS does not receive a response message from the primary SCS within this interval, the backup SCS sends one more polling message. A lack of response to this polling message from the primary SCS within the same time period is interpreted as a loss of connection. For more information about ADDP, see Advanced Disconnection Detection Protocol.

If you want to change the ADDP timeout between the primary and backup SCS, configure the **addp-timeout** option on the Host object of the primary SCS. The value configured in the primary SCS host is reduced by 2 to take the value of addp time between

SCS primary and SCS backup. If you also want to enable primary SCS polling messages to the backup SCS, configure the **addp-remote-timeout** option. Both of these options are set in the annex of the `Host` object configured for the system on which the primary SCS runs. The default value of **addp-timeout** is 7 and for **addp-remote-timeout**/**addp-trace**, it is `off`. For detailed instructions on these options, refer to the Framework Configuration Options Reference Manual.

To avoid false disconnect states that might occur because of delays in the data network, Genesys recommends setting the ADDP timeouts to values equal to or greater than 10 seconds. Also, set the same options in the SCS backup host as well.

SNMP Master Agent

For the Management Layer to use SNMP, you must configure an SNMP Master Agent Application object in the Configuration Database, and configure a connection to this Application object in Solution Control Server (SCS). You must create an Application object of type Genesys SNMP Master Agent regardless of the type (Genesys or third party) of the actual Master Agent you are deploying. If you deploy a Genesys Master Agent object, use the Application Template from its installation package; otherwise, use the SNMP Master Agent template shipped with SCS to create an object to represent the third-party SNMP Master Agent.

This section provides instructions for deploying a stand-alone SNMP Master Agent application. See Redundant (HA) SNMP Master Agents for information about deploying SNMP Master Agents as an HA pair.

Depending on the solutions for which you want to enable SNMP monitoring, you may need to install several instances of SNMP Master Agent, using the same approach given in this section. Generally, you have to install and configure one instance of SNMP Master Agent on each computer on which you will be using SNMP functionality.

Starting in release 8.5.1, configuration of an SNMP Master Agent on a host is somewhat different, depending on how you are implementing SNMP on that host (Genesys SNMP Master Agent or Net-SNMP), as determined by the **netsnmp-enable** configuration option in the **[snmp]** section. If set to `true`, the Management Layer considers this Application object associated with a third-party Net-SNMP Master Agent; otherwise, it is considered to be associated with Genesys SNMP Master Agent. SCS and LCA support both implementations, and use this option to determine how to work with the particular Master Agent.

You can also re-configure an existing Genesys SNMP Master Agent object to use Net-SNMP, as described in the *Framework Migration Guide*.

To deploy a new SNMP Master Agent, do the following:

> **Important**
>
> Genesys recommends using Net-SNMP for your deployments. Refer to Net-SNMP Master Agent for prerequisites for deploying Net-SNMP.

1. Configure an SNMP Master Agent Application object.

   ### Prerequisites

   - You are logged in to Genesys Administrator.
   - If you are configuring a Net-SNMP Master Agent:
     - Net-SNMP is installed on any host that will be running Net-SNMP Master Agents.
     - Batch or shell scripts for starting and stopping the Net-SNMP application are available.

Procedure

a. In Genesys Administrator, go to **Provisioning > Environment > Applications**, and select **New** in the toolbar. This opens a **Browse** dialog box that lists available application templates. If an SNMP Master Agent template file is not listed, import the **SNMP_Master_Agent_*current-version*.apd** file from the Management Framework 8.5 product CD.

b. In the **Browse** dialog box, select the SNMP Master Agent template file. The **Configuration** tab for the new SNMP Master Agent Application object appears in the Details panel.

c. In the **General** section, enter a descriptive name in the **Name** field, for example `SNMP_MA`.

d. In the **Server Info** section:

1. In the **Host** field, select the Host object on which this SNMP Master Agent is running.
2. For each listening port that an application must use to connect to SNMP Master Agent:
   a. In the **Listening Ports** field, click **Add**.
   b. Enter the port properties in the **Port Info** dialog box.
   c. Click **OK**.
3. For the **Working Directory**, **Command Line**, and **Command Line Arguments** fields, enter the appropriate information, as follows:
   - If this SNMP Master Agent is to implement SNMP using Genesys SNMP Master Agent, do one of the following:
     - Enter the appropriate information in the three text boxes. For information about command-line parameters, see SNMP Master Agent.
     - Type a period (`.`) in the **Working Directory** and **Command Line** text boxes, and leave the **Command Line Arguments** text box blank. The information will be filled in automatically when you install SNMP Master Agent, but only if the Installation Package can connect to Configuration Server.
   - Otherwise, this SNMP Master Agent is to implement SNMP using Net-SNMP. Do the following:
     1. In the **Working Directory** field, enter the path to the Net-SNMP installation folder.
     2. In the **Command Line** field, enter the name of the Net-SNMP executable or binary file.
     3. In the **Command Line Arguments** field,specify the same command-line argurments that are used to start the service/process. This information is used by LCA to identify that Net-SNMP is started and running as a third-party application.

e. (For Net-SNMP Master Agents only) In the **Options** tab, specify the SNMP implementation used by this SNMP Master Agent, as follows:

1. Create the section **[snmp]**, if it does not already exist.
2. In the **[snmp]** section, set the **netsnmp-enable** option to `true`. This enables Net-SNMP in this SNMP Master Agent object. If this option is not set, or set to `false` (the default), SCS and LCA will treat the object as a Genesys SNMP Master Agent.

f. (For Net-SNMP Master Agents only) In the **Annex** tab, specify how to start and stop Net-SNMP, as follows:

1. Create the section **[start_stop]** if it does not already exist.
2. In the **[start_stop]** section, set the **start_command** and **stop_command** options to the path and filename of the batch or shell scripts to start and stop (respectively) the Net-SNMP

application.

> **Important**
>
> SCS reads the configuration settings of the SNMP Master Agent Application object and uses the option values to connect to Net-SNMP. Therefore, you must ensure that the option values configured for the SNMP Master Agent Application object in the Configuration Database match the actual configuration settings in your SNMP Master Agent.

g. Click **Save** or **Apply** in the toolbar to save the new object. The new object will appear in the list of Applications.

2. Install an SNMP Master Agent.

On UNIX

> **Warning**
>
> During installation on UNIX, all files are copied into the directory you specify. The installation process does not create any subdirectories within this directory, so do not install different products into the same directory.

Prerequisite

- An SNMP Master Agent Application object exists.

Procedure

a. On the Management Framework 8.5 product CD in the appropriate **management_layer/snmp_master_agent/*operating_system*** directory, locate a shell script called **install.sh**.

b. Type the file name at the command prompt, and press **Enter**.

c. To specify the host name for this SNMP Master Agent, do one of the following:

- Type the name of the host, and press **Enter**.
- Press **Enter** to select the current host.

d. Enter the Configuration Server host name, and press **Enter**.

e. Enter the Configuration Server network port, and press **Enter**.

f. Enter the Configuration Server user name, and press **Enter**.

g. Enter the Configuration Server password, and press **Enter**.

h. The installation displays the list of Application objects of the specified type configured on this Host object. Type the number corresponding to the SNMP Master Agent Application object you configured above, and press **Enter**.

i. To specify the destination directory, do one of the following:

- Press **Enter** to accept the default.
- Enter the full path of the directory, and press **Enter**.

j. If the target installation directory has files in it, do one of the following:

- Type 1 to back up all the files in the directory, and press **Enter**. Specify the path to which you want the files backed up, and press **Enter**.
- Type 2 to overwrite only the files in this installation package, and press **Enter**. Then type y to confirm your selection, and press **Enter**.
  Use this option only if the application already installed operates properly.
- Type 3 to erase all files in this directory before continuing with the installation, and press **Enter**. Then type y to confirm your selection, and press **Enter**.

  The list of file names will appear on the screen as the files are copied to the destination directory.

k. For the product version to install, do one of the following:

- Type 32 to select the 32-bit version, and press **Enter**.
- Type 64 to select the 64-bit version, and press **Enter**.

On Windows

> **Warning**
> Genesys does not recommend installation of its components via a Microsoft Remote Desktop connection. The installation should be performed locally.

Prerequisite

- An SNMP Master Agent Application object exists.

Procedure

1. On the Management Framework 8.5 product CD in the appropriate **management_layer\snmp_master_agent\windows** directory, locate and double-click **setup.exe** to start the Genesys Installation Wizard.
2. Click **About** on the wizard's **Welcome** page to review the read_me file. The file also contains a link to the server's Release Notes file.
3. Click **Next** to start the installation.
4. On the **Connection Parameters to the Genesys Configuration Server** page, specify the host name, port, user name, and password of Configuration Server, and then click **Next**.
5. On the **Select Application** page, select the name of the SNMP Master Agent Application object that you configured above, and then click **Next**.
6. On the **Choose Destination Location** page, the wizard displays the destination directory if specified in the **Working Directory** property of the server's Application object during

configuration. If you entered a period (.) in this field when configuring the object, or if the path specified in this property is invalid, the wizard generates a path to the destination directory in the **c:\Program Files\GCTI*Product Name*** format.
If necessary, do one of the following:

- Click **Browse** to select another destination folder. In this case, the wizard will update the Application object's **Working Directory** property in the Configuration Database.
- Click **Default** to reinstate the path specified in the **Working Directory** property.

Click **Next** to proceed.

7. On the **Ready to Install** page, click:
   - **Back** to update any installation information.
   - **Install** to proceed with the installation.
8. On the **Installation Complete** page, click **Finish**.

As a result of the installation, the wizard adds Application icons to the:

- Windows **Start** menu, under **Programs > Genesys Solutions > Management Layer**.
- Windows **Add or Remove Programs** window, as a Genesys server.
- Windows **Services** list, as a Genesys service, with Automatic startup type.

3. Add a connection from Solution Control Server to this SNMP Master Agent.
   a. Open the Solution Control Server Application object's **Configuration** tab.
   b. In the **General** section, add the connection to the SNMP Master Agent object just created. In the **Connections** field:
      1. Click **Add** to open the **Connection Info** dialog box.
      2. Enter the properties of the connection.
      3. Click **OK**.
   c. Click **Save** or **Apply** in the toolbar to save the configuration changes for SCS.

For more information about using SNMP Master Agents, refer to the *Framework Management Layer User's Guide*.

Deploying the Rest of Your Framework

Now that you deployed the Configuration Layer and, if required, the Management Layer, you can deploy the rest of the Framework components and the contact center environment.

Recommended Order

Manual deployment of the other Framework components and contact center environment objects involves:

- Configuring the components using Genesys Administrator. Refer to *Genesys Administrator 8.1 Help* for more information.
- Manually installing the configured components.

Before you proceed, make sure that the Configuration Layer and Management Layer components are installed, configured, and running. To help you prepare accurate configuration information and become familiar with the configuration process, review Deployment Planning for help with object-configuration information.

Follow this order for the manual deployment of the other Framework components and contact center environment objects:

1. Media Layer:

- T-Server
- HA Proxy for a specific type of T-Server (if applicable)

> **Important**
>
> Deployment instructions for T-Server and HA Proxy (if applicable) are located in the latest version of the Framework T-Server Deployment Guide for your specific T-Server.

2. Telephony Objects:

- Switching Offices
- Switches
- Agent Logins
- DNs

> **Important**
> Configuration instructions for telephony objects are located in the latest version of the Framework T-Server Deployment Guide for your specific T-Server.

3. Contact Center Objects:

- Access Groups
- Skills
- Persons
- Agent Groups
- Places
- Place Groups

4. Services Layer:

- Stat Server

Genesys recommends registering only those entities that you plan to use in the current configuration. The more data in the Configuration Database, the longer it takes for your system to start up, and the longer it takes to process configuration data. Remember that adding configuration objects to the Genesys Configuration Database does not cause any interruption in the contact center operation.

Depending on how much work it is to configure all applications and objects, consider registering more `User` objects first, with a set of privileges that lets them perform configuration tasks.

> **Warning**
> When configuring redundant applications, do not select the redundancy type Not Specified unless using a switchover mechanism other than that provided by the Management Layer. It is acceptable, however, to leave the redundancy type Not Specified for non-redundant applications (that is, applications that do not have backup servers associated with them).

.

Media Layer

Component (T-Server and HA Proxy, if applicable) configuration and installation for the Media Layer is covered in the latest version of the Framework T-Server Deployment Guide for your specific T-Server. Also covered in that Guide is information about deploying components for redundant and multi-site configurations.

Telephony Objects

The configuration of Configuration Database objects for the telephony equipment used in the contact center is described in the latest version of the Framework T-Server Deployment Guide for your specific T-Server.

Contact Center Objects

Configure Configuration Database objects for the contact center personnel and related entities.

Access Groups

Before deciding what kind of Access Groups you must configure, look at the default Access Groups the Configuration Layer supports and the default access control settings in general.

The default security system may cover all of your needs. If a more complex access control system makes sense for your contact center, Genesys recommends managing it through Access Groups and folders rather than at the level of individuals and objects.

To define an Access Group and its permissions:

1. Identify groups of people that are handling specific activities in the customer interaction network.
2. Create the required Access Group objects.
3. Set Access Group privileges with respect to the object types, using the folders' **Permissions** and **Roles** tabs.

In addition, to simplify the security settings, make sure that permissions are set and changed recursively using the permission propagation mechanism.

Skills

Define agent skills that might be considered as criteria for interaction processing. Skills are configured as independent configuration objects; any Agent can be associated with more than one configured Skill. Therefore, it may be more practical to register Skills before the Agents are configured.

Persons

Persons are more commonly referred to as *Users*, of which there are two major categories: Agents and Non-agents. The latter category includes all Users other than agents that need access to the CTI applications; for example, Center Administrators, Data Network and Telephony Network personnel, designers of interaction-processing algorithms, and Supervisors.

The characteristics of your business environment and your current priorities completely determine the order of registering Persons. Most often, you will want to first configure a few registered Non-agents with a high level of access to help you set up the Configuration Database.

Each user is assigned a unique Username and Password to log in to Genesys software. A valid Username can consist of any characters except \ (backslash) or ? (question mark). For example, a common Username is the User's corporate email address, such as `<first name>.<last name>@<company name>.com`. Refer to the "User Passwords" section of the *Genesys Security Deployment Guide* for detailed information about user passwords.

Assigning Agent Logins and Skills Registering Agents

> **Important**
>
> You create Agent Logins when you are configuring the **Switch** object. Refer to the latest version of the Framework T-Server Deployment Guide for your specific T-Server for instructions.

If a few agents have a certain skill of the same level, consider using a wizard that adds the skill to multiple User objects after you create them. To launch the wizard, select two or more User objects that have the **Is Agent** check box selected, right-click, and select **Manage Skills**. Refer to *Genesys Administrator 8.1 Help* for more information.

Remember that the Configuration Layer requires that you assign a unique user name to each User, including agents. Consider using employee IDs configured in User objects as default user names and passwords.

New Users by default are not automatically assigned to any access group, by default. They must be assigned to one or more Access Groups explicitly. If you want new users to be added automatically to predefined Access Groups, you must manually disable this feature using the configuration option **no-default-access**. Refer to the section "No Default Access for New Users" in the *Genesys Security Deployment Guide* for more information about this feature, and how to use or disable it.

Some GUI applications also use Ranks to determine what functionality is made available to the User currently logged in. Unless Agents are required to use rank-dependent applications in their work, you do not have to assign any specific Ranks to them.

Ranks and access privileges are more important when registering non-agents. When registering non-agents, consider the role they have in the customer interaction business. Do these Users need to monitor agents' performance? Will they need to configure the telephony resources? Are they going to design routing strategies? Having answers to these questions makes it easier to correctly set up the access privileges with respect to configuration objects, and Ranks with respect to different applications objects.

Remember that Ranks with respect to applications are not the same as access privileges with respect to the configuration objects. You must explicitly define Ranks. Access privileges are assigned by default, according to whether the User is an agent or not.

Genesys does not recommend changing the default access-control setting unless absolutely necessary. Remember, the more complex the security system implemented, the more difficult it becomes to administer the database, and the more it affects the performance of the Configuration Layer software.

Agent Groups

Agent Groups are an indispensable element of almost every contact center. Remember that you can assign an agent to more than one group at a time. If you create agent groups based on Skills, use the **Find** command or the **Dependency** tab of a Skill to quickly identify all the agents that have the Skill in question.

Places

If you use Genesys CTI applications to distribute calls to individual agents or agent groups that are not limited by the switch ACD configuration, set up Places and assign individual DNs to them. Because a typical Place consists of more than one DN, prepare the actual layout of the numbering plan to correctly configure the Places, and assign DNs to them.

Place Groups

Define Place Groups and assign individual Places to them only if they will be used for distributing calls to groups of Places and, therefore, you will need to collect availability information and real-time statistics for such groups.

Services Layer

Genesys recommends that you configure and install components of the Services Layer when you deploy the solution they will serve.

Stat Server

The configuration and installation procedures for Stat Server are described in the documentation for Stat Server.

Redundant Configurations

You can increase the availability of your Genesys solutions by deploying redundant pairs of primary and backup servers of the same type, controlled by the Management Layer. You must have special licenses to use these configurations.

All Management Framework servers support the *warm standby* redundancy type, meaning that a backup server application remains initialized and ready to take over the operations of the primary server. Redundancy types are described in the *Genesys Security Deployment Guide*.

Configuration Layer and Management Layer also support switchovers between redundant client applications, regardless of the redundancy type specified by those applications.

Important

- The instructions in this section assume that the primary server is already installed and operating. This section provides only instructions for installing the backup server and configuring the primary and backup servers to operate as a redundant pair.
- When configuring the backup component in a redundant pair, use the same account as for the primary component. Two applications with different accounts cannot be linked (configured) as a redundant high availability (HA) pair.
- If you need to make changes in the configuration of one or both servers in the HA pair, you must unlink the two servers before any changes are made. They can then be linked together and restarted.

Redundant (HA) Configuration Servers

This section describes how to deploy redundant Configuration Servers.

Redundancy

Redundant Configuration Servers support only the Warm Standby redundancy type.

Both the primary and backup Configuration Servers operate with the same Configuration Database. The backup Configuration Server does not accept client connections or make changes to the data until its role is switched to primary. When the backup Configuration Server starts, it establishes a connection to the primary Configuration Server. During the operation, the primary Configuration Server sends notifications about all changes made in the Configuration Database to the backup Configuration Server.

If there are any Configuration Server Proxies connected to the primary Configuration Server when it fails, those Proxy servers connect to the backup Configuration Server when it assumes the primary role.

Deploying Redundant Configuration Servers

This section describes how to install and set up redundant Configuration Servers.

Installation Recommendations

- To ensure proper redundancy, Genesys recommends running the primary and backup Configuration Servers on separate computers.

> **Warning**
>
> - When both the primary and backup Configuration Servers are running, do not remove the backup Configuration Server Application object from the configuration.
> - You are responsible for ensuring that the configuration options of the primary and backup Configuration Servers are the same, with some exceptions: the log options in the primary Configuration Server can differ from those in the backup Configuration Server configuration.

Prerequisites

- Configuration Layer components are installed and running as described in Deploying Configuration Layer.
- You are logged into Genesys Administrator.

> **Important**
> Once installed, both Configuration Servers must be started from the default account.

Installation and Configuration

1. Configure an Application object for the backup Configuration Server.
   a. Go to **Provisioning > Environment > Applications**, and click **New**.
   b. In the **General** section of the **Configuration** tab:
      1. Enter a descriptive name (not `confserv`) in the **Name** text box.
      2. Select the appropriate template, as follows:
         a. Click the search icon in the **Application Template** field to open a **Browse** dialog box that lists the available application templates. If a Configuration Server template file is not listed, close the dialog box and import the **Configuration_Server_*current-version*.apd** file from the Management Framework 8.5 product CD.
         b. In the **Browse** dialog box, select the Configuration Server template file.
         c. Click **OK**.
   c. In the **Server Info** section:
      1. Select the Host object on which this Configuration Server runs.
      2. Specify the **Listening Ports** that Configuration Server clients must use to connect to this Configuration Server.
      3. In the **Working Directory**, **Command Line**, and **Command Line Arguments** text boxes, do one of the following:
         - Enter the appropriate information in each of the text boxes. For information about command-line parameters, see Starting a Backup Configuration Server.
         - Type a period (`.`) in the **Working Directory** and **Command Line** text boxes, and leave the **Command Line Arguments** text box blank. The information will be filled in automatically when you install the backup Configuration Server, but only if the Installation Package can connect to the primary Configuration Server.
      4. Enter appropriate values for the other mandatory fields (those indicated by red asterisks).
   d. Click **Save & Close** to save the configuration.
2. Install the backup Configuration Server.

Prerequisite

- The backup Configuration Server Application object exists.

Installing on UNIX

a. On the Management Framework 8.1 product CD, go to **configuration_layer/configserver/*operating_system*/**.

b. Type **install** at the command prompt, and press **Enter**.

c. For the installation type, type 2 to select **Configuration Server Master Backup**, and press **Enter**.

d. For the external authentication option, type the number corresponding to the type of External Authentication that will be used (LDAP, Radius, both, or neither), and press **Enter**.

> **Important**
>
> If you select LDAP, be prepared with the URL to access the LDAP Server. For more information about LDAP configuration, see the *Framework External Authentication Reference Manual*

.

e. For the host name of this backup Configuration Server, do one of the following:

- Specify the host name, and press **Enter**.
- Press **Enter** to select the host on which this backup Configuration Server is being installed.

f. Specify the primary Configuration Server, as follows:

1. Specify the primary **Configuration Server Hostname**, and press **Enter**.
2. Specify a value for the **port** for the primary Configuration Server, and press **Enter**.
3. Specify the **User name** of the primary Configuration Server, and press **Enter**.
4. Specify the **Password** for the primary Configuration Server, and press **Enter**.

g. Type the number corresponding to the Application object for the backup Configuration Server that you created, and press **Enter**.

h. Specify the full path of the destination directory, and press **Enter**.

i. If the target installation directory has files in it, do one of the following:

- Type 1 to back up all the files in the directory, and press **Enter**. Then specify the path to where you want the files backed up, and press **Enter**.
- Type 2 to overwrite only the files in this installation package, and press **Enter**. Then, type y to confirm your selection, and press **Enter**. Use this option only if the application already installed operates properly.
- Type 3 to erase all files in this directory before continuing with the installation, and press **Enter**. Then, type y to confirm your selection, and press **Enter**.

The list of file names will appear on the screen as they are extracted and written to the destination

directory.

j. For the product version to install, do one of the following:
   - Type 32 to select the 32-bit version, and press **Enter**.
   - Type 64 to select the 64-bit version, and press **Enter**.

k. Do one of the following:
   - Type y to configure the backup Configuration Server during installation (now), and press **Enter**. Continue to the next step to specify values for the configuration file. For information about Configuration Server configuration options and their values, refer to the *Framework Configuration Options Reference Manual*.
   - Type n to not configure backup Configuration Server during installation. In this case, you have finished installing Configuration Server; do not continue to the next step in this procedure. Before you can start Configuration Server, however, you must create a configuration file and set the configuration options in it, as described in Configuration Server Configuration File.

l. For the **confserv** section:
   1. Specify a value for the backup Configuration Server port, and press **Enter**.
   2. Specify a value for the backup Configuration Server management port, and press **Enter**.

m. For the **dbserver** section:
   1. Type the number corresponding to the database engine that this Configuration Server uses (**dbengine**), and press **Enter**.
   2. Specify the name or alias of the DBMS that handles the Configuration Database (**dbserver**), and press **Enter**.
   3. To specify the name of the Configuration Database (**dbname**), do one of the following:
      - If you are using an Oracle database engine (that is, you typed 3 in Step i), press **Enter**. This value is not required for Oracle.
      - If you are using any other database engine, specify the name of the Configuration Database, and press **Enter**.
   4. Specify the Configuration Database **username**, and press **Enter**.
   5. To specify the Configuration Database **password**, do one of the following:
      - Specify the password, and press **Enter**.
      - Press **Enter** if there is no password; that is, the password field is empty, with no spaces.

When the installation process is finished, a message indicates that installation was successful. The process places the backup Configuration Server in the directory specified during the installation process. The installation script also writes a sample configuration file, `confserv.sample`, in the directory in which the backup Configuration Server is installed.

> **Important**
>
> - If you chose to configure the backup Configuration Server during installation, the sample configuration file, **confserv.sample**, is renamed **confserv.conf**, and the parameters specified in Steps l and m are written to this file.

- If you chose to configure the backup Configuration Server after installation, you must manually rename the sample file as **confserv.conf** and modify the configuration options before you start the backup Configuration Server. See Configuration Server Configuration File.

Installing on Windows

Warning

Genesys does not recommend installation of its components via a Microsoft Remote Desktop connection. The installation should be performed locally.

a. On the Management Framework 8.5 product CD, go to **configuration_layer/configserver/windows**.

b. Double-click **setup.exe** to start the Genesys Installation Wizard.

c. Click **About** on the wizard's **Welcome** page to review the **read_me** file. The file also contains a link to the server's Release Notes file.

d. Click **Next** on the **Welcome** page to proceed with the installation.

e. On the **Maintenance Setup Type** page, select **Install new instance of the application** and click **Next**.

f. On the **Configuration Server Run Mode** page, select **Configuration Server Master Backup** and click **Next**.

g. On the **Configuration Server Parameters** page, specify the **Server Port** and **Management Port** for Configuration Server, and click **Next**.

h. On the **Database Engine Option** page, select the database engine used by Configuration Server, and click **Next**.

i. On the **Database Parameters** page:

   1. Specify the **Database Server Name** and **Database Name**.
   2. Specify the **Database User Name** and **Password**.
   3. Click **Next**.

j. On the **Configuration Server External Authentication** page, select the type of external authentication Configuration Server uses, or select **None** if Configuration Server is not using external authentication. Click **Next**.

k. On the **Connection Parameters to the Genesys Configuration Server** page:

   1. Specify the **Host name** and **Port** of the primary Configuration Server.
   2. Specify the **User name** and **Password** for the primary Configuration Server.

3. Click **Next**.

l. In the upper pane of the **Select Application** page, select the backup Configuration Server Application object that you just configured, and click **Next**.

m. On the **Choose Destination Location** page, the wizard displays the destination directory (if specified during installation) in the **Working Directory** property of the server's Application object. If you entered a period (.) in this property, or if the specified path is invalid, the wizard generates a path to the destination directory in the **c:\ProgramFiles\GCTI\Product Name** format. If necessary, click:

- Browse to select another destination folder. In this case, the wizard will update the Application object's **Working Directory** in the Configuration Database.
- Default to reinstate the path specified in the **Working Directory** property.

Click **Next** to proceed.

n. On the **Ready to Install** information page, click:

- **Back** to update any installation information.
- **Install** to proceed with the installation.

o. On the **Installation Complete** page, click **Finish**.

As a result of the installation, the wizard adds Application icons to the:

- Windows **Add or Remove Programs** window, as a Genesys server.
- Windows **Services** list, as a Genesys service, with Automatic startup type.

For more information about the Configuration Server configuration file, see Configuration Server Configuration File. For information about Configuration Server configuration options and their values, refer to the relevant chapters in the *Framework Configuration Options Reference Manual*.

3. Modify the primary Configuration Server Application object to work with the backup Configuration Server.

Prerequisite

- The primary and backup Configuration Server Application objects exist.

Procedure

a. Go to **Provisioning > Environment > Applications**, and click the primary Configuration Server Application object (named **confserv**) to open its properties.

b. On the **Configuration** tab, open the **Server Info** section.

c. Use the **Browse** button next to the **Backup Server** property to locate the backup Configuration Server Application object you want to use as the backup server.

d. Select **Warm Standby** as the **Redundancy Type**.

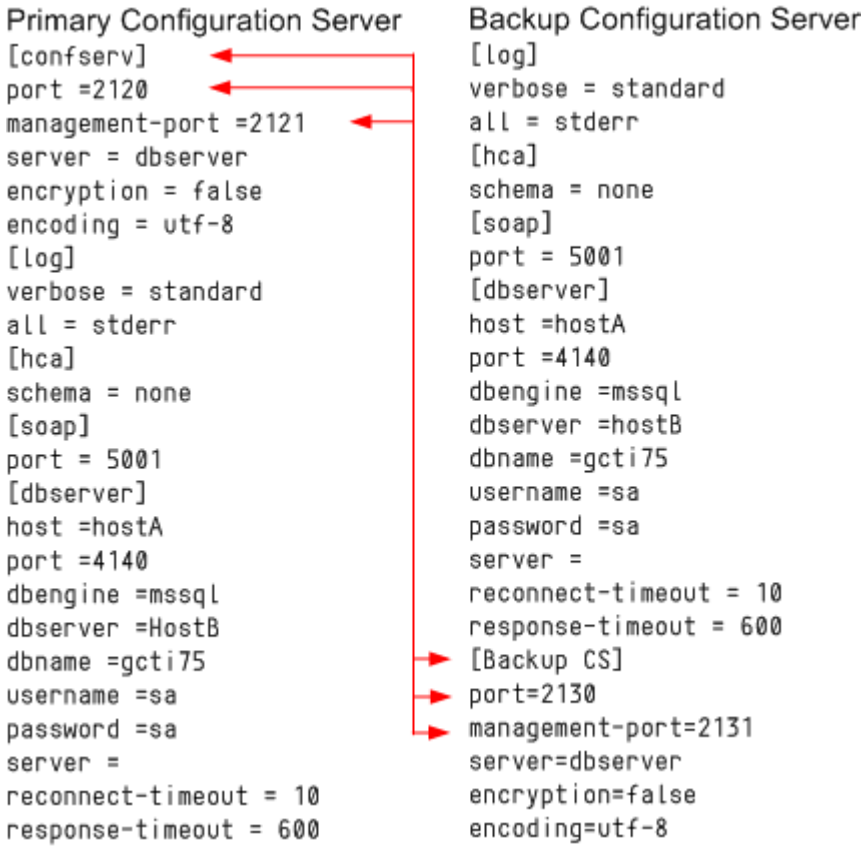
e. Select **Auto-Restart**.

f. Click **Save & Close** to save the changes.

4. If you installed the backup Configuration Server on UNIX and chose to configure it after installation, create and modify the configuration file for the backup Configuration Server. The configuration file for the backup Configuration Server must be the same as that for the primary Configuration Server with the following exceptions:

- The name of the section in the backup Configuration Server configuration file must match the name of the backup Configuration Server Application object.
- The values for the **port** and **management-port** options in the backup Configuration Server configuration file must be those values specified as **Communication Port** and **Management Port** values, respectively, during installation of the backup Configuration Server.
- The log options can be different.

The name of the Configuration Server section must be exactly the same as the name of the Application object for the backup Configuration Server. For both the primary and backup Configuration Servers, specify the same database and user account for accessing this database. The No Default Access for New Users feature must be configured the same in both the primary and backup Configuration Servers. In other words, both Configuration Servers must have the feature either configured or not. Sample configuration files are shown side-by-side in the figure below. The arrows show the differences described in this section.



Sample Configuration Files for Primary and Backup Configuration Servers.

If you installed the backup Configuration Server on UNIX, modify the **run.sh** file to enable the backup server to be started.

a. In a text editor, open the **run.sh** file.

b. Add the following at the end of the command line in the file:

```
-s <section name> -c <configuration file name>
```

3. Manually synchronize options and ports between the redundant Configuration Servers.

4. Manually synchronize high-availability (HA) ports between the redundant Configuration Servers. When Configuration Servers operate in a high-availability (HA) environment, the backup Configuration Server must be ready to take on the primary role when required. This requires that both Configuration Servers are running and that they must have the same information. When you configure redundant Configuration Servers to operate with the Warm Standby redundancy type, the primary Configuration Server uses the connection to the backup to deliver synchronization updates. Genesys recommends that you enable Advanced Disconnect Detection Protocol (ADDP) for this connection.

> **Important**
>
> You can configure multiple ports for any Application of type Server. When multiple ports are configured for a server in a Warm Standby redundancy pair, the number of ports, their **Port IDs**, and the **Listening Mode** settings of the primary and backup servers must match, respectively.

Solution Control Server is required for HA Configuration Servers to switch over. Modify and start the SCS responsible for that pair to work with the Configuration Server running in Primary mode.

The SCS configuration file has a filename extension of **.cfg** (for Windows), and **.conf** (for UNIX). Here is a sample of the contents:

```
[backup_configserver]
host=<backup CS host name>
port=<backup CS port>
name=<SCS application name>
server=primary_configserver
[primary_configserver]
host=<primary CS host name>
port=<primary CS port>
name=<SCS application name>
server=backup_configserver
```

When using HA Configuration Servers, you must restart Solution Control Server to enable it to connect it to the Configuration Server running in Primary mode. If a Master Configuration Server is part of an HA pair, the SCS responsible for that pair of servers must be provisioned with a startup option as follows:

```
scs.exe -f SCS configuration file
```

Starting the Backup Configuration Server

When starting a backup Configuration Server, specify the following values in the startup command line:

-s	The name of the Configuration Server section within the configuration file for the backup Configuration Server.
-c	The name of the configuration file that contains configuration information for the backup Configuration Server.

For a description of the command-line parameters specific to Configuration Server, refer to Configuration Server.

Starting on UNIX

> **Important**
> Make sure you have modified the **run.sh** file as directed in step 5, above.

- To start from Genesys Administrator, refer to Starting and Stopping with the Management Layer.
- To start manually, go to the directory in which the backup Configuration Server is installed, and do one of the following:
  - To use only the required command-line parameters, type the following command line:
    ```
    sh run.sh
    ```
  - To specify the command line yourself, or to use additional command-line parameters, type the following command line:
    ```
    confserv -s <section name> -c <configuration file name> [<additional parameters as required>]
    ```

Starting on Windows

- To start as a Windows Service, refer to Starting and Stopping with Windows Services Manager.
- To start from Genesys Administrator, refer to Starting and Stopping with the Management Layer.
- To start manually, do one of the following:
  - Use the **Start > Programs** menu.
  - To use only the required command-line parameters, go to the directory in which the backup Configuration Server is installed, and double-click the file **startServer.bat**.
  - To specify the command line yourself, or to use additional command-line parameters, open the MS-DOS window, go to the directory in which the backup Configuration Server is installed, and type the following command line:
    ```
    confserv.exe -s <section name> -c <configuration file name> [<additional parameters as required>]
    ```

Operating Configuration Server in HA mode

Configuration Server won't support a manual switchover request and is stopped instead to promote another instance to become primary. If Configuration Server receives a request to switch over from the management layer, then it should be recycled at a later time before it can become primary again.

The **backupmode-restart** option, along with the **AutoRestart** option, ensures this happens automatically.

Configuring ADDP between Redundant Configuration Servers

Advanced Disconnect Detection Protocol (ADDP) is supported between primary and backup Configuration Servers. Use the new configuration options **protocol**, **addp-timeout**, **addp-remote-timeout**, and **addp-trace**, setting them in the configuration server section of the configuration files for both Configuration Servers. For the primary Configuration Server, this section is called **[confserv]**. For the backup Configuration Server, this section has the same name as the backup Configuration Server Application object. For detailed descriptions of these options, refer to the *Framework Configuration Options Reference Manual*.

For example, in a primary Configuration Server configuration file, the ADDP options would appear as follows:

```
[confserv]
...
protocol=addp
addp-timeout=16
addp-remote-timeout=32
addp-trace=both
...
```

For more information about ADDP, see Advanced Disconnect Detection Protocol.

Redundant (HA) Message Servers

This section describes how to deploy redundant Message Servers.

Redundancy

Redundant Message Servers support only the Warm Standby redundancy type, with the addition that the data is synchronized between the primary and backup servers.

Deploying Redundant Message Servers

This section describes how to install and set up redundant Message Servers.

Installation Recommendations

If you are installing the primary and backup Message Servers on the same host computer:

- Install them in different directories.
- Specify a different port number for each server.

Prerequisites

- Configuration Layer components are installed and running as described in Deploying Configuration Layer.
- You are logged into Genesys Administrator.

> **Important**
> Once installed, the two Message Servers must be started from the same account.

Installation and Configuration

> **Tip**
> (Optional) If the backup Message Server is to reside on a remote Host, you can deploy

> it to that Host using Genesys Administrator Extension. For instructions, refer to the *Framework 8.1 Genesys Administrator Extension Help*.

1. Configure an Application object for the backup Message Server.
   a. Go to **Provisioning > Environment > Applications**.
   b. If the Application object for this backup Message Server does not already exist, click **New** to create it.
   c. In the **General** section of the **Configuration** tab:
      1. Enter a descriptive name in the **Name** text box.
      2. Select the appropriate template, as follows:
         a. Click the search icon in the **Application Template** field to open a **Browse** dialog box that lists the available application templates. If a Message Server template file is not listed, close the dialog box and import the **Message_Server_*current-version*.apd** file from the Management Framework 8.1 product CD.
         b. In the **Browse** dialog box, select the Message Server template file.
         c. Click **OK**.
   d. In the **Server Info** section of the **Configuration** tab, enter the following information, as required:
      1. In the **Host** field, click the magnifying glass icon to select the Host object on which this Message Server is running.
      2. For each listening port that an application must use to connect to this Message Server:
         a. In the **Listening Ports** field, click **Add**.
         b. Enter the port properties in the **Port Info** dialog box.
         c. Click **OK**.
      3. For the **Working Directory**, **Command Line**, and **Command Line Arguments** fields, do one of the following:
         - Enter the appropriate information in each of the text boxes. For information about command-line parameters, see Starting a Backup Message Server.
         - Type a period (`.`) in the **Working Directory** and **Command Line** text boxes, and leave the **Command Line Arguments** text box blank. The information will be filled in automatically when you install Message Server, but only if the Installation Package can connect to Configuration Server.
      4. Select the **Auto-Restart** check box.
   e. Click **Save and Close** in the toolbar to save the new object.
2. If you did not deploy the backup Message Server using Genesys Administrator Extension, install it now.
3. Modify the primary Message Server Application object to work with the backup Message Server.

Prerequisite

- The primary and backup Message Server Application objects exist.

Procedure

a. Go to **Provisioning > Environment > Applications**, and double-click the primary Message Server Application object to open its properties.

b. In the **Server Info** section of the **Configuration** tab:

1. Select the backup Message Server Application object.
2. Select `Warm Standby` as the redundancy type.
3. Select `Auto-Restart` if required.

c. Click **Save and Close** to save the configuration.

4. If you installed the backup Message Server on UNIX, check the **run.sh** file and modify it, if necessary, so the Application can be started properly.

a. In a text editor, open the **run.sh** file.

b. Check if the following parameters are currently in the command line in the file, and if not, add them:
```
-host <configuration server host> -port <configuration server port> -app
<application object name>
```

5. Synchronize options and ports between the redundant Message Servers.

Starting the Backup Message Server

When starting a backup Message Server, be sure to use the following command-line options:

-host	The name of the host on which Configuration Server is running.
-port	The communication port that client applications must use to connect to Configuration Server.
-app	The exact name of the backup Message Server `Application` object as configured in the Configuration Database.

For a description of the command-line parameters specific to Message Server, refer to Message Server.

On UNIX

Starting the Backup Message Server on UNIX

Prerequisite

- The **run.sh** file has been modified accordingly. See Step 5, above.

Procedure

- To start from Genesys Administrator, refer to Starting and Stopping with the Management Layer.
- To start manually, go to the directory in which the backup Message Server is installed, and do one of the following:
  - To use only the required command-line parameters, type the following command line:
    ```
    sh run.sh
    ```
  - To specify the command line yourself, or to use additional command-line parameters, type the following command line:
    ```
    MessageServer -host <Configuration Server host> -port <Configuration Server port>
    -app <backup Message Server Application> [<additional parameters and arguments as
    required>]
    ```

On Windows

Starting the Backup Message Server on Windows

- To start as a Windows Service, refer to Starting and Stopping with Windows Services Manager.
- To start from Genesys Administrator, refer to Starting and Stopping with the Management Layer.
- To start manually, do one of the following:
  - Use the **Start > Programs** menu.
  - To use only the required command-line parameters, go to the directory in which the backup Message Server is installed, and double-click the file **startServer.bat**.
  - To specify the command line yourself, or to use additional command-line parameters, open the MS-DOS window, go to the directory in which the backup Message Server is installed, and type the following command line:
    ```
    MessageServer.exe -host <Configuration Server host> -port <Configuration Server
    port> -app <backup Message Server Application> [<additional parameters and
    arguments as required>]
    ```

Redundant (HA) Solution Control Servers

This section describes how to deploy redundant Solution Control Servers.

Redundancy

Redundant Solution Control Servers support only the Warm Standby redundancy type, with the addition that the data is synchronized between the primary and backup servers.

Deploying Redundant Solution Control Servers

This section describes how to install and set up redundant Solution Control Servers.

Recommendations

If you are installing the primary and backup Solution Control Servers on the same host computer:

- Install them in different directories.
- Specify a different port number for each server.

Prerequisites

- Configuration Layer components are installed and running as described in Deploying Configuration Layer.
- You are logged into Genesys Administrator.

> **Important**
> Once installed, the two Solution Control Servers must be started from the same account.

Installation and Configuration

> **Tip**
> (Optional) If the backup Solution Control Server is to reside on a remote Host, you can

deploy it to that Host using Genesys Administrator Extension. For detailed instructions, refer to *Genesys Administrator Extension Help*.

1. Configure an Application object for the backup Solution Control Server.
   a. Go to **Provisioning > Environment > Applications**.
   b. If the Application object for this backup Solution Control Server does not already exist, click **New** to create it.
   c. In the **General** section of the **Configuration** tab:
      1. Enter a descriptive name in the **Name** text box.
      2. Select the appropriate template, as follows:
         a. Click the search icon in the **Application Template** field to open a **Browse** dialog box that lists the available application templates. If a Solution Control Server template file is not listed, close the dialog box and import the **Solution_Control_Server_*current-version*.apd** file from the Management Framework 8.1 product CD.
         b. In the **Browse** dialog box, select the Solution Control Server template file.
         c. Click **OK**.
   d. In the **Server Info** section of the **Configuration** tab, enter the following information, as required:
      1. In the **Host** field, click the magnifying glass icon to select the Host object on which this Solution Control Server is running.
      2. For each listening port that an application must use to connect to this Solution Control Server:
         a. In the **Listening Ports** field, click **Add**.
         b. Enter the port properties in the **Port Info** dialog box.
         c. Click **OK**.
      3. For the **Working Directory**, **Command Line**, and **Command Line Arguments** fields, do one of the following:
         - Enter the appropriate information in each of the text boxes. For information about command-line parameters, see Starting a Backup Solution Control Server.
         - Type a period (`.`) in the **Working Directory** and **Command Line** text boxes, and leave the **Command Line Arguments** text box blank. The information will be filled in automatically when you install Solution Control Server, but only if the Installation Package can connect to Configuration Server.
      4. Select the **Auto-Restart** check box.
   e. Click **Save & Close** in the toolbar to save the new object.
2. If you did not deploy the backup Solution Control Server remotely using Genesys Administrator, install it now.
3. Modify the primary Solution Control Server Application object to work with the backup Solution Control Server.

Prerequisites

- The primary and backup Solution Control Server Application objects exist.

Procedure

a. Go to **Provisioning > Environment > Applications**, and double-click the primary Solution Control Server Application object to open its properties.

b. In the **Server Info** section of the **Configuration** tab:

1. Select the backup Solution Control Server Application object.
2. Select `Warm Standby` as the redundancy type.
3. Select `Auto-Restart` if required.

c. Click **Save & Close** to save the configuration.

4. If you installed the backup Solution Control Server on UNIX, check the **run.sh** file and modify it, if necessary, so the application can be started properly.

a. In a text editor, open the **run.sh** file.

b. Check if the following parameters are currently in the command line in the file, and if not, add them: `-host <configuration server host> -port <configuration server port> -app <SCS application object name>`

5. Synchronize options and ports between the redundant Solution Control Servers.

Starting the Backup Solution Control Server

When starting a backup Solution Control Server, be sure to use the following command-line options:

-host	The name of the host on which Configuration Server is running.
-port	The communication port that client applications must use to connect to Configuration Server.
-app	The exact name of the backup Solution Control Server Application object as configured in the Configuration Database.

For a description of the command-line parameters specific to Solution Control Server, refer to Solution Control Server.

On UNIX

Prerequisite

- The **run.sh** file has been modified accordingly. See Step 4, above.

Procedure

- To start from Genesys Administrator, refer to Starting and Stopping with the Management Layer.
- To start manually, go to the directory in which the backup Solution Control Server is installed, and do one of the following:
  - To use only the required command-line parameters, type the following command line:
    ```
    sh run.sh
    ```
  - To specify the command line yourself, or to use additional command-line parameters, type the following command line:
    ```
    scs -host <Configuration Server host> -port <Configuration Server port> -app
    <backup Solution Control Server Application> [<additional parameters and arguments
    as required>]
    ```

On Windows

Starting the Backup Solution Control Server on Windows

- To start as a Windows Service, refer to Starting and Stopping with Windows Services Manager.
- To start from Genesys Administrator, refer to Starting and Stopping with the Management Layer.
- To start manually, do one of the following:
  - Use the **Start > Programs** menu.
  - To use only the required command-line parameters, go to the directory in which the backup Solution Control Server is installed, and double-click the file **startServer.bat**.
  - To specify the command line yourself, or to use additional command-line parameters, open the MS-DOS window, go to the directory in which the backup Solution Control Server is installed, and type the following command line:
    ```
    scs.exe -host <Configuration Server host> -port <Configuration Server port> -app
    <backup Solution Control Server Application> [<additional parameters and arguments
    as required>]
    ```

Redundant (HA) SNMP Master Agents

You can configure redundant SNMP Master Agent Applications for use by Solution Control Server (SCS) to communicate using the AgentX protocol if the current SNMP Master Agent is not available.

> **Important**
>
> You must use 8.5.1 or later versions of SCS and LCA to be able to configure multiple SNMP Master Agent applications as HA pairs that SCS will recognize and connect. With earlier versions of Management Layer components, you must configure each SNMP Master Agent as standalone and make sure autorestart is enabled to allow the automated restart of SNMP in case of failure. In this situation, SCS will always use a single configured SNMP MA to report its status using the AgentX protocol.

Both Net-SNMP and Genesys SNMP Master Agent implementations of SNMP support redundant configuration when used with SCS and LCA 8.5.1 or newer. A redundant configuration consists of two SNMP Master Agent Application objects, one primary and one backup. When SCS loses a connection with the primary SNMP Master Agent, SCS switches all NMS communications to the backup. The only difference is in the redundancy type set in the primary Genesys SNMP Master Agent Application object—`Hot Standby` for the Genesys SNMP Master Agent implementation, and `Not Specified` for Net-SNMP.

> **Important**
>
> The Application status for both the primary and backup modes is displayed as Primary for the SNMP Master Agent applications (Genesys Net-SNMP) that are configured as an HA pair.

Configuring Redundant SNMP Master Agents

To configure redundant SNMP Master Agents, do the following:

1. Configure two SNMP Master Agent Application objects using the same instructions as for stand-alone SNMP Master Agents. Determine which one of these will be, in normal operation, the primary object, and which will be the backup object.
2. In the **Server Info** section of the **Configuration** tab of the Application object that is to be the primary SNMP Master Agent:
   - Select the other SNMP Master Agent Application object (the one that is to be the backup object) as the backup server.
   - Specify the **Redundancy Type** appropriate for which SNMP implementation you are using, as follows:

- For Net-SNMP, select `Not Specified`.
- For Genesys SNMP Master Agent, select `Hot Standby`.

Starting and Stopping a Backup SNMP Master Agent

To start and stop a backup SNMP Master Agent, whether you are using Net-SNMP or Genesys SNMP Master Agent, refer to Starting and Stopping Framework Components.

Sharing the Load Configurations

Large enterprises often run contact center operations at numerous locations worldwide. Yet, for Genesys software to function as a single unit it is usually critical that all configuration objects comprising an enterprise be stored in a single Genesys Configuration Database. Under these circumstances, network delays, component failures, and similar factors might complicate or slow down the operations of a large enterprise.

By operating two Framework components in different modes you can somewhat simplify the operation of a distributed installation with a single Configuration Database:

- Distributed configuration environments
- Distributed management environments
- Distributing call loads

Starting Configuration Server in Proxy mode or Solution Control Server in Distributed mode requires special licenses. Refer to the *Genesys Licensing Guide* for more information.

Distributed Configuration Environments

In a distributed configuration environment, the master Configuration Server is running at the site where the Configuration Database is located. Configuration Servers at multiple remote sites are working in so-called Proxy mode and are connecting to the master Configuration Server.

Distributed Management Environments

In a distributed management environment, Solution Control Servers are communicating with each other and controlling a particular part of the Genesys environment while running at multiple remote sites (but within the same configuration environment).

Distributing Loads

Genesys recommends deploying additional of Configuration Server Proxies and Solution Control Servers in Distributed mode to distribute loads.

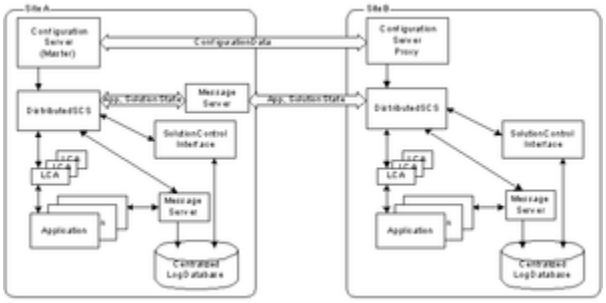
The number of instances deployed at the same site for purposes of load distribution should be calculated based on the number of clients to serve (for Configuration Server Proxy) and the number of hosts to control (for Distributed Solution Control Servers). Refer to the Management Framework section of the *Genesys Hardware Sizing Guide*.

Genesys also recommends using Configuration Server Proxy and Distributed Solution Control Servers

in multi-site environments.

Architecture

The figure below illustrates how Configuration Server Proxy and Distributed Solution Control Servers fit into a Genesys configuration environment. This diagram does not include distributed components for Disaster Recovery/Business Continuity.



Distributed Installation

Configuration Server Proxy

Using Configuration Server Proxy increases the robustness of the whole system, decreases the number of client connections to Configuration Server, and minimizes network traffic. When Configuration Server is configured, existing clients can continue, and new clients start, their operations when Configuration Server fails. In addition, after Configuration Server recovers, the client reconnect takes far less time than if all clients were directly connected to Configuration Server.

Configuration Server Proxy is an Application of Configuration Server type operating in a special mode. As such, it seemlessly replaces Configuration Server for the clients. You can also configure Configuration Server Proxy permissions so that clients of a particular proxy access only the part of the configuration environment relevant to their site. See User Authorization or the *Genesys Security Deployment Guide* for more information about setting permissions.

How it Works

In a distributed configuration environment, the master Configuration Server is running at the site where the Configuration Database is located. Configuration Server Proxies at multiple remote sites are connecting to the master Configuration Server.

Instead of sending all the requests to Configuration Server, Configuration Server clients that require read-only access to Configuration Server can operate with one or more Configuration Server Proxies. Configuration Server Proxy passes messages to and from Configuration Server. Moreover, the proxy keeps the configuration data in its memory and responds to client data requests. Any configuration data updates are passed immediately to Configuration Server Proxy, so that it is always up to date; no additional configuration is required to specify an update interval.

Configuration Server Proxy Functions

- Receives subscription requests from clients and handles them without passing the requests to Configuration Server.
- Stores in internal memory all configuration data it receives from Configuration Server.
- Receives notifications on data changes from Configuration Server, updates internal memory, and passes notifications to clients.
- Receives read-data requests from clients and responds to them using the data stored in the internal memory.

> **Important**
>
> - Always run Configuration Server Proxy under the default account **Environment\default**.
> - A hierarchical configuration of Configuration Server Proxies—for example, a Configuration Server Proxy application working with another Configuration Server Proxy that operates directly with Configuration Server—is not supported.

Deploying Configuration Server Proxy

Important

- To ensure faultless operation, all Configuration Servers in the configuration environment must be running the same release. Configuration Server Proxy may start with a master Configuration Server running a later release, but only during the migration process. Refer to the *Framework Migration Guide* for more information.
- When deploying Configuration Server Proxy, keep in mind that redundancy type is critical. Specifically:
  - If Configuration Server Proxy is running as a single proxy server, set the redundancy type to `Not Specified`.
  - If Configuration Server Proxy is part of a HA pair and/or is configured as a primary or backup, set the redundancy type to `Warm Standby`.

Prerequisites

- The Configuration Layer components, including the master Configuration Server, are installed and running as described in Deploying Configuration Layer.
- You are logged in to Genesys Administrator.

Installation and Configuration

1. Configure as many instances of Configuration Server Proxy as needed.

Prerequisite

- You are logged in to Genesys Administrator.

Procedure

a. Go to **Provisioning > Environment > Applications**, and select **New** in the toolbar. This opens a **Browse** dialog box that lists available application templates. If a Configuration Server Proxy template file is not listed, do one of the following:
   - Import the **Configuration Server Proxy_*current-version*.apd** file from the Management Framework 8.5 product CD.
   - Create a new template using the procedure in Application Templates, and repeat this step.

b. In the **Browse** dialog box, select the Configuration Server Proxy template file.

c. In the **General** section of the **Configuration** tab:

   i. Enter a descriptive name in the **Name** text box.

   ii. In the list of **Connections**, add a connection to the master Configuration Server Application object. If redundant master Configuration Servers are configured, specify a connection to the primary Configuration Server.

d. In the **Server Info** section:

   i. Select the **Host** object on which this Configuration Server Proxy runs.

   ii. Specify the **Listening Ports** that Configuration Server Proxy clients must use to connect to this Configuration Server.

   iii. In the **Working Directory**, **Command Line**, and **Command Line Arguments** text boxes, do one of the following:

   - Enter the appropriate information in each of the text boxes. For information about command-line parameters, see Starting Configuration Server Proxy.
   - Type a period (`.`) in the **Working Directory** and **Command Line** text boxes, and leave the **Command Line Arguments** text box blank. The information will be filled in automatically when you install Configuration Server Proxy, but only if the Installation Package can connect to the master Configuration Server.

   iv. Enter appropriate values for the other mandatory fields (those indicated by red asterisks).

   v. In the **Log On As Account** field, you must use the default account, **Environment\default**.

> **Warning**
> Always run Configuration Server Proxy under the default account **Environment\default**.

e. (Optional) On the **Options** tab:

   - If you want this Configuration Server Proxy to be writable, set the option **proxy-writable** in the **[csproxy]** section to `true`.
   - Set the values of the log configuration options.

f. Click **Save & Close** to save the configuration.

2. Install the corresponding number of Configuration Server Proxies.

Prerequisite

- The Configuration Server Proxy Application object is created.

On UNIX

Installing Configuration Server Proxy on UNIX

a. On the Management Framework 8.5 product CD, go to **configuration_layer/configserver/operating_system**.

b. Type `install.sh` at the command prompt, and press **Enter**.

c. For the installation type, type 3 to select Configuration Server Proxy, and press **Enter**.

d. To specify the host name for this Configuration Server Proxy, do one of the following:

- Type the name of the host, and press **Enter**.
- Press **Enter** to select the current host.

e. Enter the Master Configuration Server host name, and press **Enter**.

f. Enter the Master Configuration Server network port, and press **Enter**.

g. Enter the Master Configuration Server user name, and press **Enter**.

h. Enter the Master Configuration Server password, and press **Enter**.

i. The installation displays the list of Application objects of the specified type configured for this Host object. Type the number corresponding to the Configuration Server Proxy Application object you configured in step 1, and press **Enter**.

j. To specify the destination directory, do one of the following:

- Press **Enter** to accept the default.
- Enter the full path of the directory, and press **Enter**.

k. If the target installation directory has files in it, do one of the following:

- Type 1 to back up all the files in the directory, and press **Enter**. Specify the path to which you want the files backed up, and press **Enter**.
- Type 2 to overwrite only the files in this installation package, and press **Enter**. Then type y to confirm your selection, and press **Enter**. Use this option only if the application already installed operates properly.
- Type 3 to erase all files in this directory before continuing with the installation, and press **Enter**. Then type y to confirm your selection, and press **Enter**.

The list of file names will appear on the screen as the files are copied to the destination directory.

l. Specify the full path to, and the exact name of, the license file that Configuration Server Proxy will use, and press `Enter`.

When the installation process is finished, a message indicates that installation was successful. The process places Configuration Server Proxy in the directory that you specified during installation.

On Windows

Installing Configuration Server Proxy on Windows

> **Warning**
> Genesys does not recommend installation of its components via a Microsoft Remote Desktop connection. The installation should be performed locally.

a. On the Management Framework 8.5 product CD, **configuration_layer/configserver/windows**.

b. Locate and double-click **setup.exe** to start the Genesys Installation Wizard.

c. Use the **About** button on the wizard's **Welcome** page to review the **read_me** file. This file also contains a link to the server's Release Notes file.

d. Click **Next**.

e. On the **Configuration Server Run Mode** page, select **Configuration Server Proxy**.

f. On the **Connection Parameters to the Genesys Configuration Server** page, specify the host name, port, user name, and password for the Master Configuration Server, then click **Next**.

g. On the **Select Application** page, select the name of the Configuration Server Application object that you created in step 1, and click **Next**.

h. On the **Access to License** page, specify the license access type and the appropriate parameters, and click **Next**.

i. On the **Choose Destination Location** page, the wizard displays the destination directory specified in the **Working Directory** property of the server's Application object. If the specified path is invalid, the wizard generates a path to **c:\Program Files\GCTI*Singletenant or Multitenant* Configuration Server**. If necessary, click:

   - **Browse** to select another destination folder. In this case, the wizard will update the Application object's **Working Directory** property in the Configuration Database.
   - **Default** to reinstate the path specified in the **Working Directory** property.

   Click **Next** to proceed.

j. On the `Ready to Install` information page, click:

   - **Back** to update any installation information.
   - **Install** to proceed with the installation.

k. On the **Installation Complete** page, click **Finish**. When the installation process is finished, a message indicates that installation was successful. The process places Configuration Server Proxy in the directory that you specified during the installation process.

3. Modify each Configuration Server Proxy client to work with Configuration Server Proxy.

Prerequisites

- The Configuration Server Proxy Application object is created.

- You have identified the client applications that are to operate with this particular Configuration Server Proxy.
- You are logged in to Genesys Administrator.

> **Important**
> Repeat this procedure for each application that is to be a client of Configuration Server Proxy.

Procedure

a. Go to **Provisioning > Environment > Applications**, and double-click the client Application object that you want to connect to Configuration Server Proxy.
b. In the **General** section of the **Configuration** tab, add a Connection to the Configuration Server Proxy to which the client application should connect.
c. Click **Save & Close** to save the configuration changes. Now, when you start the client application, it will operate with the given Configuration Server Proxy.
d. Start the client application using one of the following methods:
   - From Genesys Administrator.
   - From the command line. In this case, you must use the parameters **-host** and **-port** to point to the Configuration Server Proxy with which the application will be operating.
e. Click **Save & Close** to save the changes.

4. (Optional) Configure redundant Configuration Server Proxies.

Prerequisites

- A primary Configuration Server Proxy Application object already exists.
- You are logged in to Genesys Administrator.

Procedure

a. Configure an Application object for the backup Configuration Server Proxy as described in step 1, above.
b. Install a backup Configuration Server Proxy as described in step 2, above.
c. In Genesys Administrator, go to **Provisioning > Environment > Applications** and double-click the primary Configuration Server Proxy client Application object.
d. On the **Configuration** tab, open the **Server Info** section.
e. In the **Backup Server** field, specify the Configuration Server Proxy application you want to use as the backup server.
f. Open the **Properties** dialog box of the Configuration Server Proxy application that you want to configure as a primary server.
g. In the **Redundancy Type** field, select `Warm Standby`.
h. Select `Auto-Restart`.

i. Click **Save & Close** to save the configuration changes.

Configuring a Dedicated Port for Client User Interface Applications

Warning

- Genesys strongly recommends that you do not restrict the default port to accept only client UI applications. Because the backup Configuration Server communicates with Configuration Server via the default port, and because many other Genesys Server applications cannot operate properly with being connected to the default port, restricting the default port would disable you from using these additional beneficial components.
- Ports that have been dedicated as **HA sync** (in the **Server Info** section of the port's **Configuration** tab in Genesys Administrator) cannot be provisioned to accept only client UI applications.

Dedicated ports can also be configured on Configuration Server Proxy in the same way that they are configured on the master Configuration Server. Like the master server, the proxy server must sit inside the firewall, as shown in the following illustration: thumb|center|Dedicated Port on Master Configuration Server Proxy

Use the instructions here to configure the dedicated port.

Starting Configuration Server Proxy

Important

- Always run Configuration Server Proxy under the default account **Environment\default**.
- If using a primary-backup pair of Configuration Server Proxies, follow the same starting procedure for both primary and backup applications but make sure you specify the correct application name for each.

The startup command line for Configuration Server Proxy must identify the:

- Configuration Server Proxy executable file
- Configuration Server Proxy application name (the **-app** parameter)
- Configuration Server host (the **-host** parameter)
- Configuration Server port (the **-port** parameter)
- Configuration Server Proxy license file or license server location (the **-l** parameter)

Configuration Server Proxy supports the command-line parameters common to Genesys server applications, as described in Starting and Stopping Manually.

> **Tip**
> If you want to generate logs logging during startup and initialization (referred to as *bootstrap logging*) of Configuration Server Proxy, start the server from the command line and include the **-log-<log option name> <log-type>** parameter. Optionally, you can store these logs in a file separate from the operations logs—also include the **-log-<log-type> <filename>** parameter in the startup command.

On Unix

Starting Configuration Server Proxy on UNIX

Go to the directory in which Configuration Server Proxy is installed, and do one of the following:

- To use only the required command-line parameters, type the following command line: `sh run.sh`
- To specify the command line yourself, or to use additional command-line parameters, type the following command line:
  ```
  confserv -host <Configuration Server host> -port <Configuration Server port> -app <CS proxy application objects name> [<additional parameters and arguments as required>]
  ```

On Windows

Starting Configuration Server Proxy on Windows

Do one of the following:

- Use the **Start > Programs** menu.
- To use only the required command-line parameters, go to the directory in which Configuration Server Proxy is installed, and double-click the **startServer.bat** file.
- To specify the command line yourself, or to use additional command-line parameters, open the MS-DOS

window, go to the directory in which Configuration Server Proxy is installed, and type the following command line:

```
confserv.exe -host <Configuration Server host> -port <Configuration Server port> -app
<CS proxy application objects name> [<additional parameters and arguments as
required>]
```

Writable Configuration Server Proxies

By default, Configuration Server Proxy provides read-only access to configuration data. Configuration Server clients that require write access to Configuration Server must still connect directly to Configuration Server. Some of Genesys Supervisor- and Agent-facing applications (such as Workspace Desktop Edition), while deployed in high numbers, require write access to configuration data and should be deployed against Configuration Server Proxy in Writable mode.

Administrative applications, such as Genesys Administrator, should still connect to the Master Configuration Server to perform complex configuration updates, because Configuration Server Proxy in writable mode is not designed to handle all types of configuration updates. Updates made in bulk might result in a significant extra load on the system when done by the Proxy server rather than the Master server.

To configure a Configuration Server Proxy as writable, use the Configuration Server Proxy configuration option **proxy-writable**. For more information about this option, refer to the *Framework Configuration Options Reference Manual*.

Redundant Configuration Server Proxies

The high-availability (HA) architecture implies the existence of redundant applications, a primary and a backup, monitored by a management application.

Like Configuration Server, Configuration Server Proxy supports the Warm Standby redundancy type between redundant Configuration Server Proxies. For more information, refer to Redundant Configuration Servers.

HA Configuration Server Proxy supports ADDP between the pair of proxy servers if ADDP has been enabled between the master Configuration Server and Configuration Server Proxy in the Connections tab of the proxy server. The primary and backup Configuration Server Proxies also use these ADDP settings to communicate with each other.

Prior to release 8.1.3, when a switchover occurred between the primary and backup Configuration Server Proxies, Configuration Server Proxy clients had to read configuration information anew and reestablish the connections to the backup server themselves. Especially in large configuration environments, this often led to detrimental effects on system performance, leading clients to question the usefulness of the backup proxy server.

Starting in release 8.1.3, client connections are restored automatically to the backup Configuration Server Proxy when it switches to primary mode, if the connection between the client and primary Configuration Server Proxy is lost, because the primary proxy server is stopped. This makes the switchover practically invisible to clients, and essentially eliminates the performance impact on the

system. This restoration is made possible by the backup Configuration Server Proxy keeping its own record of client connections and disconnections. Under normal conditions, the primary proxy server notifies the backup proxy of client connections and disconnections, which the backup stores in its History Log Database. When the backup switches to primary mode, it is able to restore client connections based on the connection and disconnection information it has stored.

If the connection between the primary and backup servers is lost, prior to switchover, the session is not restored. Clients of the Configuration Server Proxy must reregister and read all data from scratch.

> **Important**
>
> You cannot separate two Configuration Server Proxies configured as an HA pair into two standalone servers at runtime. You must stop, re-configure, and then restart each server.

Limiting the Number of Proxy Servers Loading and Reloading Data

This feature limits the number of Configuration Server Proxies that can load or reload data from the master Configuration Server and also provides an option to delay the actual moment when each Configuration Server Proxy attempts to reload its cached data after master Configuration Server becomes available following a prolonged disconnect or downtime.

This feature can be implemented on both master Configuration Server and each of Configuration Server proxies, with both implementations complementing each other. On Configuration Server Proxy side, it can delay sending first read request to the master Configuration Server after the connection has been re-established and it has been confirmed that full reload is required to get the system in-sync. This delay can future be tailored for primary and backup instances of each Configuration Server proxy HA pair. On Configuration Server master side, it can delay responses to a Configuration Server Proxy requests attempting full data re-reads, based on current load the server and number of proxies already performing re-read. This feature will not affect default behavior of master and proxy servers during short connection loss, as session restoration does not involve client disconnects and data reload; it only works in the case when synchronization require re-reading of entire configuration data by some or all proxy servers.

Implementation

To specify the maximum number of Configuration Server Proxies allowed to concurrently load or reload data from the master Configuration Server, use the following options in the **system** section of the master Configuration Server Application object:

1. Ensure that transaction serialization is enabled (serialize-write-transactions is set to `true`). If transaction serialization is not enabled, you cannot use this feature.
2. Set proxy-load-max to the maximum number of proxy servers allowed. If this option is not configured or is set to `false`, there is no limit.

3. Set the delay reload period for Configuration Server Proxies using the delay-reload and delay-reload-backup options. If these options are set to `0` (the default), the delay reload feature is disabled.

Using Configuration Server Proxy with External Authentication Systems

In distributed systems prior to release 8.0, external authentication was configured only on the Master Configuration Server, and each Configuration Server Proxy passed authentication requests to it. Now, RADIUS and LDAP external authentication, starting in release 8.0 and 8.1 respectively, can be configured on the Master Configuration Server and on each Configuration Server Proxy. Therefore, each Configuration Server Proxy can process authentication requests itself, and does not need to pass them on to the Master Configuration Server. For more information about setting up external authentication on Configuration Server Proxy, refer to the *Framework External Authentication Reference Manual*.

Load-Balanced Configuration Server Proxies for Agent-Facing Applications

Starting in release 8.5.1, you can integrate load balancing into a system of Configuration Server Proxies. This enables a group of Configuration Server Proxies to share the processing load (client connections).

The benefits of load-balancing are two-fold:

- Deploying a pool of Configuration Server Proxies enables you to easily manage environments in which the capacity of a single proxy server is not enough to handle all agent-facing clients (such as with the Workspace Desktop Edition).

> **Important**
>
> Refer to the Hardware Sizing Guide to determine the capacity of maximum incoming connections for a single Configuration Server Proxy.

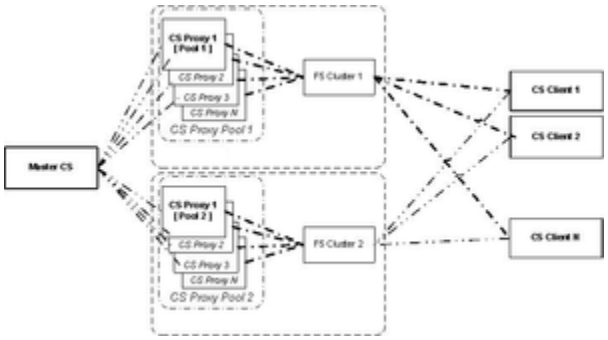
- If any Configuration Server Proxy is not operational, new client connections are not distributed to that proxy server automatically.

High Level Architecture

This solution requires the use of a third-party load balancer (F5). A set of standalone Configuration Proxy Instances are deployed on several hosts, subject to limitations noted in the Hardware Sizing Guide. The resources of each host (memory and number of CPUs) must be sufficient to allow the launching of, and running under load, all instances of Configuration Server Proxy assigned to it. The

master Configuration Server maintains a connection to each Configuration Server Proxy. An F5-based hardware load balancer is connected to all of the proxy servers in the group, and provides a single virtual IP address and port to which the clients of those proxies connect.

This can be extended to multiple groups of Configuration Server Proxies, each group served by a different load balancer. This is shown in the following diagram.



Load-Balanced Configuration Server Proxies

Support of Agent-Facing user Interface Applications

The following Genesys application supports working with a pool of Configuration Server Proxies behind a hardware load balancer: Workspace Desktop Edition (formerly called Interaction Workspace).

Limitations

This solution also has these limitations:

- Session restoration on the Connection Server Proxy side is not supported in this type of deployment.
- The built-in Kerberos protection against ticket sniffing by caching used tickets is turned off. Clients are connecting to a pool of servers, and each proxy server has a separate ticket cache.

Warning

Support for this solution has been tested by Genesys on a limited number of applications. Connecting unsupported applications to Configuration Server Proxy using a hardware load balancer may result in performance issues and feature degradation. Refer to the documentation for a particular application to confirm that it supports the Configuration Server Proxy load-balancing architecture.

Configuration

To set up the load-balanced solution, do the following:

1. Install and configure F5 with VIP, and use the round-robin methodology to distribute connections to clients.

2. Install and configure the Configuration Server Proxies as individual **Application** objects of type ConfigurationServer. Do not specify any backup instance, to ensure that all instances are independent from each other.
3. Create a Host object for the machine associated with the External Configuration Server Proxy, and set its LCA port to `0` (zero).
4. Configure another Application object of type ConfigurationServer, to create an External Configuration Server Proxy that represents the F5 load balancer. Set its host and port to the values for F5.
5. Provision all client applications with a connection to the External Configuration Server Proxy. If required, configure ADDP on those connections.
6. In the **[csproxy]** section of each Configuration Server Proxy in the proxy pool, set **proxy-cluster-name** to the name of the External Configuration Server Proxy object. For more information about this option, refer to the *Framework Configuration Options Reference Manual*.
7. If you are planning to use Kerberos authentication, do the following:
   - Configure all Configuration Server Proxies in the pool to use the same SPN and the same .keytab file.
   - Set the **KRB5RCACHETYPE** environment variable to `none`.
8. Set each of the proxy servers in the pool to autorestart, to enable Solution Control Server (SCS) to detect application failure and/or host unavailability. Configure any other monitoring features, such as hangup detection, as required. The External Configuration Server Proxy object, representing the F5 load balancer, is not monitored by SCS.

TLS Configuration

To configure TLS between agent-facing Applications and Configuration Server Proxy clusters using the F5 load balancer, do the following:

1. Obtain Certification Authority (CA) security certificates for each Configuration Server Proxy host and agent-facing client host. Store the certificates in the **Trusted Root Certification Authorities Certificates** folder. Refer to the Microsoft article Installing a Root Certificate.
2. Request and obtain security certificates for Server authentication. Make sure that the name in the **Subject** field of the certificates matches the Fully Qualified Domain Name (FQDN) of the F5 host name registered in DNS. The certificate must also have a private key that corresponds to that certificate. Host names are case-sensitive and must match DNS and Active Directory records. Refer to the Microsoft article Obtain a Certificate and to the *Genesys Security Deployment Guide*.
3. To enable key archival and recovery, set the following in the certificate template and on the CA:
   - The specific certificate template must be configured to allow key archival.
   - At least one key recovery agent must be identified on the CA, and a key recovery agent certificate must be issued to that agent.
   - Key archival must be configured on the CA.
4. Import the F5 host certificate to each host running Configuration Proxy Servers, storing the certificate in the Personal Certificates folder of the Computer account. Refer to the Microsoft article Import a Certificate.
5. On each Configuration Server Proxy, set the Listening Mode of the ports used for TLS communications to **Auto-detect** or **Secure** and attach the F5 host certificate. Refer to the *Genesys Security Deployment Guide*.

F5 Configuration

To ensure that replies from servers always traverse the load balancer on the way back to the client, SNAT (Secure Network Address Translation) is used. One of the most popular SNAT modes is the automap feature that allows mapping of all original client IP addresses to the self address of the F5 unit. The SNAT pool allows mapping of all the original client IP addresses to the IP addresses of the SNAT pool.

SNAT with a single IP address has a limit of 65535 ports. The SNAT connections might fail if a large number of client requests are traversing the SNAT. To mitigate port collisions, create SNAT pools or use SNAT automap with an appropriate number of self IP addresses on the Virtual LAN to support the expected level of concurrent connections using SNAT.

The following sample configuration is for a deployment where two IP addresses are used for the pool. In the sample, the following placeholders are used:

	<vsCSP IP>	Virtual Server of Configuration Server Proxies
	<node1 IP address>	Host 1 IP address
	<node2 IP address>	Host 2 IP address
	<F5 IP address>	F5 box IP address
	<default GW IP>	Default GateWay IP address

```
vlan vlanPerfExternal {
   tag 4094
   interfaces 1.2
}
self <F5 IP address> {
   netmask 255.255.255.0
   vlan vlanPerfExternal
   allow default
}
route default inet {
   gateway <default GW IP>
}
monitor TCP-9070 {
   defaults from tcp
   dest *:9070
}
monitor TCP-9075 {
   defaults from tcp
   dest *:9075
}
profile tcp tcp-idle600 {
   defaults from tcp
   idle timeout 600
}
node <node1 IP address> {
   monitor icmp
   screen MFfirstNode
}
node <node2 IP address> {
   monitor icmp
   screen MFsecondNode
}
pool poolCSP01 {
   monitor all TCP-9070 and TCP-9075
```

```
   members
      <node1 IP address>:9070
         monitor TCP-9070
      <node1 IP address>:9075
         monitor TCP-9075
      <node2 IP address>:9070
         monitor TCP-9070
      <node2 IP address>:9075
         monitor TCP-9075
}
virtual vsCSP {
   snat automap
   pool poolCSP01
   destination <vsCSP IP>:9070
   ip protocol tcp
   profiles tcp-idle600 {}
}
```

Business Continuity

Genesys Workspace Desktop Edition integrates load-balanced Configuration Server Proxies into a Business Continuity solution, by keeping a pool of proxy servers at each Site (active and stand-by) of the configuration. In this case, a separate application and host object that represent the F5 load-balancer at each site must be created. Refer to Genesys Workspace Desktop Edition documentation for more information about how to set up Business Continuity for Agent Desktop when using Configuration Server Proxy objects from preferred and backup sites.

Support for Multi-Language Environments

You do not need to perform any additional configuration to have Configuration Server Proxy support multi-language environments. If the master Configuration Server supports UTF-8 encoded data, all Configuration Server Proxies connected to that master Configuration Server also support UTF-8 encoding. See Multi-language Environments for more information about using UTF-8 encoding to enable multi-language environments.

Configuration Server Proxy and Configuration History Log

You can configure a history log with Configuration Server Proxy to store historical information about client sessions and changes to configuration objects. Refer to Configuration History Log for more information.

Failure of Configuration Server Proxy

When Configuration Server Proxy fails or disconnects from its clients, the clients attempt to reconnect to Configuration Server Proxy. If it is not available and if a backup Configuration Server Proxy is configured, the clients attempt to connect to the backup.

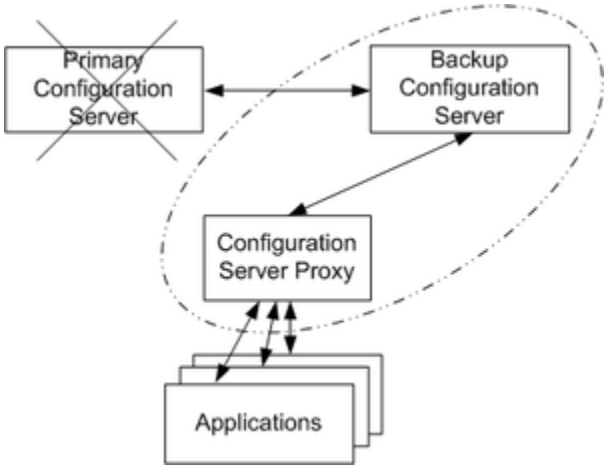
When Configuration Server Proxy fails, you must restart it manually or use the Management Layer for autorestart.

Failure of Master Configuration Server

When the master Configuration Server fails or the connection to it is lost, the clients of Configuration Server Proxy continue their normal operations. Configuration Server Proxy initiates reconnection attempts to the master Configuration Server. Meanwhile, Configuration Server Proxy responds to client requests using the configuration data stored in its memory.

When the master Configuration Server fails, you must restart it manually or use the Management Layer for autorestart.

The following diagram shows Configuration Server Proxy behavior when a primary-backup pair of master Configuration Servers is configured.



Distributed Installation

When the primary master Configuration Server fails or the connection to it is lost, Configuration Server Proxy tries to reconnect to the master Configuration Server and, if it is not available, to the backup Configuration Server. If the connection to the backup Configuration Server is established, Configuration Server Proxy remains connected to the backup server until:

- The connection to the backup Configuration Server is lost.
- The backup Configuration Server fails.
- Configuration Server Proxy fails or is restarted.

Configuration Server Proxy operation from persistent read-only database

Configuration Server Proxy remains operational during prolonged disconnect from master

Configuration Server. This comprises ability to restart Configuration Server Proxy using locally cached data. And, upon master Configuration Server's availability, Configuration Server Proxy can be restarted to re-read and restore full synchronization.

If the master Configuration Server is down/inaccessible during Configuration Server Proxy startup, then it CS Proxy makes a connection with DBMS and directly loads data from DB in persistent mode. Also, Configuration Server Proxy indicates in the log that it started in persistent mode. To enable this feature, Configuration Server Proxy (CSProxy) should be started with the command-line option **-proxy-persistent-mode**. The related options are:

- db-persistent-failover-tmout
- -cs-persistent-failover-tmout

To support this feature in Configuration Server Proxy’s host, DBMS client must be installed. Also, you must provide the **confserv.cfg/conf** file under Configuration Server Proxy working directory, as with master Configuration Server.

> ## Important
>
> Any instance of Configuration Server proxy that activated persistent mode will start as primary proxy and allow client connections, will ignore SCS requests except for stop and will periodically log out message indicating persistent mode is activated. Because proxy in this mode won't have connection to master Configuration Server, it cannot get aware of master Configuration Server going back online and require restart to re-syncornize data after master server is fully recovered.

Local VAG calculation on selected Configuration Server Proxies

This feature enables Virtual Agent Group (VAG) processing on the Configuration Server proxy, connected to the master Configuration Server with VAG processing disabled.

Overview

Default behavior of configuration layer is to calculate VAGs by the master Configuration Server and other Configuration Server clients, including proxies, see VAGs as regular agent groups. VAG membership is calculated upon initial data load or runtime reload. Upon runtime, certain changes in the configuration cause Configuration Server to recalculate VAG membership and notify clients of the VAG changes the same way as it does for regular agent groups. With majority of Genesys applications not depending on VAG notifications made by configuration layer, it is possible to disable such calculation on the master Configuration Server to rely on local calculation. Having VAGs calculated by configuration layer creates additional load on the master Configuration Server and generates extra notifications, which could be unneeded by the clients, not interested in the VAG content or performing VAG calculations on their own.

Proxy-side VAG processing allows to optimize VAG processing by running master Configuration Server with VAG processing disabled. VAGs in this case can be processed on specially configured Configuration Server proxies instead. In such configurations, clients, which do rely on Configuration Server VAG calculations, can be connected to the configuration through such proxies, while the other clients may connect to the master or regular proxies.

VAGs in this case remain empty on the master and regular proxies. VAG-processing proxies do VAG calculations upon startup / data reload and recalculate VAGs at runtime, generating additional change notifications with the events related to the VAG changes. They use the same algorithm of VAG calculation as the regularly configured master.

Limitations

This feature is only intended to support Genesys historical reporting solution with required information about VAGs membership. VAG calculating prox(ies) shall only be deployed to support Interaction Concentrator(s) ability to store correct VAG history.

Interaction Concentrator may not be able to automatically receive all VAG changes and require manual synchronization for them to appear in its database, if it was down for an extended period of time, before reconnecting back to the Configuration Server proxy responsible for VAG calculations for ICON. To detect that you have to monitor respective Configuration Server proxy for events that indicate history log transactions cannot be replied to a newly connected client and, when detected, initiate resync on ICON side by using the **start-cfg-sync** option. See Interaction Concentrator documentation for more details.

Feature configuration

The **force-vag-calculation** option is set in the **csproxy** section in the csproxy application object. The configuration option **csproxy/force-vag-calculation** enables VAG calculations on a Configuration Server proxy connected to the master Configuration Server with disabled VAG processing. Refer to Framework Configuration Options Reference Manual for information on **csproxy/force-vag-calculation**.

HA considerations

Both primary and backup proxy instances, configured for proxy-side VAG processing, perform VAG processing in parallel. In case of switch-over of the proxies, the clients are expected to be able to successfully restore session to the backup after loosing connection to the primary. This is an ideal “all-synced” case.

Warning

Utilizing this feature requires special attention to the configuration.

1. Both primary and backup master Configuration Server should be configured with **[cs_main_section]\disable-vag-calculation**=`true` only in the **.conf/.cfg** file of the primary and backup master Configuration Server.
2. Regular proxies (both primary and backup), which are not supposed to calculate VAGs, should not have the **force-vag-calculation** option, or set it to `false`.
3. Special VAG-processing proxies, both primary and backup instances, should be configured with **[csproxy]\force-vag-calculation**=`true`.

If the above configuration changes, all affected instances of both master and proxies should undergo hard restart (not switchovers).

Steps to roll out the solution based on proxy-side VAG processing

Starting from traditional configuration with master-side vag handling.

1. The first and the most important step is to decide:
    1. which clients are to be connected directly to the master
    2. which go through regular proxies
    3. which will use the proxy(ies) configured for VAG processing. It should be clearly understood that after this reconfiguration clients, 1) and 2) will no longer receive VAG membership information. They will see VAGs as having no members, although VAG scripts and the rest on VAG object content will still be available to them. For these reasons, clients which are not interested in VAG membership or

capable to perform VAG calculations on their own may remain connected to the master or through regular proxies. Clients which do rely on Configuration Server to calculate VAGs should be reconnected trough the VAG calculation proxies.

2. Proxies, which you plan to use for VAG calculation should be upgraded to the version that supports the feature. Upgrade of the master and other proxies is not mandatory.
3. Set **[cs_main_section]\disable-vag-calculation**=`true` for both primary and backup instances of the master Configuration Server in their **.conf/.cfg** files. Set **[csproxy]\force-vag-calculation**=`true` for both primary and backup instances of the proxies, you plan to use as VAG-processing, in their Application configuration objects. No changes in the configuration of the regular proxies are needed.
4. Perform had restart of both instances of the master Configuration Server. Switchover is not enough: both primary and backup should be stopped concurrently first, then started again. Restart of the proxies is not required. Proxies, configured with **force-vag-calculation**=`true`, once reconnected to restated master with **disable-vag-calculation**=`true`, will automatically enable VAG calculation on their side. The remaining proxies after reconnect and data reload will see the content of VAGs as empty.
5. Reconfigure the clients you planned to use with VAG-processing proxies, so they connect to these proxies. Then restart them to connect to these proxies. Whether hard restart of the client HA pairs is needed depends on specific client. However, hard restart is always the safe choice.
6. To double-check the configuration, connect GA or CM to both master and VAG-processing prox(ies) to verify that VAGs on the master have no members, while VAGs on the VAG-processing prox(ies) have their member lists calculated as expected according to the VAG scripts.

Distributed Solution Control Servers

Multiple Solution Control Servers operating in Distributed mode (referred to as *Distributed Solution Control Servers*) distribute management-related tasks among the sites in a distributed enterprise that uses a single Genesys Configuration Database.

What are Distributed Solution Control Servers?

In these installations, each SCS controls its own subset (defined by you) of the Hosts, Applications, and Solutions, and communicates with the others through a dedicated Message Server.

Specifically, a Distributed Solution Control Server performs the following functions:

- Performs the same functions of monitoring, control, alarm detection, and alarm processing as the SCS in non-Distributed mode, but on a subset of Hosts, Applications, and Solutions explicitly assigned to this SCS in the Configuration Database.
- Communicates all the updates to statuses of the assigned objects to other Distributed Solution Control Servers, using a dedicated Message Server.
- Receives notifications about updates to the status of non-assigned objects (that is, objects assigned to other Solution Control Servers) from Message Server.
- When receiving a control command on an object not assigned to this SCS, forwards this command via Message Server to the appropriate SCS.

Because Distributed Solution Control Servers communicate with each other, they all have the same information about all hosts, applications, and solutions. Thus, you can connect the interface object associated with Genesys Administrator to any Distributed SCS and monitor and control the whole environment as a single entity (given appropriate permissions). When a Distributed SCS receives a control command for an object that this SCS does not control, it forwards this command to the appropriate SCS and passes any further notifications back to the requestor.

Using Distributed Solution Control Servers helps you resolve some problems common to distributed installations:

- It eliminates false switchovers that occur when SCS disconnects from LCA at a remote site because of the slow network connection between sites or because of temporary network problems.
- It prevents a single point of failure. A failure of one Distributed SCS only means a temporary loss of control over a subset of Hosts, Applications, and Solutions; other Distributed Solution Control Servers continue to control the rest of the environment.

Limitation

In distributed mode, Solution Control Server reports host statistics (such as CPU, User Time, Kernel Time, and Non-IDLE Time) for only the host it controls when Genesys Administrator or Genesys Administrator Extension (GAX) is connected to SCS. It does not provide host information for other hosts that are controlled by other Solution Control Servers in distributed mode.

For example, assume the following distributed SCS environment:

host_a
host_b
SCS_a is located on, and controls, host_a
SCS_b is located on, and controls, host_b
GAX can be connected to either SCS_a or SCS_b

When GAX is connected to SCS_a, it reports host statistics for only host_a.
When GAX is connected to SCS_b, it reports host statistics for only host_b.

Deploying Distributed Solution Control Servers

Warning

- Do not use Solution Control Servers in Distributed and non-Distributed modes simultaneously within the same Configuration environment. If you plan to use Distributed SCS in your installation, turn on Distributed mode for all Solution Control Servers you install.
- When using Distributed Solution Control Servers, always ensure that each Solution Control Server, either by itself or as part of a high-availability pair, is running on the host which it controls. Failure to do so can, in some cases, result in unpredictable behavior of the Solution Control Servers in the Distributed configuration. For example, different Solution Control Servers may start competing for control over Applications on the Host.
- When working with applications in HA pairs, the same distributed SCS must be configured to control both applications working in each HA pair.

1. Configure Distributed Solution Control Servers in Distributed mode.

a. Configure as many Solution Control Server Application objects as necessary, as described in Solution Control Server.

b. Turn on Distributed mode for each Solution Control Server Application object, by setting the following configuration options in the **[general]** section:

   - **distributed_mode=**on
   - **distributed_rights=**default

c. If you are planning to leave any of the Host, Application, or Solution objects unassigned—that is, without specifying which SCS is to control them—dedicate one SCS to the control of all unassigned hosts, applications, and solutions. To instruct one SCS to work in this mode, set the following values for configuration options in the **[general]** section for that particular SCS application:

   - **distributed_mode=**on

- **distributed_rights=**`main`

> **Important**
>
> Only one of the Distributed Solution Control Servers can have the value `main` for the **distributed_rights** configuration option.

2. Divide your configuration environment between the Solution Control Servers.

When you are using Distributed Solution Control Servers, you must explicitly configure the servers' ownership of Hosts, Applications, and Solutions. That is, you must associate each Host, Application, and Solution object with a particular SCS by changing the object's properties:

> **Important**
>
> To distribute control over the primary and backup servers in a redundant pair between different Distributed Solution Control Servers, all Solution Control Servers in the configuration must be running release 7.6 or later.

Recommendations

- Do not distribute control over the primary and backup servers in a redundant pair between different Distributed Solution Control Servers if any SCS in the configuration environment is running a pre-7.5 release. Genesys recommends that you configure the same SCS to control both the primary and backup servers in a redundant pair.
- When you are distributing control over the configuration objects among Distributed Solution Control Servers, ensure that the same SCS that controls a solution also controls all applications included in this solution. Although one SCS can technically control a solution while other servers control applications included in that solution, avoiding this configuration helps minimize network traffic between Solution Control Servers.
- Genesys strongly recommends that you not assign each component in an HA pair to different Solution Control Servers in a distributed environment. In this configuration, the functionality of each Solution Control Server in the HA pair might be limited to handling simple application failures only (the failure of an application within the pair). In addition, the state of each component in the monitored HA pair might become inconsistent if network failures occur between the Distributed Solution Control Servers.

Assigning a Distributed Solution Control Server

- To control a Host: In the Host object, specify the SCS Application in the **Solution Control Server** field in the **General** section of the **Configuration** tab.
- To control an Application: Do not make any changes to the Application object. Specifying SCS ownership of the Application's Host is enough. The Distributed SCS automatically controls any Applications assigned to the Host this SCS controls.
- To control a Solution: In the Solution object, specify the SCS Application in the **Solution Control Server** field in the **General** section of the **Configuration** tab.

3. Configure a dedicated Message Server through which the Distributed Solution Control Servers will communicate with each other.

Recommendations

Distributed Solution Control Servers communicate with each other through Message Server. Genesys recommends that you use a dedicated Message Server for this purpose.

Prerequisites

- An Application object exists for each Distributed SCS in the configuration environment.
- You are logged in to Genesys Administrator.

Configuring a Dedicated Message Server

a. Configure a Message Server Application object with appropriate configuration parameters. Refer to Message Server.

b. Double-click the Message Server Application object, and click the **Options** tab.

c. Create a new configuration options section called **[MessageServer]**.

d. In this section, create a new configuration option called **signature** and set its value to `scs_distributed`. Each Distributed SCS will process this option to determine which of the Message Servers specified in its **Connections** to use for communications with other Solution Control Servers.

e. In the Application object for each Distributed Solution Control Server, add a connection to this Message Server, as follows:

1. Enter `ADDP` as the **Connection protocol**.
2. Set the **ADDP Local Timeout** and **Remote Timeout** to values that are less than half the minimum **alive_timeout** values of all Distributed Solution Control Servers in the configuration environment.
   In other words:
   $T_{addp} < T_{scs} * 0.5$
   where:
   T_{addp} = ADDP timeout
   T_{scs} = minimum `alive_timeout` of all Distributed Solution Control Servers
   Refer to the *Framework Configuration Options Reference Manual* for a detailed description of the **alive_timeout** option.

4. (Optional) Configure a Message Server for centralized logging at each site with Distributed Solution Control Servers.

For distributed environments using a single Configuration Database, Genesys recommends using a dedicated Message Server for centralized logging at each site. In most cases, you have to configure as many Message Servers as there are Distributed Solution Control Servers.

> **Important**
> You can configure as many Message Servers for centralized logging as you need per site. These are in addition to the Message Server dedicated to handle communications between the distributed servers.

After you have installed the Message Servers, you should verify that each Message Server used for centralized logging is configured and connected to a Solution Control Server and to each of the applications controlled by that Solution Control Server as follows:

Prerequisites

- Distributed Solution Control Servers are set up in the configuration environment.

- The Message Server used for centralized logging in this environment is installed.
- You are logged in to Genesys Administrator.

Verifying Configuration of Message Servers used for Centralized Logging

a. Go to **Provisioning > Applications**, and double-click a Solution Control Server Application object to open its Configuration tab.

b. In the **General** section, make sure that a connection to the Message Server that is providing the centralized logging is added to the list of **Connections**.

c. For each Application object that this particular Solution Control Server controls:

   i. Open the **Configuration** tab of the Application object.

   ii. In the **General** section, make sure that a connection to that same Message Server is added to the list of **Connections**.

5. (Optional) Configure redundant Application objects for Distributed Solution Control Servers.

Distributed Solution Control Servers support the Warm Standby redundant configuration in the same way as other Genesys servers, with the added benefit that the backup maintains data synchronization with the primary. That is, you can configure a primary and a backup pair of Distributed Solution Control Servers to operate with Warm Standby redundancy. Refer to Redundant Solution Control Servers for more information.

6. After you are finished with the configuration tasks, physically install all instances of Solution Control Server and Message Server to match the configuration.

Starting Distributed Solution Control Servers

> **Important**
>
> Starting a Solution Control Server in Distributed mode requires a special license. Refer to the *Genesys Licensing Guide* for more information.

Start each Distributed Solution Control Server in the same way as you would start a non-distributed SCS. See Starting SCS and Starting a Backup SCS for more information.

Disaster Recovery / Business Continuity

This section describes a recommended architecture to ensure successful disaster recovery, or business continuity, following a scenario in which the main site was rendered inoperable because of some natural or other disaster.

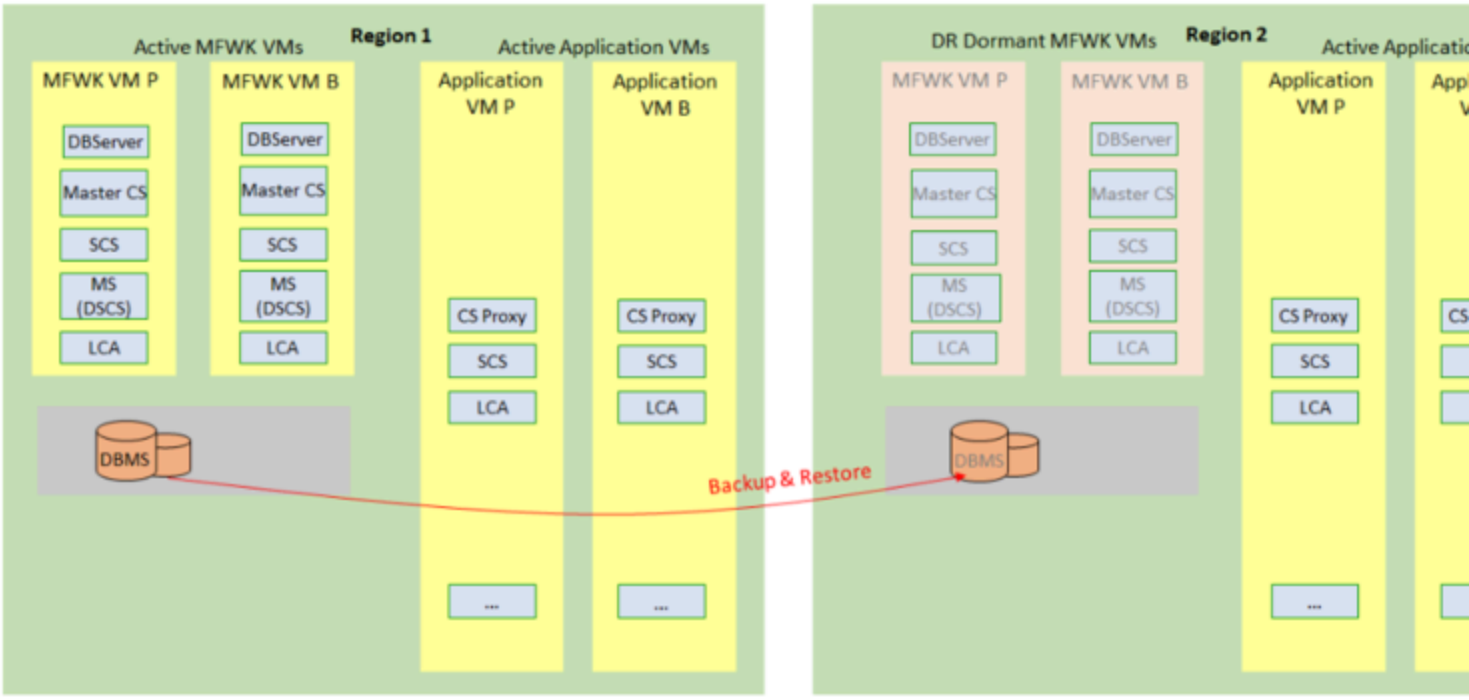
> **Warning**
> The information provided herein is a reference architecture and not a step-by-step deployment procedure. The main focus of this chapter is to highlight major requirements when deploying Genesys software in Disaster Recovery mode. Actual deployment may require additional steps and/or configuration that is beyond the scope of this document.

Overview

The Genesys system configuration is stored in a single database, and can be accessed by only one primary master Configuration Server connection at a time. The Configuration Database is constantly modified by Configuration Server clients, and is archived periodically to prevent the loss of data. However, database maintenance and periodic backup can cause significant downtime. It cannot prevent partial or whole loss of configuration data if a major disaster occurs, such as one in which the Configuration Database and all updates and modifications made since the last backup is completely lost. To improve the robustness of the Management Framework solution and to reduce downtime for system maintenance, this architecture replicates a secondary standby database.

Architecture

The following diagram illustrates the disaster recovery architecture for a multi-site configuration operating under normal conditions.



MFWK – Management FrameWorK
Master CS – Master Configuration Server
SCS – Genesys Solution Control Server
LCA – Genesys Local Control Agent
MS(DSCS) – Genesys Message Server for Distributed SCS
CS Proxy – Genesys Configuration Server Proxy
RDS Multi Zone – Relational Database Server configured for multi zone operation
DBMS – DataBase Management System (MSSQL/Postgres/Oracle)
MFWK VM P – Management Framework Virtual Machine Primary
MFWK VM B – Management Framework Virtual Machine Backup
Application VM P – Application Virtual Machine Primary
Application VM B – Application Virtual Machine Backup

- Failed part
- Dormant MFWK Instanc
- Active Instances
- Genesys Component

Components

Important

Genesys recommends deployment of components on Virtual Machines (VMs). It is recommended that Framework VMs be cloned, to have identical copies across sites, and then make targeted adjustments.

Framework components for all Framework VMs should be set up identically in this architecture.

Framework components on Application VMs are set up as required to provide service in each region, and include other details listed in this section and keeping in mind future requirements if necessary. Application components are set up as needed, and must use Configuration Server Proxy in the given region, instead of the master Configuration Server, to obtain their configuration. There must be two FQDNs that can be resolved globally for Framework VMs. Each pair of Framework VMs (Active and Dormant) must have the same pair of FQDNs and different IP Addresses.

Region 1

- A primary DBMS, containing the Configuration Database and the optional Log Database.
- An active redundant master Configuration Server primary/backup (HA) pair in the Framework Virtual Machine (VM).
- An active DB Server HA pair in the Framework VM, which is required for Configuration Server (8.1 only) to connect to Configuration Database.
- An active Solution Control Server (SCS) in distributed mode; as the main SCS, it is required to control the main master Configuration Server pair in the Framework VM.
- An active Message Server HA pair configured in distributed mode in the Framework VM to support communication between the Solution Control Servers that are controlling components such as Configuration Server Proxy pairs, and Log Message Servers.
- A Local Control Agent in each VM.
- A Configuration Server Proxy HA pair in the Application VM and connected to the currently active primary Configuration Server.
- A Solution Control Server HA pair in distributed mode set as `default` in the Application VM, to control other components.
- (Optional) A Log Message Server HA pair in both the Framework and Application VMs for network logging, and connected to the Log Database.

Region 2

- A secondary DBMS, containing the replicated Configuration Database and the optional replicated Log Database, which gets updated by the primary instance in case of any changes.
- A dormant (non-active) redundant master Configuration Server primary/backup (HA) pair in the Framework VM.
- A dormant DB Server HA pair in the Framework VM, which is required for Configuration server (8.1 only) to connect to the Configuration Database.
- A dormant Solution Control Server (SCS) in distributed mode; as the main SCS, it is required to control the main master Configuration Server pair in Framework VM.
- A dormant Message Server HA pair configured in distributed mode in the Framework VM to support communication between Solution Control Servers controlling components, such as Configuration Server Proxy pairs and Log Message Servers.
- A Local Control Agent in each VM.
- Configuration Server Proxy HA pair in the Application VM and connected to the currently active primary Configuration Server.
- A Solution Control Server HA pair in distributed mode set as `default` in the Application VM, to control

other components.

- (Optional) A Log Message Server HA Pair in both the Framework and Application VMs for network logging, and connected to the Log Database.

Solution Control Servers

All Solution Control Servers used in this deployment are configured in Distributed SCS mode. They should all be configured in HA pairs in each Region.

In each Region, one SCS is deployed on the Framework VMs, and is dedicated to management applications, specifically Configuration Server and the dedicated Message Server for the distributed Solution Control Servers, described below.

For distributed Solution Control Servers to communicate with each other, a Message Server dedicated for use by the distributed Solution Control Servers (**[MessageServer].signature**=`scs_distributed`) is also installed in each Framework VM.

Each Region also has an SCS HA pair deployed on the Application VMs.

All Solution Control Servers in all VMs must always connect to the main Configuration Server, not to the Configuration Server Proxies. Solution Control Servers must be provisioned to start using a configuration file (use the `-f` command line option) that points to the FQDNs of master Configuration Servers.

Depending on the number of applications, it is possible to deploy additional distributed Solution Control Servers for load balancing.

Message Servers

An HA pair of Message Servers is dedicated for communications between the distributed Solution Control Servers deployed in the Framework VM.

Optionally, each Region can have its own instance of a Log Message Server to be used for network logging by applications running at the same site. One pair of Message Servers is installed on the Framework VM, and handles logging for all components in this VM. Likewise, a Message Server pair is also installed on the Application VM, is managed by the SCS in that VM, and handles logging for all components in this region. If a Log Message Server is configured, a dedicated Log Database is required at each site, one as active primary and the other as dormant replicated secondary.

Configuration Server Proxies

In each region, a Configuration Server Proxy HA pair is deployed in one pair of Application VMs that connects to the active pair of Master Configuration Server.

All applications deployed in all Application VMs, except SCS, must connect only to the Configuration Server Proxy in the corresponding region for any read/write operation.

DB Server

DB Server HA pairs should be deployed in the Framework VMs to connect with the corresponding

database, but only if required by the master Configuration Server.

Replicating DBMS

The DB Server must be replicated in Region 2 for the Configuration and Log Databases. This is required so that, after failover, the secondary Configuration Server and Log Message Servers can connect with the replicated DBMS and continue working.

If a major disaster occurs, the secondary database can be accessed by a secondary master Configuration Server that is brought online from the dormant state, and changing the IP address name resolution for Configuration Server Proxies to the host running that secondary master Configuration Server. Operations at sites will continue uninterrupted, but in limited mode, without a configuration change until the secondary master Configuration Server is brought online and restored to normal mode after the proxy servers reconnect to the secondary master Configuration Server.

Block any accidental connection between the dormant DBMS VM and either active and dormant Framework VMs by running the `iptables` command (on Linux) or creating firewall rules (on Windows) on the dormant DBMS host.

To block an active Configuration Server from accessing a dormant DBMS:

- On Linux:

```
sudo iptables -A INPUT -p tcp --dport <DBMS PORT> --src <Framework VM IP>/24   -j
REJECT
```

- On Windows, create firewall rules that block any connection to a DBMS port from the Framework VMs.

After the replicated dormant DBMS become active, and after the previously active DBMS instance is stopped or failover has occurred, use the `iptables` command or remove the firewall rules to restore the connection with the Framework VMs.

Use the same `iptables`/firewall rules to block connections for the currently dormant DBMS VM. To restore access:

- On Linux:

```
sudo iptables -D INPUT -p tcp --dport <DBMS PORT>…
```

- On Windows, remove the firewall rules for Framework VMs.

For more information about replicating Framework databases for Disaster Recovery using MS SQL Server and Oracle, refer to the following:

- Framework Database Replication for Disaster Recovery Using MS SQL Cluster with AlwaysOn
- Framework Database Replication for Disaster Recovery Oracle GoldenGate

Deploying Genesys Components

Deploy Genesys components as follows:

1. Using the initialization scripts in the Installation Package, create the database objects for the Configuration Database and if you want, the Log Message Server Database.
2. In one Region, deploy the initial set of Framework VMs, including the following components in HA pairs, as services to be started when the Framework VMs start:
    - Master Configuration Server
    - DB Server (if required for the selected master Configuration Server deployment mode)
    - Local Control Agent
    - Main Solution Control Server
3. On active Framework machines, deploy an HA pair of Message Servers. In the Annex of the Message Server objects, set the **[sml].autostart** option to `true`.
4. In both Regions (1 and 2), deploy the following components in HA pairs as services to be started when the Application VMs start:
    - Configuration Server Proxy
    - Local Control Agent
    - Solution Control Server
5. If required, deploy Log Message Server HA pairs as services in the Application VMs. Set the **[sml].autostart** option to `true` in the Annex of the Application objects, so the application will be started automatically by Management Layer when the system is started.
6. Clone Framework VMs to another site and start them as not connected, using a firewall or the `iptables` command to make sure that no connections can be made from components running on the closed machines to anywhere within the environment. Adjust the local VM configuration files as needed (typically, you have to point to Region-specific DBMS endpoints), and then stop the cloned VMs.

DNS-based Disaster Recovery

Both Full and Partial Failovers can be done with the help of DNS.

The DNS Server configures a record type (call it type *A* for purposes of this discussion) to resolve the IP address of the host running the live master Configuration Server HA pair. It resolves the IP address to the main host in normal mode, and to the secondary host in failover mode.

To avoid false situations, name resolution from this record of type A in the **/etc/hosts** file on the main and secondary hosts points to the IP Address of the local host.

Within each set of Framework VMs, all fully-qualified domain names (FQDNs) of Framework components are resolved using the **hosts** file in IPs on these VMs. This is required because the DNS service resolves Management Framework FQDNs to VMs running in the active Region and currently serves other VMs of this environment.

The DNS TTL (Time To Leave) on type A records for Framework FQDNs must be set according to the

expected period of time after which components running on Application VMs are expected to automatically reconnect to newly introduced instances of Framework VMs during the Disaster Recovery event.

Operation

1. Start the Framework VM pair in Region 2.
2. Run the necessary procedure to switch the cfgmaster host name IP resolution to a MAIN live system.
3. On the host running Configuration Server Proxies, run the necessary procedure to clean out the DNS cache.

Failover Scenarios

This section describes the various failover scenarios that are handled by this architecture.

Full Failover

Region 1, hosting the currently active Framework components, goes down completely so no VM in this region is up and running. The Region goes through Full Failover.

The DBMS in Region 2 becomes active after the previously active DBMS is stopped or is going through failover. Set the iptables command or firewall rules, as described in Replicating DBMS.

In Region 2, restart the Framework VMs to bring the dormant Framework components back into service. These reactivated components will connect to the corresponding databases in the currently active DBMS (in Region 2).

Change the IP address name resolution for Configuration Server Proxies in all Application VMs to the host where the secondary master Configuration Server is running.

Partial Failover

This scenario is the case in which only one (not both) of the Framework VMs or the DBMS VM, which are currently active in Region 1, go down.

Framework VMs Failover

In Region 2, restart the Framework VMs to bring the dormant Framework components back into service. These reactivated components will connect to the corresponding databases in the currently active DBMS (in Region 2).

Change the IP address name resolution for Configuration Server Proxies in all Application VMs to the host where the secondary master Configuration Server is running.

DBMS Failover

The DBMS in Region 2 becomes active after the previously active DBMS is stopped or is going through failover. Use the `iptables` command or firewall rules, as described in Replicating DBMS.

Active Framework components will connect to the corresponding databases in the currently active DBMS (in Region 2).

Starting and Stopping Framework Components

You can start and stop a Framework component in any of the following ways:

- Use the startup file created by the installation script. This file can only be used to start the component, you must use one of the other ways to stop the component.
- Use the Management Layer.
- Manually, specify command-line parameters.
- Use the Windows Services Manager, available only in Windows.

> **Warning**
> If you are using SNMP, you must start or restart SCS immediately after the SNMP <license option> has been (re)activated. Otherwise, you will lose all SNMP functionality until SCS has been (re)started.

Using Startup Files

Startup files are files named **run.sh** (on UNIX) or **startServer.bat** (on Windows), and that installation scripts create and place in the applications' directories during installation. For additional information about how to use startup files, refer to the *Framework Management Layer User's Guide*.

> **Important**
>
> You must manually modify the `run.sh` file created for a redundant server before you can use it to start the server. Refer to Configuring Redundant Components for more information.

Prerequisites

- The startup parameters in the startup file are correct.
- The required Applications that should be running for this Application to start are installed and running. See the appropriate sections in Starting Components to identify which Applications should be running for a particular Application to start.

On UNIX

To start the Application on UNIX, go to the directory in which the application is installed and type the following on the command line:

```
sh run.sh
```

On Windows

To start the application on Windows, do one of the following:

- Go to the directory in which the application is installed and double-click the following:
  `startServer.bat`
- From the MS-DOS window, go to the directory in which the application is installed and type the following on the command line:
  `startServer.bat`

Using the Management Layer

You can use Genesys Administrator to start and stop Applications via the Management Layer.

> **Important**
>
> To operate with the Management Layer, Genesys Administrator must be configured as described in the *Genesys Administrator Deployment Guide*.

Before starting an Application with the Management Layer, make sure the Application's startup parameters are correctly specified in the Application properties. In the **Server Info** section of the Application's **Configuration** tab, check that the following entries are correct:

- **Working Directory**—Directory in which the application is installed and/or is to run
- **Command Line**—Name of the executable file
- **Command Line Arguments**—Command-line parameters

See Command-line parameters for descriptions of the parameters for Framework components.

After you correctly specify the command-line parameters, you can start and stop the following Framework components from Genesys Administrator:

- Configuration Server (the Command Line Arguments are not required for the primary Configuration Server)

> **Important**
>
> For the Management Layer to start Configuration Server, you must modify the Configuration Server application.

- Configuration Server Proxy
- Message Server
- SNMP Master Agent
- T-Server
- HA Proxy
- Stat Server

The Management Layer can also restart failed applications; to enable the autorestart functionality for a particular application, select the corresponding check box in the properties of the Application.

When an application is started (or restarted) via the Management Layer, it inherits environment

variables from LCA, which executes the startup command. Therefore, you must also set the environment variables required for the application for the account that runs LCA.

A limitation of the Linux operating system, however, does not allow LCA to take the default environment variables of the user, or system, if LCA is started as a service. As a result, LCA is unable to pass them on to the application. If you are using Linux, and an application requires a user's environment variables, you can do any one of the following to bypass this limitation:

- Set the environment variable in the startup file of the respective application.
- Start the LCA process manually and not as a service.
- Add the environment variable to the LCA service file.

> **Warning**
> Stopping an application via the Management Layer is not considered an application failure. Therefore, the Management Layer does not restart applications that it has stopped unless you have configured an appropriate alarm condition and alarm reaction for them.

Stop vs. Graceful Shutdown

When you stop an Application or a Solution, it shuts down, ceasing all processing immediately. This may have a detrimental effect on the rest of the system.

Starting in release 8.0, you can stop an Application or a Solution gracefully, known as a *graceful shutdown* or *graceful stop*. Applications refuse any new requests, but continue to process their current requests. A Solution gracefully shuts down all of its composite Applications, then stops.

> **Important**
> Because a number of Solutions can share the same Applications, some solution components may continue to have a status of Started after you stop the Solution.

Only Applications and Solutions that support the graceful stop functionality can be stopped gracefully. Applications and Solutions that do not support this functionality shut down ungracefully.

If you are unsure if the application supports graceful shutdown, you can use the **suspending-wait-timeout** configuration option to configure a timeout. If the status of the Application changes to Suspending within this time, the Application supports graceful shutdown. If the status does not change to Suspending within the timeout, the Application does not support graceful shutdown, and the Application will then stop ungracefully after the timeout expires. Refer to the *Framework Configuration Options Reference Manual*, for a detailed description of this configuration option and how to use it.

Refer to *Genesys Administrator Help* for more information about stopping gracefully, and about

configuring a timeout.

Starting Manually

When using a manual procedure to start an application, specify the startup parameters in the command prompt. In the command prompt, command-line parameters must follow the name of the executable file. On the **Shortcut** tab of the **Program Properties** dialog box, command-line parameters must also follow the name of the executable file.

Some Genesys interface components also require that you log in to them using preassigned login credentials. Use the procedure Logging In.

Starting in release 8.5.1, all Framework servers, except DB Server and Genesys SNMP Master Agent, support *bootstrap logging*, that is, generating logs when the server is started from the command line. These logs, called *bootstrap logs*, are generated from the very beginning, when the command line (where the log arguments are set) triggers the startup of the server, until the server loads its log configuration information from the Configuration Database. These logs are generated using the command-line parameters **-log-<log option name> <log-type>** and optionally **-log-<log-type> <filename>**, which are described in the supporting component-specific sections below.

Common Command Line Parameters

The following table lists command-line parameters that are common to all Framework components:

-host	The name of the host on which Configuration Server is running.
-port	The communication port that client applications must use to connect to Configuration Server.
-app	The exact name of an application as configured in the Configuration Database.
-l	The license address. Use for the server applications that check out technical licenses. Can be either of the following: • Full path to, and exact name of, license file used by an application. For example, **-l /opt/mlink/license/license.dat** • Host name and port of license server, as specified in SERVER line of license file, in **port@host** format. For example, **-l 7260@ctiserver**.
-v	The version of a Framework component. This parameter does not start an application, but returns its version number instead. Either uppercase (**V**) or lowercase (**v**) letter can be used.
-nco [X/Y]	The Nonstop Operation feature is activated; X exceptions occurring within Y seconds do not cause an application to exit. If the specified number of exceptions is exceeded within the specified number of seconds, the application exits or, if so configured, the Management Layer restarts the application. If you do not specify a value for the **-nco** parameter, the default value (6 exceptions handled in 10 seconds) applies. To disable the Nonstop Operation feature, specify **-nco** 0 when starting the application.
-lmspath	The full path to the log messages files (the common file named **common.lms** and the application-specific file with the extension ***.lms**) that an application uses to generate log

events. This parameter is used when the common and application-specific log message files are located in a directory other than the application's working directory, for example, when the application's working directory differs from the directory to which the application is originally installed. Note that if the full path to the executable file is specified in the startup command line (for instance, **c:\gcti\multiserver.exe**), the path specified for the executable file is used for locating the ***.lms** files, and the value of the **-lmspath** parameter is ignored.

> Warning
>
> An application that does not find its ***.lms** file at startup cannot generate application-specific log events and send them to Message Server.

Starting Components

> **Important**
>
> When an application is installed on a UNIX operating system and the application name, as configured in the Configuration Database, contains spaces (for example, **My T-Server**), you must surround the application name by quotation marks (**" "**) in the command line, as follows:
>
> **-app "My T-Server"**
>
> Specify the rest of the command-line parameters as for any other application.

This section contains prerequisites, procedures, and other information about starting each Framework component, as follows:

- Configuration Server
- Configuration Server Proxy
- Local Control Agent
- Message Server
- Solution Control Server
- SNMP Master Agent

Prerequisites for starting other Framework components are also provided, as follows:

- License Manager
- Genesys Administrator
- HA Proxy
- T-Server
- Stat Server

Configuration Server

Prerequisite

- FlexNet Publisher License Manager is installed and running.

Configuration Server does not require any of the common command-line parameters for startup.

You can use the following command-line parameters when starting Configuration Server:

-log-<log option name> <log-type>	
	Specifies the amount and level of bootstrap logging to occur at startup of Configuration Server, before the server retrieves its log configuration information from the Configuration Database. The log option name must be the same as is specified in the **[log]** section. All options from the **[log]** section are supported by this parameter except those related to centralized logging by Message Server. For example: `confserv -c confserv.cfg -s confserv -log-verbose all` **Important** If the value for the **verbose** option in both the Configuration Database and the Configuration Server Application object (that is, the value used for the operation logs, not the startup logs) is either not given or is invalid, the value of **<log-type>** is also used for the operational logs, but only for that session.
-log-<log-type> <filename>	
	Used with **-log-<log option name> <log-type>** to specify the name of the file that stores only the startup logs. After Configuration Server has started up and initialized, the operational logs are stored in the file specified in the Configuration Server Application object. For example: `confserv -c confserv.cfg -s confserv -log-verbose all -log-all CSstartup`
-c	Configuration Server reads its configuration settings from a configuration file with the specified name. If you set this parameter, its value overrides the default name of the configuration file (**confserv.conf** on UNIX or **confserv.cfg** on Windows).
-s	Configuration Server reads its configuration

	settings from a configuration section with the specified name. The section must be configured within Configuration Server's configuration file; the section name must be the same as the name of the Configuration Server application configured in the Configuration Database. Use this parameter to start a backup Configuration Server.
-p	Forces an instance of Configuration Server to start, encrypt the database password in the configuration file, and terminate. Refer to Encrypting the Configuration Database Password for more information about encrypting the Configuration Database password.
-keys [<encryption file name> <decryption file name>]	
	Forces Configuration Server to use an asymmetric encryption algorithm to encrypt and decrypt the Configuration Database password. If the subparameters are specified, the encrypt key is stored in the <encryption file name> and the decryption (private) key in the <decryption file name>. If not specified by this parameter, the keys are stored in the enc.pem and dec.pem in the directory in which Configuration Server is installed. Refer to Encrypting the Configuration Database Password for more information about encrypting the Configuration Database password.
-cfglib-conn-async-tmout	
	Sets a timeout (in seconds) for a client to expect a TCP success or failure response from the server to which it is connecting. If the timeout expires, all pending connection requests are cancelled, enabling the connection to be made. Without the timeout, the connecting client might eventually stop trying to reconnect, and then stop responding completely because of network issues. This timeout applies to the following connections: • Primary or backup Configuration Server Proxy connecting as a client to primary master Configuration Server • Backup master Configuration Server connecting as a client to primary master Configuration Server If no value is specified for this parameter, a default value of 20 seconds is used. Tip For the backup master Configuration Server only, this value can be set in the configuration file for the backup server, using the **cfglib-conn-async-tmout** configuration option. Refer to the *Framework Configuration Options Reference Manual* for more information about this option.

-cfglib_port	
	Configuration Server opens the listening port specified in the command line. The port is opened in unsecured mode. This port is not written to the Configuration Server Application object, and does not survive a restart of Configuration Server. Do not use this option as a part of normal startup. Use it only as a last resort when regular secure ports cannot be accessed because of a configuration problem, such as incorrect or expired security certificates, or when a duplicate port (not necessarily secure) is specified in the configuration and therefore cannot be opened.
-upgrade-mode 1	Use this command-line parameter during migration of HA Configuration Servers to force one Configuration Server to ignore its peer server in the HA pair and initialize itself as the primary Configuration Server. This parameter functions the same way as the **upgrade-mode** configuration option, but does not require a change to the configuration file. This parameter can also be used when a restart of the server is required by the database, such as to recover from corrupt data in Configuration Server memory. For more information about using this parameter, refer to Upgrading an HA Pair of Configuration Servers in the *Management Framework Migration Guide*.

On UNIX

Go to the directory in which Configuration Server is installed and do one of the following:

- To use only the required command-line parameters, type the following on the command line:
  ```
  sh run.sh
  ```
- To specify the command line yourself, or to use additional command-line parameters, type the following command on the command line:
  ```
  confserv [<additional parameters and arguments as required>]
  ```

On Windows

Do one of the following:

- Use the **Start > Programs** menu.
- To use only the required command-line parameters, go to the directory in which Configuration Server is installed, and double-click the **startServer.bat** file.
- To specify the command line yourself, or to use additional command-line parameters, open the MS-DOS window, go to the directory in which Configuration Server is installed, and type the appropriate

command on the command line:

```
confserv.exe [<additional parameters and arguments as required>]
```

Configuration Server Proxy

Configuration Server Proxy supports only the command-line parameters common to Framework server components; it does not support any of the additional command-line parameters specific to Configuration Server.

Prerequisites

- The Master Configuration Server is installed and running.
- License Manager is installed and running.

You can also use the following command-line parameters when starting Configuration Server Proxy:

-log-<log option name> <log-type>	
	Specifies the amount and level of bootstrap logging to occur at startup of Configuration Server Proxy, before the proxy server completes its connection to the master server and reads log options from the master server Application object. The log option name must be the same as is specified in the **[log]** section. All options from the **[log]** section are supported by this parameter except those related to centralized logging by Message Server. For example: `confserv -host localhost -port 2020 -app csproxy -log-verbose all` Important If the value for the **verbose** option in both the Configuration Database and the Configuration Server Proxy Application object (that is, the value used for the operation logs, not the startup logs) is either not given or is invalid, the value of **<log-type>** is also used for the operational logs, but only for that session.
-log-<log-type> <filename>	
	Used with **-log-<log option name> <log-type>** to specify the name of the file that stores only the startup logs. After Configuration Server Proxy has started up and initialized, the operational logs are stored in the file specified in the Configuration Server Proxy Application object. For example: `confserv -host localhost -port 2020 -app`

	`csproxy -log-verbose all -log-all CSPstartup`
-cfglib-conn-async-tmout	
	Sets a timeout (in seconds) for a client to expect a TCP success or failure response from the server to which it is connecting. If the timeout expires, all pending connection requests are cancelled, enabling the connection to be made. Without the timeout, the connecting client might eventually stop trying to reconnect, and then stop responding completely because of network issues. This timeout applies to the following connections: • Primary or backup Configuration Server Proxy connecting as a client to primary master Configuration Server • Backup Configuration Server Proxy connecting as a client to primary Configuration Server Proxy If no value is specified for this parameter, a default value of 20 seconds is used.

On UNIX

Go to the directory in which Configuration Server Proxy is installed and do one of the following:

- To use only the required command-line parameters, type the following on the command line:
  `sh run.sh`
- To specify the command line yourself, or to use additional command-line parameters, type the following command on the command line:
  `confserv [<additional parameters and arguments as required>]`

On Windows

Do one of the following:

- Use the **Start > Programs** menu.
- To use only the required command-line parameters, go to the directory in which Configuration Server Proxy is installed, and double-click the **startServer.bat** file.
- To specify the command line yourself, or to use additional command-line parameters, open the MS-DOS window, go to the directory in which Configuration Server Proxy is installed, and type the appropriate command on the command line:
  `confserv.exe [<additional parameters and arguments as required>]`

Local Control Agent

With default settings, Local Control Agent starts automatically every time a computer is started or rebooted. In Windows, you can manually start LCA from the **Start > Programs** menu. You can also change the default LCA port value, following the instructions in Step 3 when Creating a Host.

Starting LCA on Linux Without Root Privileges

RHEL 6

On Red Hat Enterprise Linux 6 systems, you can configure LCA to start automatically when the Host starts, and without root privileges.

To configure the runlevel for LCA on Linux 6, do the following:

- For runlevel 3:
  - LCA: `ln -s /etc/init.d/gctilca /etc/rc3.d/S99gctilca`
- For runlevel 5:
  - LCA: `ln -s /etc/init.d/gctilca /etc/rc5.d/S99gctilca`

> **Important**
>
> For RHEL 6, do not use `/etc/rc.local`, which will cause LCA to start at run levels 2, 3, 4, and 5, which you do not need.

RHEL 7

On Red Hat Enterprise Linux 7 systems, you can configure LCA to start automatically when the Host starts, using the available service method, and without root privileges.

To configure LCA auto start using service method on Linux 7, do the following:

1. Run `cd /etc/systemd/system/`
2. Create the service for LCA (`lca.service`). Name it per your environment standard.
3. Edit `lca.service`:

```
[Unit]

Description=Script to start the LCA

[Service]

Type=simple

WorkingDirectory=/home/admin/bin/LCA

ExecStart=/home/admin/bin/LCA/lca 4999

User=genesys
```

```
KillMode=process

[Install]

WantedBy=default.target
```

4. Run `systemctl daemon-reload`
5. Run `systemctl start lca.service`
6. Run `systemctl enable lca.service`

To configure the runlevel for LCA on Linux 7, do the following:

- For runlevel 3:
  - LCA: `ln -sf /lib/systemd/system/runlevel3.target /etc/systemd/system/default.target`
- For runlevel 5:
  - LCA: `ln -sf /lib/systemd/system/runlevel5.target /etc/systemd/system/default.target`

> **Important**
> For RHEL 7, do not use /lib/systemd/system/rc-local.service, which will cause LCA to start at run levels 2, 3, 4, and 5, which you do not need.

There are various run levels available for Linux, and some of them are listed in the following table. Refer to the Linux website for a complete list of run levels.

Run Level	Description
0	System halt; no activity, the system can be safely powered down.
1	Single user; rarely used.
2	Multiple users, no Network File System (NFS); rarely used.
3	Multiple users, command-line (all-text mode) interface; the standard runlevel for most Linux-based server hardware.
4	User-definable.
5	Multiple users, graphical user interface; the standard runlevel for most Linux-based desktop systems.

Configuring different (but non-root) <user> and <group> for LCA on Linux

1. Install the LCA as root. For RHEL7, use the `systemd` service method to start LCA service. So, during LCA installation in RHEL7, for the step `Do you want to add LCA to startup (RC) files (y/n)?`, specify **n**.
2. Select the <user> and <group> that you want to use as a replacement for the user "root" and the group "root".

3. Ensure that the <user> and <group> each have the adequate privileges for the folders/directories in which LCA is installed, and for the other applications and modules that will be controlled/managed by LCA.
4. Change the owner and group for LCA, as follows:
   a. Set the current working directory to the location where LCA is installed.
   b. Enter the following commands and press `Enter` after each:
      - `chown <user> lca`
      - `chgrp <group> lca`
5. Change the current user from root to <user>.
6. On the command line, enter `su - <user>`, and press **Enter**.
7. Set the `setUID` and `setGID` attributes for LCA. On the command line, enter `chmod ug+s lca`, and press `Enter`.
   This essentially equates the user/group ID to `<user>/<group>` when LCA is launched by another user.
8. Change the current user from <user> to root, and check how LCA will be launched using the root account, by entering the following commands on the command line, pressing `Enter` after each:

```
su -
./lca &
ps -ef | grep lca
```

You should see something like this:

```
UID process
<user> ./lca
```

This indicates that the effective user for LCA is <user> and all applications launched by LCA should have the same effective user id <user>. Normally, this approach of setting UID and GID is used to elevate privileges, but in this case, it is used to downgrade privileges.

Message Server

Message Server supports the common command-line parameters.You can also use the following command-line parameters when starting Message Server:

-log-<log option name> <log-type>	
	Specifies the amount and level of bootstrap logging to occur at startup of Message Server and before it retrieves its log configuration information from the Configuration Database. The log option name must be the same as is specified in the **[log]** section. For example: `MessageServer -host <CS host> -port <CS port> -app <MS Application> -log-verbose all`
-log-<log-type> <filename>	
	Use with **-log-<log option name> <log-type>**

	to specify the name of the file that stores only the logs captured at startup. After Message Server has started up and initialized, the operational logs are stored in the file specified in the Message Server Application object. For example: `MessageServer -host <Configuration Server host> -port <Configuration Server port> -app <Message Server Application> -log-verbose all -log-all MSstartup`

Prerequisites

- Configuration Server is installed and running.

On UNIX

Go to the directory in which Message Server is installed and do one of the following:

- To use only the required command-line parameters, type the following on the command line:
  `sh run.sh`
- To specify the command line yourself, or to use additional command-line parameters, type the following command on the command line:
  `MessageServer -host <Configuration Server host> -port <Configuration Server port> -app <Message Server Application> [<additional parameters and arguments as required>]`

On Windows

Do one of the following:

- Use the **Start > Programs** menu.
- To use only the required command-line parameters, go to the directory in which Message Server is installed, and double-click the **startServer.bat** file.
- To specify the command line yourself, or to use additional command-line parameters, open the MS-DOS window, go to the directory in which Message Server is installed, and type the appropriate command on the command line:
  `MessageServer.exe -host <Configuration Server host> -port <Configuration Server port> -app <Message Server Application> [<additional parameters and arguments as required>]`

Solution Control Server

Solution Control Server uses the command-line parameters common to Framework server components described above. You can also use the following command-line parameters when

starting Solution Control Server:

-f <SCS configuration file>	

	SCS gets Configuration Server's settings from the SCS configuration file. Because the SCS configuration file contains a list of Configuration Servers to which it should try to connect, this option allows SCS to connect to Configuration Server that is running in primary mode.
	
	The SCS configuration file has the filename extension **.cfg** for Windows; **.conf** for UNIX. Here is a sample of the contents:
	
	`[backup_configserver]`
	`host=<backup CS host name>`
	`port=<backup CS port>`
	`name=<SCS application name>`
	`server=primary_configserver`
	
	`[primary_configserver]`
	`host=<primary CS host name>`
	`port=<primary CS port>`
	`name=<SCS application name>`
	`server=backup_configserver`
	
	where `host` is the name of the Host object on which the appropriate Configuration Server is running, as defined in the Configuration Database.

-log-<log option name> <log-type>	
	Specifies the amount and level of bootstrap logging to occur at startup of Solution Control Server and before it retrieves its log configuration information from the Configuration Database. The log option name must be the same as is specified in the **[log]** section. For example: `scs -host <Configuration Server host> -port <Configuration Server port> -app <Solution Control Server Application> -log-verbose all`
-log-<log-type> <filename>	
	Use with **-log-<log option name> <log-type>** to specify the name of the file that stores only the logs captured at startup. After Solution Control Server has started up and initialized, the operational logs are stored in the file specified in the Solution Control Server Application object. For example: `scs -host <Configuration Server host> -port <Configuration Server port> -app <Solution Control Server  Application> -log-verbose`

	`all -log-all SCSstartup`

Prerequisites

- Configuration Server is installed and running.
- If you are starting SCS in Distributed mode, or if HA support or SNMP functionality is required, License Manager must be installed and running.

> **Warning**
> If you are using SNMP, you must start or restart SCS immediately after the SNMP <license option> has been (re)activated. Otherwise, you will lose all SNMP functionality until SCS has been (re)started.

On UNIX

Go to the directory in which SCS is installed and do one of the following:

- To use only the required command-line parameters, type the following on the command line:

  ```
  sh run.sh
  ```

- To start SCS with values from the configuration file, type the following command on the command line:

  ```
  scs -f <name of SCS configuration file>
  ```

  Values for the `-host`, `-port`, and `-app` parameters are read from the specified configuration file.

- To specify the command line yourself, or to use additional command-line parameters, type the following command on the command line:

  ```
  scs -host <Configuration Server host> -port <Configuration Server port> -app <SCS
  Application> [<additional parameters and arguments as required>]
  ```

> **Important**
> If you are operating on a dual-stack machine, and dual stack is enabled, add the following start-up parameter on the command line:
>
> `-transport-ip-version 6,4`
>
> This specifies what internet protocol versions you are using, in this case IPv4 and IPv6.

On Windows

Do one of the following:

- Use the **Start > Programs** menu.
- To use only the required command-line parameters, go to the directory in which SCS is installed, and double-click the **startServer.bat** file.
- To start SCS with values from the configuration file, type the following command on the command line:

  ```
  scs.exe -f <name of SCS configuration file>
  ```

  Values for the `-host`, `-port`, and `-app` parameters are read from the specified configuration file.

  </source>

- To specify the command line yourself, or to use additional command-line parameters, open the MS-DOS window, go to the directory in which SCS is installed, and type the appropriate command on the command line:

  ```
  scs.exe -host <Configuration Server host> -port <Configuration Server port> -app <SCS Application> [<additional parameters and arguments as required>]
  ```

> **Important**
>
> If you are operating on a dual-stack machine, and dual stack is enabled, add the following start-up parameter on the command line:
>
> ```
> -transport-ip-version 6,4
> ```
>
> This specifies what internet protocol versions you are using, in this case IPv4 and IPv6.

SNMP Master Agent

Starting SNMP Master Agent depends on how it is configured—as a Net-SNMP Master Agent or as a Genesys SNMP Master Agent. See SNMP Master Agent for information about the two configurations.

Prerequisites

- Configuration Server is installed and running.
- If you plan to use SNMP alarm signaling, Message Server must be installed and running.

Net-SNMP Master Agent

If your SNMP Master Agent, either a Genesys SNMP Master Agent or a third-party SNMP Master Agent, is configured to work as a Net-SNMP Master Agent, it can be started manually using the commands specific to the platform on which it is running.

Additional Prerequisite:

- Net-SNMP is installed and running.

On UNIX

Go to the directory in which SNMP Master Agent is installed and do one of the following:

- On the AIX 64-bit platform:

```
snmpd -Lsd -Lf /dev/null -p /var/run/snmpd.pid -a
```

- On Linux and Solaris 64-bit platforms:

```
snmpd -Lsd -Lf /dev/null -p /var/run/snmpd.pid
```

Where:

-L	Defines content and make up of logs. Sub-parameters are: **s** — Log to syslog (via the specified facility). **d** — Dump sent and received SNMP packets. **f** — Do not fork from the shell.
p	Stores the process id in FILE.
a	Logs addresses. This parameter is used only on AIX.

On Windows

Go to the directory in which SNMP Master Agent is installed, and type the following on the command line:

```
snmpd.exe -service
```

Genesys SNMP Master Agent

Genesys SNMP Master Agent uses the command-line parameters common to Framework server components, described above.

On UNIX

Go to the directory in which Genesys SNMP Master Agent is installed and do one of the following:

- To use only the required command-line parameters, type the following on the command line:
  ```
  sh run.sh
  ```
- To specify the command line yourself, or to use additional command-line parameters, type the following command on the command line:
  ```
  gsnmpmasteragent -host <Configuration Server host> -port <Configuration Server port>
  -app <Genesys SNMP Master Agent Application> [<additional parameters and arguments as
  required>]
  ```

On Windows

Do one of the following:

- Use the **Start > Programs** menu.
- To use only the required command-line parameters, go to the directory in which Genesys SNMP Master Agent is installed, and double-click the **startServer.bat** file.
- To specify the command line yourself, or to use additional command-line parameters, open the MS-DOS window, go to the directory in which Genesys SNMP Master Agent is installed, and type the appropriate command on the command line:
  ```
  gsnmpmasteragent.exe -host <Configuration Server host> -port <Configuration Server
  port> -app <Genesys SNMP Master Agent Application> [<additional parameters and
  arguments as required>]
  ```

When starting a third-party backup SNMP Master Agent, make sure that you use the following command-line options:

-host	The name of the host on which Configuration Server is running.
-port	The communication port that client applications must use to connect to Configuration Server.
-app	The exact name of the backup SNMP Master Agent Application object as configured in the Configuration Database.

License Manager

For information about starting License Manager, see the *Genesys Licensing Guide*.

Genesys Administrator

Information about starting and stopping Genesys Administrator is located in the *Framework 8.1 Genesys Administrator Deployment Guide*.

Prerequisites

- Configuration Server is installed and running.

HA Proxy

Details on starting and stopping HA Proxy, if applicable, are located in the latest version of the Framework T-Server Deployment Guide for your specific T-Server.

If one or more HA Proxy components are required for T-Server connection to its switch, start HA Proxy before starting T-Server.

Prerequisites

- Configuration Server is installed and running.

T-Server

Details on starting and stopping T-Server are located in the latest version of the Framework T-Server Deployment Guide for your specific T-Server.

Before starting T-Server, be sure that the following components are running:

- Configuration Server
- License Manager

> **Important**
> If an HA Proxy component is required for T-Server connection to its switch, you must start HA Proxy before starting T-Server.

Stat Server

Details on starting and stopping Stat Server are located in the documentation for your release of Stat Server.

Prerequisites

- Configuration Server is installed and running.

> **Important**
> For Stat Server to operate correctly, T-Server must also be running.

Stopping

Server Applications

On UNIX

To stop a server application on UNIX, use one of the following commands:

- `Ctrl+C`
- `kill <process number>`

On Windows

To stop a server application on Windows, do one of the following:

- Type `Ctrl+C` in the application's console window.
- Click **End Task** in the Windows Task Manager.

GUI Applications

Windows-based

To stop a Windows-based GUI application, select **File > Exit** in the main window.

Web-based

To stop a web-based GUI application, such as Genesys Administrator, click **Logout** on the main page.

Using Windows Service Manager

The Genesys setup procedures on Windows operating systems automatically install Genesys daemon applications as Windows Services, with the autostart capability.

When starting an application installed as a Windows Service, make sure that the startup parameters of the application are correctly specified in the ImagePath in the application folder that you can find in the Registry Editor.

The ImagePath must have the following value data:

```
<full path>\<executable file name> -service
<Application Name as Service> -host
<Configuration Server host> -port
<Configuration Server port> -app
<Application Name> -l <license address>
```

where the command-line parameters common to Framework server components are described here and where `-service` is the name of the Application running as a Windows service (typically, it matches the application name specified in the `-app` command-line parameter).

Framework components installed as Windows services with autostart capability are automatically started each time a computer on which they are installed is restarted.

To start Framework components installed as Windows Services with manual start capability, click **Start** in Services Manager.

> **Important**
>
> Use the **Windows Services** window to change the startup mode from Automatic to Manual and vice versa.

To stop any Framework components installed as Windows Services, regardless of the start capability, click **Stop** in Services Manager.

Additional Information

The following pages contain additional information that will help you use Genesys Framework:

- Minimum Required Permissions and Privileges
- Silent Setup
- Generic Configuration Procedures
- Generic Installation Procedures
- Standard Login
- Configuration History Log
- Accessing History of Configuration Changes
- Advanced Disconnect Detection Protocol
- Monitoring Your Configuration Environment
- Transaction Serialization
- MS SQL Cluster Database with MultiSubnet Listener for Framework Applications
- Framework Database Replication for Disaster Recovery
- Internet Protocol version 6 (IPv6)
- IPv6 vs. IPv4 Overview
- Genesys Implementation of Secure Protocol Connections
- Windows Authentication with MS SQL Server for Framework Applications

Minimum Required Permissions and Privileges

This section describes the minimum permissions required to install and run Management Framework components. For information about minimum permissions required for other Genesys components, refer to product- or component-specific documentation.

Minimum System Permissions

The following table provides the minimum system permissions required to install and run Framework components.

Component	Minimum Permissions (UNIX)	Minimum Permissions (Windows)
Configuration Server	Users group	Administrators group [a]
Solution Control Server	Users group	Administrators group
Message Server	Users group	Administrators group
SNMP Master Agent	Users group	Administrators group
Local Control Agent [b]	`root`	Administrators group

a. The user account for the running process is usually determined by the user or object that started the process. For example, if a process is started by LCA, then the process inherits its permissions from LCA.

b. `root` or Administrators permission is required to install the component because, during installation, it updates the startup file and registry.

After a component is installed, you can update the component to start under a different user account with lower privileges. However, before doing so, make sure that you updated the working directories with the correct read and write permissions.

Example

To run LCA as a non-root user, do one of the following, depending on your operating system:

On UNIX

Create startup scripts for LCA that set up LCA to run under the non-root user. For these scripts, it is assumed that LCA is installed in **/home/genesys/GCTI**, and the name of the non-root user is genesys. See LCA Startup Script-gctilca for examples of the script. To install the startup script, put it in the directory **/etc/rc.d/init.d/** and run one or both of the following command:

```
chkconfig -add gctilca
```

On Windows

Change the account associated with the LCA service. One way to do this is through Windows Administrative Services, as follows:

1. Go to **Start > Settings > Control Panel > Administrative Services > Services**, right-click **LCA**, and select **Properties**.
2. Open the **Log On** tab and in the **Log on as** section, select **This account**, and change the account associated with the LCA service.

Minimum Database Privileges

This section describes the minimum required database privileges required to deploy and access the Configuration Database and Log Database.

> **Important**
>
> PostgreSQL 10.x contains an upgraded SCRAM (Salted Challenge Response Authentication Mechanism) authentication. If you are going to use PostgreSQL 10.x and the new SCRAM, set the password encryption algorithm before running any of the PostgreSQL queries mentioned on this page, as follows:
>
> ```
> SET password_encryption TO 'scram-sha-256'
> ```

Configuration Database

This section describes the minimum required database privileges required to deploy and access the Configuration Database.

Deploying the Configuration Database

A database user that accesses the Configuration Database on behalf of Configuration Server, that is, the user identified in the Configuration Server configuration file, requires basic database privileges, as defined in this section.

> **Important**

> When a new database is created, the following instructions and examples assume that the database is accessed by the same user that created it, and that any initialization scripts are run by that same user. If another user runs the scripts, they might not work as described herein.

To deploy the Configuration Database, the user must possess the following minimum required database privileges:

MS SQL

For MS SQL 2000, grant the `public` role to the new database user on the **Database Access** tab of the **SQL Server Login Properties** dialog box for the new user. Grant the following privileges to the new user:

```
GRANT CREATE TABLE TO <DB user>
GRANT CREATE PROCEDURE TO <DB user>
```

For MS SQL 2005 and later, grant the `public` and `db_owner` roles to the new database user.

Oracle

After the new database user is created, grant the following privileges:

```
GRANT CONNECT TO <DB user>
GRANT CREATE TABLE TO <DB user>
GRANT UNLIMITED TABLESPACE TO <DB user>
GRANT CREATE PROCEDURE TO <DB user>
```

PostgreSQL

From `pgAdmin`, create a User and grant the following privileges:

- Can log in
- Can create database object

Or, you can execute the following query:

```
CREATE ROLE <DB user> LOGIN ENCRYPTED PASSWORD <encrypted password> NOINHERIT CREATEDB VALID UNTIL 'infinity';
```

To configure client authentication, update the **pg_hba.conf** file, located in the data directory under the PostgreSQL installation folder. For example:

```
host GCTI_Test gctitest <IP address1>/32 trust
host GCTI_Test gctitest <IP address2>/32 trust
```

This enables the DB user `gctitest` to connect to the `GCTI_Test` database from the hosts `IPaddress1` and `IPaddress2`.

Accessing the Configuration Database

A database user that accesses the Configuration Database on behalf of Configuration Server, that is, the user identified in the Configuration Server configuration file, requires basic database privileges, as defined in this section.

For a user to access the Configuration Database through Configuration Server, the following database privileges are required:

MS SQL

Grant the `public` role to the new database user and grant the following privileges:

```
GRANT SELECT TO <DB User>
GRANT INSERT TO <DB User>
GRANT UPDATE TO <DB User>
GRANT DELETE TO <DB User>
```

Oracle

Create the new user, as follows:

```
CREATE USER <DB User> IDENTIFIED BY <Password>
```

Grant the following permissions to the new user, as follows:

```
GRANT CONNECT TO <DB user>
BEGIN
  FOR x IN (SELECT owner, table_name FROM all_tables WHERE owner='<Table Owner>')
  LOOP
    EXECUTE IMMEDIATE 'GRANT SELECT ON ' || x.owner || '.' || x.table_name || ' TO <DB user>';
    EXECUTE IMMEDIATE 'GRANT INSERT ON ' || x.owner || '.' || x.table_name || ' TO <DB user>';
    EXECUTE IMMEDIATE 'GRANT DELETE ON ' || x.owner || '.' || x.table_name || ' TO <DB user>';
    EXECUTE IMMEDIATE 'GRANT UPDATE ON ' || x.owner || '.' || x.table_name || ' TO <DB user>';
    EXECUTE IMMEDIATE 'CREATE SYNONYM <DB user>.' || x.table_name || ' FOR ' || x.owner || '.' || x.table_name;
 END LOOP;
END;
/
```

PostgreSQL

From pgAdmin, create a new role with the following privilege:

- Can Login

Or, you can execute the following query:

```
CREATE ROLE <DB user> LOGIN ENCRYPTED PASSWORD <encrypted password> NOINHERIT VALID UNTIL 'infinity';
```

Log into the database and grant permissions to the user, as follows:

```
GRANT SELECT ON ALL TABLES IN SCHEMA public TO <DB User>;
GRANT INSERT ON ALL TABLES IN SCHEMA public TO <DB User>;
GRANT UPDATE ON ALL TABLES IN SCHEMA public TO <DB User>;
GRANT DELETE ON ALL TABLES IN SCHEMA public TO <DB User>;
```

To configure client authentication, update the **pg_hba.conf** file located in the data directory under the PostgreSQL installation folder. For example:

```
host GCTI_Test gctitest <IP address1>/32 trust
host GCTI_Test gctitest <IP address2>/32 trust
```

Log Database

This section describes the minimum required database privileges required to deploy and access the Log Database.

Deploying the Log Database

When a new database is created, the following instructions and examples assume that the database is created under the same user that created it, and that any initialization scripts are run by that same user. If another user runs the scripts, they might not work as described herein.

MS SQL

For MS SQL 2000, grant the `public` role to the new database user on the **Database Access** tab of the **SQL Server Login Properties** dialog box for the new user. Grant the following privileges:

```
GRANT CREATE TABLE TO <DB user>
GRANT CREATE PROCEDURE TO <DB user>
```

For MS SQL 2005 and later, grant the `public` and `db_owner` roles to the new database user.

Oracle

After the new database user is created, grant the necessary privileges as follows:

```
GRANT CONNECT TO <DB user>
GRANT CREATE TABLE TO <DB user>
GRANT UNLIMITED TABLESPACE TO <DB user>
GRANT CREATE PROCEDURE TO <DB user>
GRANT CREATE SEQUENCE TO <DB user>
```

PostgreSQL

From pgAdmin, grant the following privileges:

- Can Login
- Can create database object

Or, you can execute the following query:

```
CREATE ROLE <DB user> LOGIN ENCRYPTED PASSWORD '<encrypted password>' NOINHERIT CREATEDB VALID UNTIL 'infinity';
```

To configure client authentication, update the **pg_hba.conf** file, located in the data directory under the PostgreSQL installation folder. For example:

```
host GCTI_Test gctitest <IP address1>/32 trust
 host GCTI_Test gctitest <IP address2>/32 trust
```

This enables the DB user gctitest to connect to the GCTI_Test database from the hosts <IPaddress1> and <IPaddress2>.

Accessing the Log Database

A database user that accesses the Log Database on behalf of Message Server, that is, the user identified in the Message Server Database Access Point, requires basic database privileges, as defined in this section.

MS SQL

For MS SQL 2000, grant the public role to the new database user on the **Database Access** tab of the **SQL Server Login Properties** dialog box for the new user.
For MS SQL 2005 and later, grant the public role to the new database user.
Grant the following privileges to the user:

```
GRANT SELECT TO <DB user>
GRANT INSERT TO <DB user>
GRANT EXECUTE TO <DB user>
```

Oracle

Create the new user, as follows:

```
CREATE USER <DB User> IDENTIFIED BY <Password>
```

Grant the following permissions to the new user:

```
GRANT CONNECT TO <DB user>
GRANT ALL PRIVILEGES ON <Table Owner>.SQ_ATTR_ID TO <DB User>;
CREATE SYNONYM <DB user>.SQ_ATTR_ID FOR <Table Owner>.SQ_ATTR_ID;
GRANT EXECUTE on <Table Owner>.G_LOG_GET_RANGE to <DB User>;
CREATE SYNONYM <DB user>.G_LOG_GET_RANGE FOR <Table Owner>.G_LOG_GET_RANGE;
BEGIN
  FOR x IN (SELECT owner, table_name FROM all_tables WHERE owner='<Table Owner>')
  LOOP
    EXECUTE IMMEDIATE 'GRANT SELECT ON ' || x.owner || '.' || x.table_name || ' TO <DB user>';
    EXECUTE IMMEDIATE 'GRANT INSERT ON ' || x.owner || '.' || x.table_name || ' TO <DB user>';
    EXECUTE IMMEDIATE 'CREATE SYNONYM <DB user>.' || x.table_name || ' FOR ' || x.owner || '.' || x.table_name;
  END LOOP;
END;
/
```

PostgreSQL

From pgAdmin, create a new role:

- Can Login

Or, you can execute the following query:

```
CREATE ROLE <DB user> LOGIN ENCRYPTED PASSWORD <encrypted password> NOINHERIT VALID UNTIL 'infinity';
```

After logging in to the database, grant the following permissions to the user:

```
GRANT SELECT ON ALL TABLES IN SCHEMA public TO <DB User>;
GRANT INSERT ON ALL TABLES IN SCHEMA public TO <DB User>;
GRANT UPDATE ON ALL TABLES IN SCHEMA public TO <DB User>;
GRANT ALL ON SEQUENCE public.sq_attr_id TO <DB User>;
```

To configure client authentication, update the **pg_hba.conf** file, located in the data directory under the PostgreSQL installation folder. For example:

```
host GCTI_Test gctitest <IP address1>/32 trust
host GCTI_Test gctitest <IP address2>/32 trust
```

Sample Scripts

This section contains sample script required to run LCA on UNIX under a non-root user.

LCA Startup Script-`gctilca`

The following is an example of a script to allow LCA to run under a non-root user.

```
#!/bin/bash
#
# chkconfig: 345 80 20
# description: run lca
#
# You should put this script to /etc/rc.d/init.d and run command:
# chkconfig --add gctilca
#GCTI home dir
GCTI=/home/genesys/GCTI
DIRNAME=LCA
HOMEDIR=$GCTI/$DIRNAME
USER=genesys
SCRIPTNAME=gctilca
HOME_USER=/home/genesys
PATH=/sbin:/bin:/usr/bin:/usr/sbin
prog=lca
RETVAL=0
if [ ! -x $HOMEDIR/$prog ]; then
exit 1
fi
# Source function library.
. /etc/rc.d/init.d/functions
start () {
echo -n $"Starting $SCRIPTNAME: "
if [ -e /var/lock/subsys/$prog ]; then
echo -n $"$SCRIPTNAME is already running.";
failure $"cannot start $SCRIPTNAME: $SCRIPTNAME already running.";
echo
return 1
fi
daemon --user=$USER ". $HOME_USER/.bash_profile ; cd $HOMEDIR ;
   ./run.sh >/dev/null 2>/dev/null &"
sleep 1
CHECK=`ps -e | grep $prog | grep -v $SCRIPTNAME | awk '{print $4}'`
if [ "$CHECK" = "$prog" ]; then
RETVAL=0
else
RETVAL=1
fi
[ $RETVAL -eq "0" ] && touch /var/lock/subsys/$prog
echo
return $RETVAL
```

```
}
stop () {
echo -n $"Stopping $SCRIPTNAME: "
if [ ! -e /var/lock/subsys/$prog ]; then
echo -n $"$SCRIPTNAME is not running."
failure $"cannot stop $SCRIPTNAME: $SCRIPTNAME is not running."
echo
return 1;
fi
killproc $prog
RETVAL=$?
echo
[ $RETVAL -eq 0 ] && rm -f /var/lock/subsys/$prog;
return $RETVAL
}
usage ()
{
echo "Usage: service $PROG {start|stop|restart}"
}
case $1 in
start)
start
;;
stop)
stop
;;
restart)
stop
start
;;
*)
usage ; RETVAL=2
;;
esac
exit $RETVAL
```

Silent Setup

Genesys Silent Configuration allows for an automated electronic software distribution, also known as a *silent setup*. With silent setup, you do not have to monitor the setup or provide input via dialog boxes. Instead, the setup parameters are stored in a response file, and the silent setup runs on its own, without any intervention by the end-user.

An installation procedure for a server application differs slightly from an installation procedure for a GUI application. Both, however, require that you update a response file with the necessary parameters and then use it for the actual installation.

Genesys Silent Configuration works on both UNIX and Windows operating systems.

The following Framework components support Silent Setup installation:

- Configuration Server
- Message Server
- Solution Control Server
- T-Server
- HA Proxy
- Stat Server

Creating the Response File

A template for the response file, called **genesys_silent.ini**, is included in the Installation Package (IP) for each supporting component. This template file guides you through the task of entering required information, by providing the following information for each field:

- A full description of the field.
- If applicable, a description of valid values, either a range or a list.
- If application, any conditions in which the parameters may not be used.

Open this file and provide values for all required fields by replacing the text contained in angle brackets (<>)(see the examples). Then save the file. By default, it is saved as **genesys_silent.ini** in the installation folder.

Subsequently, you can use the same response file any time you need to install an application with the configured parameters.

Sample Response File Entries (**genesys_silent.ini**)

The following is an example of the Genesys Configuration Server information section in the **genesys_silent.ini** for Configuration Server, with values entered for the required fields.

[+] Show sample entries

```
#==============================================================================
#        Genesys Configuration Server information section
#        NOTE:        If Genesys Configuration Wizard .ini file (GCTISetup.ini file) is
#                detected in IP root directory, then Host, Port, User,
#                xPassword/Password Configuration Server parameters specified in
#                Genesys Silent Configuration file are ignored.
#==============================================================================
[ConfigServer]

#------------------------------------------------------------------------------
#        Host name where Genesys Configuration Server is running.
#------------------------------------------------------------------------------
Host=CSHost

#------------------------------------------------------------------------------
#        Port number of Genesys Configuration Server.
#------------------------------------------------------------------------------
Port=2010

#------------------------------------------------------------------------------
#        User name in Genesys Configuration Server.
#------------------------------------------------------------------------------
User=User1

#------------------------------------------------------------------------------
#        User's password in Genesys Configuration Server.
#        The password can be specified in encrypted or none encrypted form:
#                xPassword - is used to specify the encrypted password;
#                Password - is used to specify the non encrypted password;
#        If 'xPassword' key value specified then 'Password' key value is ignored.
#------------------------------------------------------------------------------
Password=<plain text password only>

#------------------------------------------------------------------------------
#        Application name in Genesys Configuration Server.
#        NOTE:        This parameter is ignored if only one application was defined in
#                GCTISetup.ini file by Genesys Configuration Wizard (Setup reads
#                application name from '[<ApplicatonName>]' section name
#                of GCTISetup.ini file).
#                This is a mandatory parameter if Installation uses application
#                template in Genesys Configuration Server and GCTISetup.ini file
#                does not exist or contains more then one defined application.
#------------------------------------------------------------------------------
ApplicationName=config
```

> **Important**
>
> The xPassword doesn't work as there is no Genesys recommended mechanism to generate an encrypted password. The silent installation accepts only clear text password.

Running the Silent Installation

The silent setup program does not display a message if an error occurs. The status information for the silent installation is recorded in a file called (by default) **genesys_install_result.log**.

Use the appropriate command line to launch the Genesys Silent Configuration, depending on your operating platform as follows:

On UNIX

```
.\install.sh -s -fr <full path to the setup response file> -fl <full path to the setup log file>
```

where:

	<full path to the setup response file>	
		The full path to the setup response file. By default, **install.sh** looks for a response file called **genesys_silent.ini** in the same directory as **install.sh**.
	<full path to the setup log file>	
		The full path to the setup log file. By default, **genesys_install_result.log** is generated in the same directory as the response file being used.

Example

```
.\install.sh -s -fr /home/user/genesys_silent.ini -fl /home/user/
genesys_install_result.log
```

On Windows

```
.\setup.exe /s /z"-s <full path to the setup response file> -sl <full path to the setup log file>"
```

where:

	<full path to the setup response file>	
		The full path to the setup

		response file. By default, **setup.exe** looks for a response file called **genesys_silent.ini** in the same directory as **setup.exe**.
	<full path to the setup log file>	
		The full path to the setup log file. By default, **genesys_install_result.log** is generated in the same directory as the response file being used.

Important

- Enclose the entire string of parameters **-s <full path to the setup response file> -sl <full path to the setup log file>** in double quotation marks.
- Do not enter a space between the **/z** parameter and its value.

Example

```
.\setup.exe /s /z"-s c:\win\genesys_silent.ini -sl c:\win\genesys_install_result.log"
```

Silent Setup Log File

The silent setup program prints installation results into a setup log file. By default, the results file is named **genesys_install_result.log**, and is stored in the same folder as **genesys_silent.ini**.

Generic Configuration Procedures

This section provides generic instructions for using Genesys Administrator to configure a Genesys Framework Application object. Refer to instructions for a particular application for any application-specific deviations from the standard configuration procedure.

Application Templates

An *Application template* provides a majority of the configuration options for server applications and the default values of those options. Using one application template, you can create as many Application objects of the same type as you need.

Before you configure an Application object, import a template for this Application. If a suitable predefined template is not available, create a new template.

> ### Tip
> Before you continue, make sure you have selected **Show Advanced views** in **User Preferences**. Refer to *Genesys Administrator 8.1 Help* for more information about setting User Preferences.

Import an Application Template

Start of procedure

1. In Genesys Administrator, go to **Provisioning > Environment > Application Templates**, and select **Import template**, located in the slide-out **Tasks** panel on the right.

   > ### Important
   > If **Application Templates** is not listed under **Environment**, open **User Preferences**, and select Show advanced views on the **General** tab. Refer to *Genesys Administrator 8.1 Help*, if necessary.

2. In the window that appears, click **Add**.
3. In the **Choose file** dialog box, locate the installation CD for your product and open the **Templates** folder.
4. Select the template file for your application.

5. Click **Open** to import the template file. The **Configuration** tab for this template is displayed.
6. Make any changes that you require, then click **Save** to save your changes and return to the list of available templates.
7. If there is metadata associated with this template, import the metadata file.

End of procedure

Create an Application Template

Start of procedure

1. In Genesys Administrator, go to **Provisioning > Environment > Application Templates**, and click **New** in the toolbar.

> **Important**
>
> If **Application Templates** is not listed under **Environment**, open **User Preferences**, and select Show advanced views on the **General** tab. Refer to *Genesys Administrator 8.1 Help*, if necessary.

2. Specify the template **Name**, select a template **Type**, and specify a **Version**.
3. If required, define default configuration options on the **Options** tab.
4. Click **Save** to save the changes and return to the list of available templates.

The new template is stored in the **Environment > Application Templates** folder, and can be used to create a new Application object; you do not have to import it.

End of procedure

Application Metadata

Starting with release 8.0, application templates for some Genesys components come with additional XML files called *Application Metadata* files. These files are used by only Genesys Administrator, and provide a user-friendly way to further configure an object. The metadata file contains all of the configuration options that can be used for the particular application, including those that are already in the template.

The metadata file is located in the same folder with the corresponding application template, and has the same filename with the extension **.xml**. To enable the metadata, you must import the metadata file and associate it with the application template.

Prerequisites

- The application template to be associated with the metadata is available.

- You are logged in to Genesys Administrator.

Start of procedure

1. Go to **Provisioning > Environment > Application Templates**, and select the application template to which the metadata is to be imported. The **Configuration** tab for this template is displayed.
2. Click **Import Metadata** in the toolbar.
3. In the window that appears, click **Add**.
4. In the **Choose file** dialog box, locate the installation CD for your particular product and open the **Templates** folder.
5. Select the metadata file for the application.
6. Click **Open** to import the metadata file and associate the metadata with the application template.

End of procedure

After the metadata is imported for a template, a new tab, **Settings**, appears in the details pane for each Application object created from that template. In that new tab, Genesys Administrator displays additional detailed information about configuration options that can be used with that application.

For more information about metadata, refer to *Genesys Administrator 8.1 Help*.

Server Applications

This section contains the procedures necessary to create and configure Server applications.

Creating and Configuring a Server Application

Prerequisites

- The Configuration Layer is installed and running.
- You are logged in to Genesys Administrator.

Start of procedure

1. Go to **Provisioning > Environment > Applications**, and select **New** in the toolbar.
2. From the list of available application templates in the **Browse** dialog box, choose the template for this application. (See Application Templates for information about templates.)
3. In the **General** section of the **Configuration** tab:
   - Enter a name for this application in the text box. The application template provides information for the application type and version.
   - If you have more than one tenant, add the tenants who will be using this application.

- In the **Connections** field, do any of the following as required:
  - Add a connection to any server application to which this application should be a client. To enable Advanced Disconnect Detection Protocol (ADDP) for this connection, see Configuring ADDP.
  - To enable ADDP between this server and Configuration Server, add the Configuration Server Application object (named **confserv**) to the connections and specify the values for the connection protocol, in seconds (see Configuring ADDP.) For more information, refer to *Genesys Administrator 8.1 Help*.
  - Add a connection to Message Server to provide alarm-signaling and centralized-logging capabilities.

4. In the **Server Info** section, specify the following:
   - The host computer on which this server is to be installed and/or to run.
   - Listening ports that applications must use to connect to this server.
   - **Working Directory**—The full path to the directory from which the application starts.
   - **Command Line properties**—The command line used for starting the application; usually, it is the name of the executable file.
   - **Command Line Arguments**—Additional parameters, if any, used for starting the application. Note that the path, command line, and command-line parameters are updated automatically during the application's installation procedure.
   - If another server application is used as a backup for this one, specify the **Backup Server** and the **Redundancy Type**.

   > **Warning**
   > You must have a special high-availability (HA) license to use redundant configurations. Otherwise, the Management Layer does not perform a switchover between the primary and backup servers. Refer to the *Genesys Licensing Guide* for details.

5. Select the **Options** tab and specify (or change) the values of the configuration options as necessary. Click the option name for its description. For additional information about the options, see:
   - The *Framework Configuration Options Reference Manual* for Configuration Layer and Management Layer component options.
   - The latest version of the *Framework T-Server Deployment Guide* for your specific T-Server and/or HA Proxy options.
   - The latest version of the *Framework 8.1 Stat Server User's Guide* for Stat Server options.

   If the application's working directory differs from the directory in which the application was originally installed, configure an option named **messagefile** in the **[log]** section. Specify the full path to the application-specific log messages file (***.lms**) as the option value. Otherwise, the application is unable to generate its specific log events.

6. Click **Save** or **Apply** to save your changes. The new application is now listed in the list of applications.

End of procedure

> **Important**
> If you configure two applications as a redundant pair (primary and backup), Genesys strongly recommends that you synchronize configuration options and server ports between the two applications. When a port is defined on the primary server application, a compatible port is automatically allocated on the backup server application. If the two server applications are configured as a redundant pair, you cannot remove or change the ports on the backup server. If the two are not linked as a redundant pair, you can delete the ports on the application that had been the backup.

Configuring ADDP

You can enable ADDP (Advanced Disconnect Detection Protocol) for a connection between any two Genesys applications that are configured as client-server pair and that support ADDP.

> **Important**
> Some applications do not support ADDP for certain connections. Refer to application-specific documentation or Release Notes to determine if your application supports ADDP.

Prerequisites

- The Configuration Layer is installed and running.
- Application objects for each application in the client-server pair exist.
- You are logged in to Genesys Administrator.

Start of procedure

1. In Genesys Administrator, go to **Provisioning > Environment > Applications**, and select the client application in the client-server pair.
2. Select the **Configuration** tab, and expand the **General** section.
3. In the **Connections** list, click **Add**.
4. In the **CfgConnectionInfo** dialog box that opens:
   a. From the list of servers, select the application name that represents the connection for which you want to configure ADDP.
   b. Specify addp as the value for the **Connection Protocol** field.
   c. Specify any integer as the value for the **Local Timeout** field. This indicates how often, in seconds,

the client application sends polling signals to the server application.

> **Tip**
> To avoid false disconnect states that might occur because of delays in the data network, Genesys recommends setting the ADDP timeouts to values equal to or greater than ten (10) seconds.

d. If you also want to enable polling signals from the server application to the client, specify any integer as the value for the **Remote Timeout** field. This timeout is also measured in seconds.

e. In the **Trace Mode** field, select one of the following:

   a. Select `Trace On Client Side` to turn on ADDP at the client. The client application will generate ADDP-related messages in its logs.

   b. Select `Trace On Server Side` to turn on ADDP at the server. The application will generate ADDP-related messages in its logs.

   c. Select `Trace On Both Sides` to turn on ADDP at both the client and server. The Client and server applications will both generate ADDP-related messages in their logs.

   d. Select `Trace Is Turned Off` to turn off ADDP tracing altogether. ADDP-related messages will not be generated.

f. Click **OK**, and then **Save** to save the configuration changes.

End of procedure

For more information about ADDP, refer to Advanced Disconnect Detection Protocol.

Graphical User Interface Applications

To create and configure a GUI Application object:

Prerequisites

- The Configuration Layer is installed and running.
- At least one of the servers to which the GUI connects is installed.
- You are logged in to Genesys Administrator.

Start of procedure

1. In Genesys Administrator, go to **Provisioning > Environment > Applications**, and select **New** in the toolbar.
2. From the available application templates in the **Browse** dialog box, choose the template for this application. See Application Templates for information about templates.
3. In the **General** section of the **Configuration** tab, enter a name for this application in the text box. The application template provides information for the application type and version.
4. Select the **Connections** tab. If necessary, add connections to any server applications to which this

GUI application must connect.

5. Click **Save** to save your changes. The new GUI application is now listed in the list of applications.

End of procedure

Generic Installation Procedures

This section provides instructions for installing a typical Genesys application that you have configured using Genesys Administrator.

Refer to the instructions for a particular application for the location of installation packages on a product CD and for any application-specific deviations from the standard installation procedure.

Server Applications

This section describes a standard installation procedure for a server application on UNIX and Windows operating systems.

Prerequisites

- An Application object exists for the server application. See Creating and Configuring a Server Application object.

On UNIX

> **Warning**
> During installation on UNIX, all files are copied into the directory you specify. The install process does not create any subdirectories within this directory; therefore, do not install different products into the same directory.

1. On the product CD, locate a shell script called **install.sh**.
2. Run this script from the command prompt by typing the file name.
3. When prompted, specify the **Host Name** of the computer on which this server is to run.
4. When prompted, specify the:
   - **Host Name** of the computer on which Configuration Server is running.
   - **Port** used by client applications to connect to Configuration Server.
   - **User Name** used to log in to the Configuration Layer.
   - **Password** used to log in to the Configuration Layer.
5. The installation displays the list of applications of the specified type configured for this host. Type the number of the server application to be installed.

6. Specify the full path of the destination directory into which this server is to be installed.
   If the installation script finds that the destination directory is not empty, it prompts you to do one of the following:
   - Back up all files in the directory.
   - Overwrite only the files contained in this package.
   - Wipe the directory clean.

   Type the number that corresponds to your selection and confirm your choice.
7. If asked which version of the product to install, 32-bit or 64-bit, choose the one appropriate to your environment.
8. If you plan to use functionality that requires a license, such as Solution Control Server (SCS) with Simple Network Management Protocol (SNMP), type y when prompted and enter one of the following:
   - The full path to the license file
   - The License Manager port and host

As soon as the installation process is finished, a message appears indicating that installation was successful. The process places the server application in the directory specified during the installation.

On Windows

> **Warning**
> Genesys does not recommend installation of its components via a Microsoft Remote Desktop connection. The installation should be performed locally.

1. On the product CD, locate and double-click **setup.exe** to start the Genesys Installation Wizard.
2. Click **About** on the wizard's **Welcome** page to review the **read_me** file. The file also contains a link to the server's Release Notes file.
3. Click **Next** on the **Welcome** page to proceed with the installation.

> **Tip**
> Click **Next** at the end of each step to proceed to the next page.

4. On the **Connection Parameters to the Genesys Configuration Server** page, specify the following login parameters:
   - `Host` and `Port` of Configuration Server
   - `User name` and `Password` used to log in to the Configuration Layer.
5. The **Select Application** page displays all applications of this type that the Configuration Database

contains. When you select one application from the list, the wizard displays some parameters configured for the selected application (such as application type, host, working directory, command line, and command-line arguments).
Select the application to install.

> **Warning**
> If the component does not require a technical license, omit step 6 and step 7. If the component requires a technical license for startup, omit step 6. If the component requires a technical license to enable a certain feature, but the license is not otherwise required, proceed with step 6.

6. On the **Run-time License Configuration** page, select one of the following options:
   - `Use License` if you plan to use features that require special licenses.
   - `Without License` if you do not plan to use features that require special licenses. In this instance, go to step 8.
     If you decide to use a licensed feature later on, reinstall the server and enter the appropriate license information through the Genesys Installation Wizard.
7. On the *Access to License* page, select one of the following options:
   - `License Manager`-You want your server application to use host and port parameters to connect to the license server. In this instance, you must enter values for the host and the port of the license server.
   - `License File`-You want your server application to retrieve license server information from the license file. Click **Browse** to navigate to the license file.
8. On the **Choose Destination Location** page, the wizard displays the destination directory, as specified in the **Working Directory** property of the server's Application object. If the path configured as **Working Directory** is invalid, the wizard generates a path to the destination directory in the **c:\Program Files\GCTI\<Product Name>** format.</bw> If necessary, do one of the following:
   - Click **Browse** to select another destination folder. In this case, the wizard will update the Application object's **Working Directory** in the Configuration Database.
   - Click **Default** to reinstate the path specified in **Working Directory**.
9. On the **Ready to Install** information page, do one of the following:
   - Click **Back** to update any installation information.
   - Click **Install** to proceed with installation. **Installation Status** displays the progress of the installation.
10. On the **Installation Complete** page, click **Finish**.

As a result of the installation, the wizard adds Application icons to the:

- Windows **Start** menu, under **Programs > Genesys Solutions**.
- Windows **Add or Remove Programs** window, as a Genesys server.
- Windows **Services** list, as a Genesys service, with Automatic startup type.

Graphical User Interface Applications

This section describes a standard installation procedure for a graphical user interface (GUI) application on Windows operating systems. If you are installing a web- or UNIX-based GUI, refer to the product documentation for installation instructions.

If you want to implement a security banner with a GUI application, make sure that you have the necessary files prepared before you start installing the GUI application. Refer to the *Genesys Security Deployment Guide* for detailed information about the Security Banner feature.

1. From the product CD, locate and double-click **setup.exe** to start the Genesys Installation Wizard.
2. Use the **About** button on the wizard's **Welcome** page to review the **read_me** file. The file also contains a link to the application's Release Notes file.
3. Click **Next** to proceed with the installation.
4. On the **Security Banner Configuration** page, choose whether you want to configure a security banner for this GUI application.
   Do one of the following:
   - If you do not want to configure a security banner for this application, clear the **Enable Security Banner** check box, and click **Next**.
   - If you want to configure a security banner for this application:
     i. Select **Enable Security Banner**.
     ii. Follow the instructions in the procedure "Installing and configuring the Security Banner" in the *Genesys Security Deployment Guide*. When you are finished that procedure, return here and finish this procedure.
5. On the **Choose Destination Location** page, the wizard displays the path to the destination directory in the **c:\Program Files\GCTI\<Product Name>** format.
   If necessary, use the:
   - **Browse** button to select another destination folder.
   - **Default** button to reinstate the wizard-generated path (**c:\Program Files\GCTI\<Product Name>**).

   Click **Next**.

> **Important**
> If the GUI application requires any non-standard installation input from the user, provide that information as prompted by the extra pages that appear here.

6. On the **Ready to Install** page, click:
   - **Back** to update any installation information.
   - **Install** to proceed with the installation. **Installation Status** displays the progress of the installation progress.
7. On the **Installation Complete** page, click **Finish**.

As a result of the installation, the wizard adds Application icons to the:

- Windows **Start** menu, under **Programs > Genesys Solutions**.
- Windows **Add or Remove Programs** window, as a Genesys application.

Troubleshooting the Installation of a Genesys Application

To determine and fix the cause of a warning generated during the installation procedure for any Genesys application that Configuration Server is unavailable and that configuration cannot be updated, do the following:

1. Finish installing the Genesys application.
2. When installation of the application is complete, open the **Configuration** tab of the corresponding Application object.
3. Select the **State Enabled** check box.
4. Verify that **Working Directory**, **Command Line**, and **Command Line Arguments** are specified correctly.
5. Save the configuration updates.

Standard Login

When you start a Framework graphical user interface (GUI) application, or if you are being forced to log in again after a period of inactivity, a `Login` dialog box displays. The Configuration Layer checks the information specified in the `Login` dialog box and determines the user's permission to view, create, and modify objects in the Configuration Database.

> **Important**
>
> Configuration Layer will not allow users whose use of Genesys Administrator has been disabled to log into Genesys applications.

Logging In

To log in to a Framework GUI, do the following:

1. Start the application. Refer to the documentation for the particular application for specific instructions.
2. In the `Login` dialog box:
   a. Enter a user name. If you are logging in to the Configuration Layer for the first time, use the Master Account user name, which is `default`. After the appropriate configuration objects of the User type are added to the configuration, use a customized user name.
   b. Enter a user password. If you are logging in to the Configuration Layer for the first time, use the Master Account password, which is `password`. After the appropriate configuration objects of the User type are added to the configuration, use your own password. Your System Administrator will notify you if any requirements or restrictions apply to your password (see User Password Requirements and Restrictions).
   If you have configured Configuration Server to allow access with a blank password, you can optionally leave the **Password** field empty. Refer to the *Framework Configuration Options Reference Manual* for information on configuring this functionality.
   c. Click either **Details** or **More** to display additional input login fields.
   d. Enter the application name, which is the instance of the application to which you are logging in, as it is registered in the Configuration Database.

      > **Important**
      >
      > The predefined name of the Genesys Administrator object is **default**. You can rename it later.

   e. Enter a host name, which is the name of the computer on which Configuration Server runs.
   f. Enter a port number, which is the number of the communication port that client applications use to connect to Configuration Server.

If your configuration uses both primary and backup Configuration Servers, your GUI applications might automatically reconnect to the backup server if they lose their connection to the primary server. You can specify automatic or manual reconnection; refer to the on-line Help file of your GUI application.

User Password Requirements and Restrictions

Starting in release 8.1.1, the System Administrator or other authorized person can configure restrictions for user passwords and how they are used. The restrictions include:

- The type and case of characters allowed in a password.
- Whether a password can expire, and after how long.
- After using a given password, how many different passwords must be used before using that password again, or if re-use is permitted at all.
- Whether the user must change their password the next time they log in.
- The number of unsuccessful login attempts that can be made after which the account is locked.

For more information about these requirements, and how to configure them, refer to the *Genesys Security Deployment Guide*.

Configuration History Log

The Configuration History Log consists of a set of a records that contains historical information about client sessions and changes to configuration objects. It enables a client to restore a session that was terminated by a service interruption, and request any changes to configuration objects that occurred during that service interruption.

For all Configuration Servers, the records are stored in the Configuration Database. Configuration Server Proxy reads the information from its primary Configuration Server. A limited number of latest records about object changes and client connections are synchronized with all Configuration Server Proxies to facilitate session restoration. Records about configuration object changes older than the value defined by the max-records and expiration options are not used for session restoration and can only be accessed from the master Configuration Server database, as explained in Accessing History Of Configuration Changes. These records provide extended audit information within the Configuration Server database.

The audit-max-records and audit-expiration options enable you to manage an extended number of audit log records in the configuration database. Configuration Server cleans up the history records based on these options every 30 minutes.

The History Log comes with default parameters when you install Configuration Server. You configure the History Log parameters in the options of the Configuration Server Application object in Genesys Administrator. Refer to the *Framework Configuration Options Reference* for detailed descriptions of the configuration options that relate to the History Log.

When requested by a client that is recovering from a service interruption, Configuration Server or Configuration Server Proxy does the following:

- Restores the client's session according to a client session record.
- Returns all data that has been changed since that client disconnected.

History Log functionality is mandatory, and cannot be turned off permanently.

History Log Maintenance

No maintenance is required for the History Log, because it is maintained automatically by Configuration Server. The history log records are stored in the Configuration Database and are maintained using configuration records. Based on the expiration parameters, Configuration Server purges information from the database, both at startup and during normal operations.

History Log Errors

Any errors that occur when writing to the History Log generate Log Event 21-22138.

> **Important**
>
> Genesys strongly recommends that you associate an alarm with this Log Event, and that you inform Genesys Customer Care if you encounter any errors or corruption.

Minimizing Performance Impacts

Depending partially on the size of the updates, the History Log can affect the performance of Configuration Server. To minimize these performance impacts, you can turn off the History Log functionality temporarily by setting the **active** option to `false` for the Configuration Server Application object. The functionality will be turned back on either when you manually reset the option (to `true`), or when you restart Configuration Server.

> **Warning**
>
> When History Log functionality is turned off, current activities are not recorded. Therefore, clients that are disconnected during this time cannot retrieve the updates necessary to restore their sessions.

If you want to keep the History Log active (that is, **active=**`true`, consider setting **write-former-value=**`false` when performing large or significant updates. This will prevent previous values from being written to the history database, but will greatly improve performance.

Refer to the *Framework Configuration Options Reference* for more information about the options used to configure the Configuration Server History Log.

Accessing History of Configuration Changes

Configuration Server uses the Configuration History Log to keep track of changes being made in the Configuration Database. The History Log within Configuration Server contains more detailed information than is output in Audit-level log messages. Starting in release 8.5.0, Genesys provides a tool to extract this detailed information.

Also in release 8.5.0, previous values (as they were before a change) can also be stored in the history log along with the actual changes.

> **Important**
>
> The history log does not provide complete information about changes to assigned access Permissions

Retrieving the History

Extract the changes history to an XML file by starting an instance of Configuration Server (any instance in the configuration will suffice; it does not have to be the Configuration Server currently running) and specifying the following parameter on the command line with the startup command:

```
-dumpauditxml <file name> [-last <days>]
```

where:

`<file name>` - The name of the XML file to which the information will be extracted. The information will be in the XML format shown in the example below.

`-last <days>` - (Optional) The results for the operations for the last number of days; if this argument is not specified, all audit history in the database is exported.

The Configuration Server starts, exports the information into the file specified in the parameter, and then terminates.

Exported File Format

The exported XML file contains two primary sections. The **CfgAuditEntry** section contain information about the type of update, what object was updated, and who updated it. Its fields are described in the table below. Each **CfgAuditEntry** section contains one or more **CfgHistoryRecord** sections, that contain the former value and the action that was taken during the update identified in the **CfgAuditEntry**. An excerpt of a sample exported file is available at the end of this topic.

Field Name	Description
id	The ID of the audit record that exists in the database.
operation_type	The type of operation performed based on the internal enumeration of the Configuration Server implementation.
operation_time	The timestamp when the operation occurred.
object_dbid	The internal DBID of the object being updated.
user_name	The username of the user performing the update.
object_data_size	The size of the audit record, as contained in the CfgHistory Record tag.
object	The name of the object being updated.
host	The name of the host or IP address from which the user performing the update is connected to Configuration Server.
application	The name of the user application that is connected to Configuration Server when performing the update operation.
application_dbid	The DBID of the user application that is connected to the Configuration Server when performing the update operation.
user_tenant	The name of the tenant to which the user performing the update belongs.
user_tenant_dbid	The DBID of the tenant to which the user performing the update belongs.
tenant	The name of the tenant to which the object being updated belongs.
tenant_dbid	The DBID of the tenant to which the object being updated belongs.

Sample File

```
<CfgAuditEntry
   id="187"
   operation_type="4"
   operation_time="[01/07/14 09:35:04]"
   object_type="CfgCampaign"
   object_dbid="101"
   user_name="default"
   object_data_size="529"
   object="CampaignA"
   host="135.17.178.16"
   application="default"
   application_dbid="100"
   user_tenant="Environment"
   user_tenant_dbid="1"
   tenant="Environment"
   tenant_dbid="1">
```

```
<CfgHistoryRecord
  id="187">
 <former_value>
  <action
    action="change">
   <CfgCampaignUpdate
     DBID="101">
    <callingLists
      action="change">
     <CfgCallingListInfo
       linkDBID="101"
       share="22"     />
    </callingLists>
   </CfgCampaignUpdate>
  </action>
 </former_value>
 <action>
  <CfgCampaignUpdate
    DBID="101">
   <callingLists
     action="change">
    <CfgCallingListInfo
      linkDBID="101"
      share="20"    />
   </callingLists>
  </CfgCampaignUpdate>
 </action>
</CfgHistoryRecord>
 </CfgAuditEntry>
```

Advanced Disconnect Detection Protocol

All but a few Genesys interfaces use the TCP/IP stack. To compensate for the manner in which this stack operates, Genesys components use the Advanced Disconnect Detection Protocol (ADDP), which periodically polls the opposite process when no actual activity occurs at a given connection. If a configurable timeout expires without a response from the opposite process, the connection is considered lost and an appropriate event is sent to the application.

Genesys recommends enabling ADDP on the links between any pair of Genesys components. ADDP helps detect a connection failure on both the client and the server side. For most connections, enabling detection on the client side only is sufficient and it reduces network traffic. However, Genesys strongly recommends that you use detection on both sides for all connections between Configuration Server and its clients (including Solution Control Interface), as well as between any two T-Servers.

To enable ADDP between two applications, specify addp as the Connection Protocol when configuring the connection between applications; also, set values for the **Local Timeout**, **Remote Timeout**, and **Trace Mode** properties. For more information, refer to the *Framework Configuration Options Reference Manual*.

For complete instructions on configuring ADDP between two applications, refer to Configuring ADDP. For instructions on configuring ADDP between primary and backup T-Servers, refer to the Deployment Guide for your specific T-Server.

After a communication session failure is detected, the application makes repeated attempts to regain access to the required resource. If a redundant process is not configured, the reaction is a repeated attempt to restore the communication session with the same process. If a redundant process is configured, the application makes alternate attempts to restore the failed communication session and to establish a session with the redundant process. This way, if the session has terminated because of a failure of the opposite process, the application eventually connects to the standby process configured to provide the same type of service.

Important

Backwards compatibility of the Keep-Alive Protocol (KPL) is not supported. If you used KPL in previous versions of Genesys, consider using ADDP instead.

Tip

If you are unfamiliar with ADDP or just want to refresh your knowledge of it, you might be interested in an ADDP Tutorial, available from the Genesys Customer Care website at https://genesyspartner.force.com/customercare/articles/Tech Tutorial 11 ADDP. You will need your Genesys login credentials to access the tutorial.

Monitoring Your Configuration Environment

Management framework provides several tools that are intended to provide observability into platform state. The Configuration Database is the backbone of your configuration environment. There are several scripts available that you can use to monitor the performance and the health of your Configuration Database. Configuration Database monitoring is based on a set of scripts to retrieve statistics from the Configuration Database that you can use to determine the potential or present issues that affect the performance and health of the database and your data. The database scripts used for monitoring are distributed as part of the Configuration Database Maintenance Scripts IP, in the **/tools** folder. Scripts are available for the MSSQL, Oracle, and PostgreSQL Database Management Systems, and can be run on Configuration Server Database Schema version 8.1 and newer. Configuration Server can be monitored and have its runtime metrics collected from statistical log files or queried directly for set of metrics in a format that is compatible with Prometheus monitoring platform.

Verifying Configuration Database Health with Health Monitoring Scripts

The following health monitoring scripts are available:

- **csv_search_backslash.sql:** This script, for PostgreSQL databases only, enables you to ensure that options containing two or more sequential backslash characters (\\) are the result of an incorrect setting in the PostgreSQL database.
- **csv_search_duplicates_flex_prop.sql:** This script enables you to determine if any objects of a given type contain duplicate Annex records.
- **search_orphan_flex_prop.sql:** This script reports any orphaned Annex options.
- **csv_check_cfg_max_dbid.sql:** This script provides metrics about the validity of the DBID for each object type, all <object-type> Group types, and the history log.

Refer to Monitoring Health of Configuration Environment for more information about these scripts.

Each script can be run against the Configuration Database at any time without stopping Configuration Server or restricting its access to the database.

Each script generates CSV-formatted output containing key metrics, and includes a header. The output can be used for further analysis, using tools that allow CSV processing, such as Microsoft Excel. You can view the output on screen, in a text editor, or import it into an application that displays the file in a spreadsheet, such as Microsoft Excel. You can also search for, and view, objects using Genesys Administrator.

The following performance scripts are available:

- **csv_objtype_sizes.sql:** This script calculates usage for major types of configuration objects.
- **csv_largest10_objbytype.sql:** This script enables you to determine if any particular object of a given type is significantly larger than the others.

- **csv_biggerthan_objbytype.sql:** This script helps you identify the number of configuration objects per object type that impact performance the most, and/or are out of the range of expected size for this type.

Refer to Monitoring Performance of Configuration Environment for more information about these scripts.

Limitations

The scripts do not support the following:

- Configuration Database version 8.1 or earlier with a non-English locale
- PostgreSQL DBMS prior to version 9.0

Viewing the Script Output

The output of the scripts can be viewed in several ways:

- Display the output on your screen.
- Save the output to a **.csv** file and open it in a text editor.
- Save the output to a **.csv** file and import it into a spreadsheet, such as Microsoft Excel.
- Save the output to a **.csv** file and upload it into Genesys Administrator, as described below.

Viewing Script Output in Genesys Administrator

You can upload the .csv file into Genesys Administrator version 8.1.309.02 or later. You can then click any object in the list to view the object itself, searching for it if necessary. You can also filter the list by object name and/or type and export it to another **.csv** file. Use the following steps:

Step 1: Upload the file and display it

To upload the file generated by the scripts, set the new **search-by-csv-file-mode** option to `true` in the new **[SpecialSearchProperties]** section of the Genesys Administrator object. Then, in Genesys Administrator, navigate to **Provisioning** > **Search**, click **Choose File**, and select the file. Click the **Search** button to display the contents of the file.

search-by-csv-file-mode
Default Value: `false`
Valid Values: `true`, `false`
Changes take effect: After logging out of Genesys Administrator

Specifies whether the **Choose File** button appears in the **Search** window in the **Provisioning** module. If set to `true` and you are logged in to Genesys Administrator, use this button to upload the file into Genesys Administrator.

Step 2: Export results to another file

To export search results to another **.csv** file, set the new **search-to-csv-export-mode** option to `true` in the same section of the Genesys Administrator object, then click **Export**.

search-to-csv-export-mode
Default Value: `false`
Valid Values: `true`, `false`
Changes take effect: After logging out of Genesys Administrator

Specifies whether the **Export** button appears in the **Search** window in the **Provisioning** module. If set to `true`, and you are logged in to Genesys Administrator, use this button to export the results of a search of the output of the Configuration Database Monitoring scripts, to another **.csv** file.

> **Tip**
> You can also use this option and the **Export** button to export the results of a search of the Configuration Database (by Object Name and Object Type) to a **.csv** file.

Monitoring Configuration Server activity and performance

Certain aspects of CS performance and its current activity can be monitored via Prometheus reporting and/or Statistical logging. These two features expose the same performance data via different interfaces/formats.

Reported data includes overall performance statistics, statistics pertinent to currently connected clients as well as statistics on certain aspects of the CS internal operation.

Prometheus support

This feature exposes CS performance and its current activity via open HTTP port in Prometheus text format. Prometheus can be used to periodically collect and keep this data for further querying and graphical representation. Alternately, web browser or curl can be used to retrieve current statistical data for real-time monitoring.

Promethus HTTP reporting - Configuration Server (CS) exposes the statistics of its internal operation via HTTP interface as a Prometheus metrics endpoint.

If configured, upon startup, CS opens a specified port for the HTTP GET requests from the Prometheus and/or web browser. Both http and https protocols are supported.

GET requests for /metrics URI path are responded with content intended for Prometheus consumption. The responses contain text in Prometheus text-based format.

Requests for paths other than /metrics are responded with http error 404 Not Found. Exposed metrics correspond to the statistical data currently exposed via the stat.log files.

Statistical logging

This feature exposes CS performance and its current activity, same as exposed for Prometheus, in text format in stat.log files. The stat.log files are created and maintained along with standard logs.

In statistical logging, the same information is produced in different formats. Statistical logging puts

this information into a log file and separate log file. Prometheus support exposes it through HTTP interface intermediate format.

Monitoring Performance of Configuration Environment

The performance of a configuration environment depends on multiple factors, including the total number of configuration objects, the number of objects of a particular type, and the size of data associated with each object. Configuration Server and Configuration Server Proxy startup performance depends on the total number of all objects and all configuration options that need to be read and processed. Startup time of a client depends on the number and size of objects of particular types this client is loading.

Configuration Database Performance Monitoring Scripts

Configuration Database performance monitoring is based on a set of scripts to retrieve statistics from the Configuration Database that, together, provide reports on overall usage (number of objects of each type and the amount of data attached to each object), the ten largest objects (in terms of the number of configuration options) per type, and objects larger than 20 kb (in terms of the size of the associated configuration options) per type. With these statistics in hand, it is possible to trace the configuration areas that consume most of the Configuration Server resources, both overall and for particular types of clients (when client behavior and configuration dependencies are known).

The following scripts are available:

- **csv_objtype_sizes.sql:** Calculates data usage for 37 major types of configuration objects.
- **csv_largest10_objbytype.sql:** Provides object names and associated sizes of their configuration options, in characters, of the ten largest objects for every major type.
- **csv_biggerthan_objbytype.sql:** Provides object names and associated sizes of their configuration options, in characters, of all objects of a particular type for which the size of its options exceeds a given upper limit.

Limitations

The scripts do not support the following:

- Configuration Database older than version 8.5 with a non-English locale.
- PostgreSQL older than version 9.0

Running the Performance Monitoring Scripts

The scripts can be executed using DBMS utilities or from the command line.

Using DBMS Utilities

Run the script using the corresponding DBMS utility, as follows:

- MSSQL scripts: Microsoft SQL Server Management Studio 2005 or newer
- Oracle scripts: Oracle SQL Developer 1.5 or newer
- PostgreSQL scripts: pgAdmin III 1.12 that supports Postgre 9.0 or newer

The script output appears on the console. Capture the output and save it as a **.csv** file, suitable for further analysis.

From the Command Line

The Configuration Database Maintenance Scripts IP also contains command-line scripts for Windows (MS SQL, Oracle, and Postgre databases) and Linux (Oracle and Postgre databases) operating systems. Each command-line script runs the three SQL scripts, and writes the output to a **.csv** file in a specified results directory. This directory must exist before running the command-line script.

Each command-line script generates three **.csv** report files, as follows:

- **csv_biggerthan_objbytype_<DBMS_TYPE>.csv**
- **csv_largest10_objbytype_<DBMS_TYPE>.csv**
- **csv_objtype_sizes_<DBMS_TYPE>.csv**

The command-line scripts are listed in the following table.

DBMS	Operating System	
Windows	**Linux**	
Windows MS SQL Server	\mssql\monitor_MSSQL_DB.bat	Not available
Oracle	\oracle\monitor_ORACLE_DB.bat	/oracle/monitor_ORACLE_DB.sh
Postgre [a]	\postgre\ monitor_POSTGRE_DB.bat	/postgre/ monitor_POSTGRE_DB.sh

[a] For Postgre scripts, configure the PostgreSQL server to allow a client connection with the host on which the scripts are run.

Command Syntax

Run the script from the folder in which it is located, using the following syntax:

```
<script> <USER> <PASSWORD> [<HOST> <PORT> or <HOST[\INSTANCE]>] [<SERVICE_NAME> or <DB_NAME>]
<CSV_RESULT_DIR>
```

where:

- <script> is the name of the script, as provided in the table above.
- <USER> is the name of a user authorized to access the database.

- <PASSWORD> is the password required for the authorized user to access the database.
- <HOST> is the name of the host on which the database is located.
- <PORT> is the number of the port on that host that is used by the DBMS.
- <HOST\INSTANCE> is the database host and instance, if not the default.
- <SERVICE_NAME> is the Oracle Instance Service Name, that is, the SERVICE_NAME value in **TNSNAMES.ORA**.
- <DB_NAME> is the name of the Database.
- <CSV_RESULT_DIR> is the path to the directory in which the output **.csv** files will be stored.

> **Tip**
>
> To see the syntax on-line, with no values for parameters, enter the script name and press **ENTER**.

Examples:

MS SQL Server

```
.\monitor_MSSQL_DB.bat sa password localhost\SQLEXPRESS TestDB D:\MF_mon\results\mssql
.\monitor_MSSQL_DB.bat sa password somehost SomeTestDB D:\MF_mon\results\mssql
```

Oracle

```
./monitor_ORACLE_DB.sh sourcedb sourcedb localhost 1521 orcl.genesys.com '/home/genesys/
results/oracle'
```

Postgre

```
./monitor_POSTGRE_DB.sh postgres password localhost 5432 TestDB '/home/genesys/results/
postgre'
```

csv_objtype_sizes.sql

This script calculates usage for major types of configuration objects, including:

- The total number of objects of this type.
- The number of objects of this type that have configuration options associated with them.
- The total size, in characters, of all options for all objects of this type.

Running this script periodically reveals an increase in object count and an associated increase in options size (if any) because of new objects being added. This provides overall statistics that can be further analyzed using other monitoring scripts and tools.

Genesys recommends that you run this script daily. The audit trail for a given day can be used to

determine the actual objects added on that day.

This script produces CSV-formatted output on the console if run in the DBMS utility, or in the **csv_objtype_sizes_<DBMS_TYPE>.csv** file if run from the command-line. There is one header row, and 37 data rows in the output. Each data row represents the data for one major object type. 37 major types of configuration options are analyzed. The header row identifies each column in the data rows, as follows:

- **type**: The type of object, as visible in Genesys Administrator. (The one exception is that a **User** or **Agent** object type is named **Person**.)
- **object_count:** The total number of objects of this type.
- **objects_with_opts:** The total number of objects of this type that have associated configuration options.
- **opt_size_chrs:** The total size of all configuration options (including the total number of characters in strings in key-value pairs) of all objects of this type, expressed as the total number of characters or bytes.

Sample Output, in CSV format

```
type,object_count,objects_with_opts,opt_size_chrs
Switch,9,4,480
DN,54604,41982,12373545
Person,23656,22349,357163117
...
```

csv_largest10_objbytype.sql

This script generates a drill down report enabling you to determine if any particular object of a given type is significantly bigger than the others, possibly requiring further review or follow up with the object's owner or its consumers.

Running this script daily can reveal the addition of a significant number of options to particular objects. The audit trail for a given day can be used to determine who actually made the changes.

This script produces CSV-formatted output on the console if run in the DBMS utility, or in the **csv_largest10_objbytype_<DBMS_TYPE>.csv** file if run from the command-line. There is one header row, and up to 370 data rows in the output. Each data row represents the data for each object that falls in the 10-largest category for the given object type. The rows are grouped by object type, with a maximum of ten rows for each type (the actual number will be less if fewer objects of the given type existed or had no associated configuration options). The rows are also sorted by size, in descending order, within each group. The header row identifies each column in the data rows, as follows:

- **type**: The type of object, as visible in Genesys Administrator. (The one exception is that a **User** or **Agent** object type is named **Person**.)
- **object_name:** The name of the object as stored in the Configuration Database, enabling a trace of the object in Genesys Administrator.
- **tenant_name:** The name of the Tenant object to which this object belongs. This information is not

provided for Folder object types.

- **object_dbid:** The DBID of the object, enabling a trace of the object in the database and the audit trail.
- **opt_size_chrs:** The total size of all options (including the total number of characters in strings in key-value pairs) for the object.

Sample Output, in CSV format

```
type,object_name,tenant_name,object_dbid,opt_size_chrs
Switch,MMK_FID1_SIPSwitch,Resources,101,120
Switch,MMK_FID2_SIPSwitch,Resources,102,120
Switch,RTP_FID1_SIPSwitch,Resources,103,120
Switch,RTP_FID2_SIPSwitch,Resources,104,120
Person,a274533,Resources,29032,59407
Person,a270369,Resources,29256,54627
Person,a360983,Resources,28542,54335
Person,a545582,Resources,28380,50052
Person,a335660,Resources,28526,49707
Person,a505259,Resources,28486,49111
Person,a023647,Resources,29113,48669
Person,a395158,Resources,28501,48259
Person,a536690,Resources,28379,45615
Person,a495796,Resources,29150,44869
```

csv_biggerthan_objbytype.sql

This script generates a drill down report in which you can identify the configuration objects per object type that impact performance the most, and/or are out of the range of expected size for this type. Audit logs can be used to trace details of a particular change.

This script produces CSV-formatted output on the console if run in the DBMS utility, or in the **csv_biggerthan_objbytype_<DBMS_TYPE>.csv** file if run from the command-line. It provides data for each object where the total size of the configuration options is greater than a threshold (currently 20,000 characters or bytes). The output contains one header row, and a variable number of data rows. The number of data rows can be large (providing data for all objects of every major type, if the threshold is set too low). Rows are grouped by object type and ordered by the size of options, in descending order, within each type. The header row identifies each column in the data rows, as follows:

- **type**: The type of object, as visible in Genesys Administrator. (The one exception is that a User or Agent object type is named Person.)
- **tenant_name:** The name of the Tenant object to which this object belongs. This information is not provided for Folder object types.
- **object_dbid:** The DBID of the object, enabling a trace of the object in the database and the audit trail.
- **opt_size_chrs:** The total size of all options (including the total number of characters in strings in key-value pairs) for the object.

Sample Output, in CSV format)

```
type,object_name,tenant_name,object_dbid,opt_size_chrs
Person,a274533,Resources,29032,59407
```

```
Person,a270369,Resources,29256,54627
Person,a360983,Resources,28542,54335
Person,a545582,Resources,28380,50052
Person,a335660,Resources,28526,49707
....
```

Monitoring Health of Configuration Environment

The health of the Configuration Database requires that you verify several key tables, and identify any typical issues that might lead to inconsistent data. The health monitoring scripts discussed in this section enable you to do this. In most cases, you can use the reports created by these scripts to create new scripts that will correct the problems.

All health monitoring scripts support Configuration Server Database Schema 8.1 or newer.

Running the Health Monitoring Scripts

The scripts can be executed using DBMS utilities or from the command line.

> **Important**
>
> Genesys recommends that you run the **search_orphan_flex_prop.sql** script from the command line. When this script is run using DBMS utilities, the output is inconsistent and does not always provide the information requested.

Using DBMS Utilities

Run the script using the corresponding DBMS utility, as follows:

- MSSQL scripts: Microsoft SQL Server Management Studio 2005 or newer
- Oracle scripts: Oracle SQL Developer 1.5 or newer
- PostgreSQL scripts: pgAdmin III 1.12 that supports Postgre 9.0 or newer

For all scripts except **search_orphan_flex_prop.sql**, the script output appears on the console, which you can capture and save as a **.csv** file, suitable for further analysis. If the output contains no results, your database does not contain any records with the specific problem.

From the Command Line

The Configuration Database Maintenance Scripts IP also contains command-line scripts for Windows (MS SQL, Oracle, and Postgre databases) and Linux (Oracle and Postgre databases) operating systems. Each command-line script runs one of the SQL scripts, and writes the output to a **.csv** file in a specified results directory. This directory must exist before running the script.

Each command-line script generates one **.csv** report file, called **csv_<health_script_name>_<DBMS_TYPE>.csv**, where

- <health_script_name> is the name of the script being run
- <DBMS_TYPE> is the name of the DBMS to which the script is to be applied

The location of the command-line scripts are listed in the following table.

DBMS	Operating System	
Windows	**Linux**	
Windows MS SQL Server	\mssql\ search_duplicates_flex_prop_MSSQL_DB.bat \mssql\search_orphan_flex_MSSQL_DB.bat \mssql\ check_cfg_max_dbid_MSSQL_DB.bat	Not available
Oracle	\oracle\ search_duplicates_flex_prop_ORACLE_DB.bat \oracle\ search_orphan_flex_ORACLE_DB.bat \oracle\ check_cfg_max_dbid_ORACLE_DB.bat	/oracle/ search_duplicates_flex_prop_ORACLE_DB.sh /oracle/ search_orphan_flex_ORACLE_DB.sh /oracle/ check_cfg_max_dbid_ORACLE_DB.sh
Postgre [a]	\postgre\ search_backslash_POSTGRE_DB.bat \postgre\ search_duplicates_flex_prop_POSTGRE_DB.bat \postgre\ search_orphan_flex_POSTGRE_DB.bat \postgre\ check_cfg_max_dbid_POSTGRE_DB.bat	/postgre/ search_backslash_POSTGRE_DB.sh /postgre/ search_duplicates_flex_prop_POSTGRE_DB.sh /postgre/ search_orphan_flex_POSTGRE_DB.sh /postgre/ check_cfg_max_dbid_POSTGRE_DB.sh

[a] For Postgre scripts, configure the PostgreSQL server to allow a client connection with the host on which the scripts are run. For more information, see https://www.postgresql.org/docs/9.3/static/auth-pg-hba-conf.html.

Command Syntax

Run the script from the folder in which it is located, using the following syntax:

```
<script> <USER> < PASSWORD> [<HOST> <PORT> or <HOST[\INSTANCE]>] [<SERVICE_NAME or DB_NAME>] <CSV _RESULT_DIR>.
```

where:

- <script> is the name of the script, as provided in the table above.
- <USER> is the name of a user authorized to access the database.
- <PASSWORD> is the password required for the authorized user to access the database.
- <HOST> is the name of the host on which the database is located.
- <PORT> is the number of the port on that host that is used by the DBMS.
- <HOST\INSTANCE> is the database host and instance, if not the default.
- <SERVICE_NAME> is the Oracle Instance Service Name, that is, the SERVICE_NAME value in **TNSNAMES.ORA**.
- <DB_NAME> is the name of the Database.
- <CSV_RESULT_DIR> is the path to the directory in which the output **.csv** files will be stored.

> **Tip**
> To see the syntax on-line, enter the script name with no parameters and press ENTER.

Examples

MS SQL Server

```
.\search_duplicates_flex_prop_MSSQL_DB.bat sa password localhost\SQLEXPRESS TestDB D:\MF_mon\results\mssql
.\search_duplicates_flex_prop_MSSQL_DB.bat sa password somehost SomeTestDB D:\MF_mon\results\mssql
```

Oracle

```
./search_orphan_flex_ORACLE_DB.sh sourcedb sourcedb localhost 1521 orcl.genesys.com '/home/genesys/results/oracle'
```

Postgre

```
./search_backslash_POSTGRE_DB.sh postgres password localhost 5432 TestDB '/home/genesys/results/postgre'
```

csv_search_backslash.sql

This script runs only to PostgreSQL databases, and retrieves a list of configuration options, including those set in the Annex, for which the configured value contains two or more sequential backslash characters (\\). This script creates a drill-down report that enables you to determine if any particular object of a given type is configured with an option or Annex value of two or more sequential backslash characters.

When using a PostgreSQL database, if the PostgreSQL database setting **standard_conforming_strings** is not set correctly, any database entries containing backslashes have those backslashes doubled, and data might be lost as a result. Typically, this problem affects Application options and/or flexible properties. This script helps you identify if **standard_conforming_strings** is set incorrectly, and what records it affects.

If the number of records in the report increases significantly immediately after a Configuration Server restart, you can also use this report to determine if the sudden increase is caused by the incorrect setting of **standard_conforming_strings**.

If there are no records in the report, your database does not contain any records with the specific problem.

This script produces CSV-formatted output on the console. There is one header row, and any number of data rows. The number of data rows is variable and can be large. The data rows are grouped by type and then by parent object DBID within each type. The header row identifies each column in the data rows, as follows:

- **`type`:** The type of object, as visible in Genesys Administrator. The one exception is that a **User** or **Agent** object type is named **Person**.
- **`object_name`:** The name of the parent object.
- **`tenant_name`:** The name of the Tenant object to which this object belongs. This information is not provided for Folder object types.
- **`object_dbid`:** The DBID of the object, enabling a trace of the object in the database and the audit trail.
- **`option_type`:** The type of configuration option, based on where it is set, **Option** or **Annex**.
- **`section_name`:** The name of the configuration option section in which the the option is set.
- **`option_name`:** The name of the option with a value containing two or more sequential backslashes.
- **`option_value`:** The option value that contains two or more sequential backslashes.

Note that if the value of an option contains one or more comma (`,`) characters, the **`option_value`** string in the CSV output might be formatted incorrectly or truncated.

Sample printout, in CSV format:

```
type,object_name,tenant_name,object_dbid,option_type,section_name,option_name,option_value
Application,confserv,Environment,99,Option,backslashOPt,backlashopt,^[\(\)\-\.\+\d\s\*#]*[\d]+[\(\)\-\.\+\d\s\*#/]*$
Application,confserv,Environment,99,Annex,backslashAnnex,backlashopt,^[\(\)\-\.\+\d\s\*#]*[\d]+[\(\)\-\.\+\d\s\*#/]*$
```

Supported DBMS

- Oracle 11g and newer, can be launched using Oracle SQL Developer 1.5 or newer
- MSSQL 2005 and newer
- PostgreSQL 9.0 and newer

csv_search_duplicates_flex_prop.sql

This script retrieves, for all object types, a list of Annex records that have the same name and parent object, but different DBIDs. You can use this drill-down report to determine if any particular objects of a given type have two or more duplicate Annex records. When duplicate Annex records exist, the record with the lowest DBID is masked by duplicate records with higher DBIDs. When the Annex option with the highest DBID is deleted by Configuration Server, the masked option with the next highest DBID becomes active in the Annex after Configuration Server is restarted. If the whole section is deleted, all of the masked duplicate Annex records with lower DBIDs are orphaned.

This report enables you to determine if database integrity was corrupted. If the result of this script is an empty list, your database does not contain any records with the specific problem.

This script produces CSV-formatted output on the console. There is one header row plus data rows. The number of rows is variable per report and can grow large. Rows are grouped by object type and ordered by parent object DBID within each type. The header row defines the columns of a data row, as follows:

- **`prop_dbid`:** The DBID of the property that has been duplicated.
- **`object_dbid`:** The DBID of the object, enabling a trace of the object in the database and the audit trail.
- **`object_type`:** The type of object, as visible in Genesys Administrator. The one exception is that a **User** or **Agent** object type is named **Person**.
- **`object_name`:** The name of the parent object.
- **`object_tenant_name`:** The name of the Tenant object to which this object belongs. This information is not provided for Folder object types.
- **`section_name`:** The name of the configuration option section in which the option is set.
- **`prop_name`:** The name of the duplicated property.
- **`prop_value`:** The value of the duplicated property.
- **`prop_part`:** The part number for the property, if the property has long multi-part values.

Sample printout, in CSV format

```
prop_dbid,object_dbid,object_type,object_name,object_tenant_name,section_name,prop_name,prop_value,prop_part
10000140,101,Switch,A0,Environment,A,a,aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa,1
10000140,101,Switch,A0,Environment,A,a,aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaabbbbbbb,1
10000139,101,Switch,A0,Environment,,A,,0
140,101,Switch,A0,Environment,A,a,aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa,1
140,101,Switch,A0,Environment,A,a,aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaabbbbbbb.1
139,101,Switch,A0,Environment,,A,,0
```

Supported DBMS

- Oracle 11g and newer; can be launched using Oracle SQL Developer 1.5 or newer
- MSSQL 2005 and newer
- PostgreSQL 9.0 and newer

search_orphan_flex_prop.sql

This script reports the DBID of any Annex options for which their parent DBID is a section that does not exist. In other words, the option is an orphan. If the result of this script is an empty list, your database does not contain any orphan records.

This script produces CSV-formatted output to the report file. There is one header row and any number of data rows, one for each orphan Annex records for which their parent DBID is of a section that does not exist. The number of rows is variable per report and can grow large. Rows are ordered by the Annex option DBID number. The header row defines the columns of a data row, as follows:

- **dbid:** The DBID of the Annex option.
- **object_dbid:** The DBID of the parent option.
- **object_type:** The numeric identifier of the object type.
- **parent_dbid:** The DBID of the section in which the option is configured.
- **prop_name:** The name of the property.
- **prop_value:** The value of the property.

Sample printout in CSV format

```
dbid,object_dbid,object_type,parent_dbid,prop_name,prop_value
208096,1123,12,208070,70921417,282254797065223d2253657
208128,1963,12,208118,328025376,282254797065223d22536572
237339,1140,16,236807,English,MH
237340,1140,16,236807,Bonds,HH
237341,1140,16,236807,Stocks,HH
442818,2718,16,442817,*warm,8667764793
477674,2718,16,442817,exit_composite,8667764793
782438,104,43,782437,value,
782440,104,43,782439,value,1
782446,104,43,782445,value,1
782448,104,43,782447,value,1
782450,104,43,782449,value,1
```

Supported DBMS:

- Oracle 11g and newer
- MSSQL 2005 and newer
- PostgreSQL 9.0 and newer

csv_check_cfg_max_dbid.sql

This script calculates metrics for each object type, all <object-type> Group types, and the history log, as follows:

- The number of objects of this type in the Configuration Database.
- The maximum value of DBID assigned to objects or records of the type or log.
- The value of MAX_DBID for that object type in the CFG_MAX_DBID table, or, for log records, the notify_id field in the CFG_REFRESH table.

The script also indicates if there are any discrepancies between the second and third values.

This script produces CSV-formatted output on the console. There are 41 data rows—one for the history log, one for all <object-type> Group types, and one for each of the 39 object types—and a header row. Each data row consists of information for the respective log, type Groups, or type. The header row defines each column of a data row, as follows:

- **`object_type`:** The type of object, as visible in Genesys Administrator with three exceptions—a User or Agent object type is labelled `Person` type, the History Log is labelled `cfg_hdb_object` type, and the cumulative data for all <object-type> Group types is labelled `All Groups` type).
- **`object_count`:** The number of objects of that type in the Configuration Database.
- **`object_max`:** The maximum value of DBIDs assigned to objects of this type, based on the maximum DBID value in the type's objects table.
- **`cfg_max`:** The value of `max_dbib` in the **cfg_max_dbid** table for the `object_type`.
- **`test`:** Any discrepancies; `PASS` if `cfg_max >= object_max` and `FAILED` if `cfg_max < object_max`.

Sample printout in CSV format

```
object_type,object_count,object_max,cfg_max,test
History log,151,194,194,PASS
All Groups,9,109,109,PASS
Switch,101,101,101,PASS
DN,101,101,101,PASS
......
Role,101,101,101,PASS
```

Supported DBMS

- Oracle 11g and newer

- MSSQL 2005 and newer
- PostgreSQL 9.0 and newer

Configuration Server Prometheus support

Configuration Server (CS) exposes the statistics of its internal operation via HTTP interface as a Prometheus metrics endpoint.

If configured, upon startup, CS opens a specified port for the HTTP GET requests from the Prometheus and/or web browser.

Both http and https protocols are supported.

GET requests for `/metrics` URI path are responded with content intended for Prometheus consumption. The responses contain text in Prometheus text-based format.

Requests for paths other than `/metrics` are responded with http error `404 Not Found`.

Exposed metrics correspond to the statistical data currently exposed via the **stat.log** files.

To know about feature configuration and information on configuration options, refer to [prom] Section in the *Management Framework Configuration Options Reference Manual*.

Sample configurations

http

```
[prom]
port=80

confserv ... -prom-port 80
```

https - Windows

```
[prom]
port=443
transport=tls=1;certificate=91a15b56b95c82214cfb9a9ecd3590a874c9f619

confserv ... -prom-port 443 -prom-transport
tls=1;certificate=91a15b56b95c82214cfb9a9ecd3590a874c9f619
```

Exposed metrics

Metric names are prefixed with - `cs_` for normal operation or with - `tenant_cs_` if tenant_id is specified in devops\contactcenterid annex entry of the CS's configuration object.

In the later case, the value **tenant_id** is added as an index for each metrics.

Metrics with "`current`" following the prefix represent values since previous scrape. These values are reset at each scrape (http request). Prometheus scrapes are distinguished from the requests from

other clients (i.e. browser), so the later do not interfere with the data collected by regular Prometheus scrapes.

Metrics representing data other than countables, Boolean or enumerators are suffixed with units in which the values are given.

Counters are additionally suffixed with “`total`”.

CS general metrics

cs_running_time_milliseconds_total	counter	
cs_current_time_since_last_scrape_milliseconds	gauge	
cs_current_mode	gauge	0 - Unknown 1 - Master Primary 2 - Master Primary Readonly 3 - Restricted Readonly 4 - Master Backup 5 - Proxy Primary 6 - Proxy Backup 7 - Offline 8 - Restricted Import
cs_main_db_connection_status	gauge	0 – down 1 - up
cs_db_sync_connection_status	gauge	0 – down 1 - up
cs_ha_peer_connection_status	gauge	0 – down 1 - up
cs_db_sync_violation_status	gauge	0 – normal operation 1 – db sync violation detected
cs_master_connection_status		0 – down 1 – up -- for proxies only
cs_persistent_mode		0 – normal operation 1 – persistent mode operation -- for proxies only
cs_average_load_percent	gauge	
cs_processing_time_milliseconds_total	counter	
cs_max_atomic_processing_time_milliseconds	gauge	
cs_current_load_percent	gauge	
cs_current_processing_time_milliseconds	gauge	
cs_current_max_atomic_processing_time_milliseconds	gauge	

Cfglib protocol metrics

cs_cfglib_number_of_clients	gauge
cs_cfglib_number_of_gui_clients	gauge
cs_cfglib_requests_received_total	counter
cs_cfglib_notifications_sent_total	counter
cs_cfglib_packets_received_total	counter

cs_cfglib_packets_sent_total	counter
cs_cfglib_received_bytes_total	counter
cs_cfglib_sent_bytes_total	counter
cs_cfglib_processing_time_milliseconds_total	counter
cs_cfglib_max_atomic_processing_time_milliseconds	gauge
cs_cfglib_current_requests_received	gauge
cs_cfglib_current_notifications_sent	gauge
cs_cfglib_current_packets_received	gauge
cs_cfglib_current_packets_sent	gauge
cs_cfglib_current_received_bytes	gauge
cs_cfglib_current_sent_bytes	gauge
cs_cfglib_current_processing_time_milliseconds	gauge
cs_cfglib_current_max_atomic_processing_time_milliseconds	gauge

Per cfglib client metrics

The following metrics are reported for each cfglib client connected to CS at the time of the scrape.

Metrics for each client are indexed with the following:

- `client_session_id` – CS unique (per CS instance run) session ID
- `socket` – socket number
- `app_type` – configuration application type
- `app_name` - configuration application name
- `account` – user account used for login
- `host` – client's host ip address
- `connected` – timestamp of the client login

cs_client_requests_received_total	counter
cs_client_notifications_sent_total	counter
cs_client_packets_received_total	counter
cs_client_packets_sent_total	counter
cs_client_received_bytes_total	counter
cs_client_sent_bytes_total	counter
cs_client_processing_time_milliseconds_total	counter
cs_client_max_atomic_processing_time_milliseconds	gauge
cs_client_current_requests_received	gauge
cs_client_current_notifications_sent	gauge

cs_client_current_packets_received	gauge
cs_client_current_packets_sent	gauge
cs_client_current_received_bytes	gauge
cs_client_current_sent_bytes	gauge
cs_client_current_processing_time_milliseconds	gauge
cs_client_current_max_atomic_processing_time_milliseconds	gauge

Client deferral metrics

cs_deferred_clients	gauge

The following metrics are reported for each client deferred at the time of the scrape.

Metrics for each client are indexed the same way as in the connected clients list.

cs_client_deferred_requests	gauge
cs_client_defer_counter	gauge

Ldap metrics

cs_ldap_requests_queue_size	gauge
cs_ldap_responses_queue_size	gauge

Ldap server failures metrics

For each failing ldap server, the following metrics is reported.

Metrics are indexed with: `search_base` - the ldap server's search base.

Ldap server is no longer considered failing once it has recovered and responded successfully for at least one auth request.

cs_ldap_server_request_failures_total	count

Sample output

```
# TYPE cs_running_time_milliseconds_total counter
cs_running_time_milliseconds_total 25198 1660751626724

# TYPE cs_current_time_since_last_scrape_milliseconds gauge
cs_current_time_since_last_scrape_milliseconds 25198 1660751626724
```

```
# HELP cs_current_mode 0 - Unknown  1 - Master Primary  2 - Master Primary Readonly  3 -
Restricted Readonly  4 - Master Backup  5 - Proxy Primary  6 - Proxy Backup  7 - Offline  8 -
Restricted Import
# TYPE cs_current_mode gauge
cs_current_mode 1 1660751626724

# TYPE cs_main_db_connection_status gauge
cs_main_db_connection_status 1 1660751626724

# TYPE cs_db_sync_connection_status gauge
cs_db_sync_connection_status 1 1660751626724

# TYPE cs_ha_peer_connection_status gauge
cs_ha_peer_connection_status 0 1660751626724

# TYPE cs_db_sync_violation_status gauge
cs_db_sync_violation_status 0 1660751626724

# TYPE cs_average_load_percent gauge
cs_average_load_percent 17 1660751626724

# TYPE cs_processing_time_milliseconds_total counter
cs_processing_time_milliseconds_total 4303 1660751626724

# TYPE cs_max_atomic_processing_time_milliseconds gauge
cs_max_atomic_processing_time_milliseconds 1503 1660751626724

# TYPE cs_current_load_percent gauge
cs_current_load_percent 17 1660751626724

# TYPE cs_current_processing_time_milliseconds gauge
cs_current_processing_time_milliseconds 4303 1660751626724

# TYPE cs_current_max_atomic_processing_time_milliseconds gauge
cs_current_max_atomic_processing_time_milliseconds 1503 1660751626724

# TYPE cs_cfglib_number_of_clients gauge
cs_cfglib_number_of_clients 1 1660751626724

# TYPE cs_cfglib_number_of_gui_clients gauge
cs_cfglib_number_of_gui_clients 1 1660751626724

# TYPE cs_cfglib_requests_received_total counter
cs_cfglib_requests_received_total 22 1660751626724

# TYPE cs_cfglib_notifications_sent_total counter
cs_cfglib_notifications_sent_total 1 1660751626724

# TYPE cs_cfglib_packets_received_total counter
cs_cfglib_packets_received_total 22 1660751626724

# TYPE cs_cfglib_packets_sent_total counter
cs_cfglib_packets_sent_total 33 1660751626724

# TYPE cs_cfglib_received_bytes_total counter
cs_cfglib_received_bytes_total 1210 1660751626724

# TYPE cs_cfglib_sent_bytes_total counter
cs_cfglib_sent_bytes_total 135722 1660751626724

# TYPE cs_cfglib_processing_time_milliseconds_total counter
cs_cfglib_processing_time_milliseconds_total 859 1660751626724
```

```
# TYPE cs_cfglib_max_atomic_processing_time_milliseconds gauge
cs_cfglib_max_atomic_processing_time_milliseconds 101 1660751626724

# TYPE cs_cfglib_current_requests_received gauge
cs_cfglib_current_requests_received 22 1660751626724

# TYPE cs_cfglib_current_notifications_sent gauge
cs_cfglib_current_notifications_sent 1 1660751626724

# TYPE cs_cfglib_current_packets_received gauge
cs_cfglib_current_packets_received 22 1660751626724

# TYPE cs_cfglib_current_packets_sent gauge
cs_cfglib_current_packets_sent 33 1660751626724

# TYPE cs_cfglib_current_received_bytes gauge
cs_cfglib_current_received_bytes 1210 1660751626724

# TYPE cs_cfglib_current_sent_bytes gauge
cs_cfglib_current_sent_bytes 135722 1660751626724

# TYPE cs_cfglib_current_processing_time_milliseconds gauge
cs_cfglib_current_processing_time_milliseconds 859 1660751626724

# TYPE cs_cfglib_current_max_atomic_processing_time_milliseconds gauge
cs_cfglib_current_max_atomic_processing_time_milliseconds 101 1660751626724

# TYPE cs_client_requests_received_total counter
cs_client_requests_received_total{client_session_id="1",socket="1956",app_type="SCE",app_name="default",account=
21 1660751626724

# TYPE cs_client_notifications_sent_total counter
cs_client_notifications_sent_total{client_session_id="1",socket="1956",app_type="SCE",app_name="default",accoun
1 1660751626724

# TYPE cs_client_packets_received_total counter
cs_client_packets_received_total{client_session_id="1",socket="1956",app_type="SCE",app_name="default",account=
21 1660751626724

# TYPE cs_client_packets_sent_total counter
cs_client_packets_sent_total{client_session_id="1",socket="1956",app_type="SCE",app_name="default",account="def
32 1660751626724

# TYPE cs_client_received_bytes_total counter
cs_client_received_bytes_total{client_session_id="1",socket="1956",app_type="SCE",app_name="default",account="d
1050 1660751626724

# TYPE cs_client_sent_bytes_total counter
cs_client_sent_bytes_total{client_session_id="1",socket="1956",app_type="SCE",app_name="default",account="defau
135656 1660751626724

# TYPE cs_client_processing_time_milliseconds_total counter
cs_client_processing_time_milliseconds_total{client_session_id="1",socket="1956",app_type="SCE",app_name="defau
788 1660751626724

# TYPE cs_client_max_atomic_processing_time_milliseconds gauge
cs_client_max_atomic_processing_time_milliseconds{client_session_id="1",socket="1956",app_type="SCE",app_name="
101 1660751626724

# TYPE cs_client_current_requests_received gauge
cs_client_current_requests_received{client_session_id="1",socket="1956",app_type="SCE",app_name="default",accou
21 1660751626724
```

```
# TYPE cs_client_current_notifications_sent gauge
cs_client_current_notifications_sent{client_session_id="1",socket="1956",app_type="SCE",app_name="default",acco
1 1660751626724

# TYPE cs_client_current_packets_received gauge
cs_client_current_packets_received{client_session_id="1",socket="1956",app_type="SCE",app_name="default",accoun
21 1660751626724

# TYPE cs_client_current_packets_sent gauge
cs_client_current_packets_sent{client_session_id="1",socket="1956",app_type="SCE",app_name="default",account="d
32 1660751626724

# TYPE cs_client_current_received_bytes gauge
cs_client_current_received_bytes{client_session_id="1",socket="1956",app_type="SCE",app_name="default",account=
1050 1660751626724

# TYPE cs_client_current_sent_bytes gauge
cs_client_current_sent_bytes{client_session_id="1",socket="1956",app_type="SCE",app_name="default",account="def
135656 1660751626724

# TYPE cs_client_current_processing_time_milliseconds gauge
cs_client_current_processing_time_milliseconds{client_session_id="1",socket="1956",app_type="SCE",app_name="def
788 1660751626724

# TYPE cs_client_current_max_atomic_processing_time_milliseconds gauge
cs_client_current_max_atomic_processing_time_milliseconds{client_session_id="1",socket="1956",app_type="SCE",ap
101 1660751626724

# TYPE cs_deferred_clients gauge
cs_deferred_clients 0 1660751626724

# TYPE cs_ldap_requests_queue_size gauge
cs_ldap_requests_queue_size 0 1660751626724

# TYPE cs_ldap_responses_queue_size gauge
cs_ldap_responses_queue_size 0 1660751626724
```

Statistical logging

This section describes the 'CS clients Monitoring' feature in CS.

CS now optionally collects certain statistics on its operation and periodically prints it out to a dedicated log file. Statistics is collected on application, reactor (currently only CfgLib) and client levels.

For each item, the following are calculated:

- total for the entire run
- for the period since last printout

Statistics log file uses the following line format for statistics output:

```
<Timestamp>  <Parameter>=<Value> [, <Parameter>=<Value>...]
```

The following data is currently collected and printed out:

Application level

- Running time
- Current mode
- Load - percentage of cpu core running CS code on the main thread vs total elapsed time; this does not include low-level commonlib code, but includes requests processing (outside conn_multi_scan()), commonlib and dblib callbacks
- Proc time - total high-level processing time - CS code on the main thread
- Proc max - max. atomic operation processing time

Reactor level

Cfglib reactor only, soap reactor is currently not included.

- Clients - current number of connected clients
- GUI - current number of connected GUI clients

These are totals for all clients, including those which disconnected during the reporting period - total since CS startup and since last printout:Requests - number of received requests

- Notificats- number of received notifications
- Pckts rcvd - number of received "packets" (that is, calls to conn_write())
- Pckts sent - number of sent "packets"
- Bytes rcvd
- Bytes sent
- Proc time - total processing time
- Proc max - max atomic operation processing time

Client level

Lists all clients connected at the time of printout:

- Sock
- App Type
- App Name
- User
- Host
- Connected - timestamp
- (+ same stats as for reactor - for that client)

The following is sample content printed in logs every x seconds.

```
2018-10-10T12:00:01.001 Running Time = 2.17:01:58.277
```

```
2018-10-10T12:00:01.001 Current Mode = Primary

2018-10-10T12:00:01.001 Load = 1%

2018-10-10T12:00:01.001 Proc time = 3279447

2018-10-10T12:00:01.001 Proc Max = 3467

....

2018-10-10T12:00:01.001 socket = 540, AppType = SCE, App Name = default, User = default, Host
= 192.168.158.1, Connected = 2018-08-10T18:46:31.688,.....

2018-10-10T12:00:01.001 socket = 588, AppType = SCS, App Name = SCS, User = SYSTEM, Host =
192.168.158.1, Connected = 2018-08-10T18:46:32.250,.....
```

Transaction Serialization

Configuration Server can process data change (write) requests so that concurrent data changes are processed correctly and do not interfere with each other. Transaction serialization prevents data change transactions from overlapping and potentially causing a loss of data integrity. It involves the deferral of transaction data change requests so that each request can be processed completely, without impacting, or being impacted by, other requests.

> **Important**
>
> Transaction Serialization might be required when a non-Genesys application has been used to modify the configuration.

This section describes several ways that you can use transaction serialization to preserve data integrity without jeopardizing system performance. You must enable transaction serialization before you can configure any of the features described in this section.

Overview

Without transaction serialization, Configuration Server starts processing data change requests as soon as they arrive, regardless of any other requests that are being processed. If the first request being processed changes something upon which another request being processed depends, problems can include corrupt data, application faults, or even system shutdown.

Transaction serialization solves this problem by ensuring that only one data change request per session is being processed at a time. Configuration Server can accept new requests while processing a request, but does not start the next one until the first request is finished. Data write and data read requests are stored in the same queue and processed in FIFO (first-in, first-out) order. Data read requests from a client whose previous write requests are deferred must also wait for the preceding request to finish before being processed. This ensures that a client can view updated data from the database.

There is one queue for each client making a request. Configuration Server and Configuration Server Proxy clients are treated the same, so there is no potential for overlap and interference between clients.

Connect/Authorize requests are an exception. They are not considered transaction requests and Configuration Server processes them immediately. Internal requests are not meant to be deferred, even those that generate data change. They are processed immediately, but the change notification sent to the clients is deferred. Some of the internal requests that do not get deferred are:

- Updates to last-login and last-login synchronization information
- Password changes

- Forced password changes as a result of the Change password on next login feature

Enabling Transaction Serialization

To enable transaction serialization, set the option serialize-write-transactions in the **[system]** section of the Configuration Server Application object to `true`. If this option is not configured or is set to `false`, requests are not deferred.

> ### Warning
> Use extreme caution if you decide to disable transaction serialization after having first enabled it. Doing so will release all deferred requests and Configuration Server might become overwhelmed. Genesys recommends that you not disable this feature unless asked to.

Expiration of Deferred Requests

> ### Important
> Transaction serialization must be enabled (serialize-write-transactions is set to `true`) to use any of the options in this section.

Requests are classified into three categories: general clients' requests; requests from Configuration Server Proxies reconnecting to the master Configuration Server; and internal requests, generated by Configuration Server itself when processing other requests. Expiration and handling is different for each type, as described later in this section.

Deferred requests from disconnected clients are removed from the queue. If the client disconnected from the master Configuration Server, a response is sent immediately that the client has disconnected. General and Configuration Server Proxy requests are discarded. Internal requests are processed.

General Requests

To specify the time interval after which general requests expire, set deferred-requests-expiration to the desired time interval, in seconds. A value of `0` (zero) means that the requests never expire.

If a deferred general request has not been processed during the specified time interval, it expires. The request is removed from the queue, and the requesting client notified with the following message

```
ERROR CFGReadOnlyOperationalModeActivated/Transaction expired
```

Deferred non-transaction requests, such as read requests, that expire are removed from the queue and processed as usual.

Requests from Configuration Server Proxies in Process of Reconnecting

Requests coming from Configuration Server Proxies in the process of reconnecting have much higher expiration time, determined automatically, based on the value of proxy-load-timeout.

If the deferred request from a reconnecting Configuration Server Proxy expires, the proxy server is disconnected, and must reconnect to the master Configuration Server from the start.

Expiration of Internal Requests from Configuration Server

Internal requests are password change requests triggered by the Change Password On Next Login feature, or an overridden account expiration because of login and last-login notifications. These requests never expire, and are processed even if the corresponding clients disconnected while the requests were deferred.

Transaction Throttling

Transaction throttling addresses the problem of poor system performance because of, for example, an accumulation of data change notifications generated by massive data updates in the outgoing data buffers of Configuration Server and/or in the incoming data buffers of the clients, or a low bandwidth connection between Configuration Server and its clients (including Configuration Server Proxy). As a result, there was sometimes a significant delay in notification processing, excessive memory usage, and ultimately disconnections due to slow client responses.

When the throttling feature is enabled, unprocessed transaction requests are deferred in the same way as they are during **transaction serialization**. Processing of the deferred requests occurs in the same manner (that is FIFO, complete one request before starting the next). The functionality is invisible except for some delay due to the deferral. Data read requests from other clients who do not submit transaction requests, remain unaffected in both the timing of the processing of their requests and their processing order.

Implementation

To enable transaction throttling, do the following options in the **[system]** section of the Configuration Server Application:

1. Ensure that transaction serialization is enabled (serialize-write-transactions is set to `true`). If transaction serialization is disabled, transaction throttling cannot be used.
2. Set throttle-updates-interval in the **[system]** section of the Configuration Server Application to the desired time interval.

> **Tip**
> When configuring this option, keep in mind that if actual load consistently exceeds the rate specified by this option for a significant time, deferred unprocessed requests will accumulate in the input queue and will be eventually cancelled as defined by the value of the deferred-requests-expiration option. To avoid this happening, consider

> adjusting throttle-updates-interval accordingly, to account for the expected actual load.

Transaction Deferral for Primary or Backup Configuration Server Proxy Loading and Reloading Data

This feature defers all transaction data requests whenever a primary or backup Configuration Server Proxy is loading (or reloading) data from the master Configuration Server. This ensures that the notifications related to the load request are sent only after the full load is completed. Otherwise, because of the size of the data being loaded, notifications might be sent before the loading is complete, potentially harming the data in the proxy server's memory. After the loading is complete, all deferred requests are released (in FIFO order), and processed.

Implementation

This feature is always enabled, so long as transaction serialization is also enabled (serialize-write-transactions=`true`).

MS SQL Cluster Database with MultiSubnet Listener for Framework Applications

This section describes how to configure the MS SQL Always On with MultiSubnet Listener feature for the Configuration Server (for the Configuration Database) and Message Server (for the Centralized Log Database) applications. No configuration is required for DB Server.

For more information about the feature, refer to MS SQL Always On with MultiSubnet Listener in the *Framework Database Connectivity Reference Guide*.

Configuration Server

You must be sure that you are using the correct dbclient. On the Windows platform, you must use the **dbclient_msql** located in the **dbclient_next** folder of the latest IP. On the Linux platform, you can use the default **dbclient_msql**.

Use the **mssql-multisubnet** option to enable the feature. This feature enables multisubnet failover using MS SQL for disaster recovery and business continuity scenarios involving the Configuration Database. For more information about this option, see the Configuration Server Release note for version 8.5.101.22.

In Configuration Server, add **[<Configuration Database section>].mssql-multisubnet=**Yes in the configuration file (**confserv.cfg** or **confserv.conf**), as shown in the following example. Note that you can specify either the listener name or IP address for the **dbserver** parameter.

```
[dbserver]
dbengine = mssql
dbserver = 135.17.38.30
dbname = db01
username =sa
password =Genesys#1
mssql-multisubnet=Yes
```

Message Server

You must be sure that you are using the correct dbclient. On the Windows platform, you must use **dbclient_msql** located in the **dbclient_next** folder of the latest IP. On the Linux platform, you can use the default **dbclient_msql**.

Use the **mssql-multisubnet** option to enable the feature. This feature enables multisubnet failover using MS SQL for disaster recovery and business continuity scenarios involving the Centralized Log Database.

In the Database Access Point (DAP) application connecting Message Server to the Log Database, use Genesys Administrator to set the **[default].mssql-multisubnet** option to Yes.

Refer to Database Access Points in the *Framework Database Connectivity Reference Guide* for instructions about configuring a DAP. For more information about the option, see the Message Server Release note for version 8.5.100.19.

Framework Database Replication for Disaster Recovery

The topics in this section describes Framework database replication for Disaster Recovery / Business Continuity based on the following software:

- Oracle GoldenGate software
- Microsoft SQL AlwaysOn software

For information about Disaster Recovery architecture and how it works, refer to Disaster Recovery/ Business Recovery.

Framework Database Replication for Disaster Recovery Using Oracle GoldenGate

This section describes the configuration of a Disaster Recovery / Business Continuity architecture, as described in Disaster Recovery Architecture. The configuration is based on the Oracle GoldenGate software.

> **Important**
>
> This page is not a step-by-step guide. It is a description of reference architecture; it tells you what components are needed and where. A customer can copy a file (on Linux), create an image of their entire Virtual Machine and clone it (this is the recommended way, keeping in mind the complexity of the Disaster Recovery architecture and the number of components involved), or deploy each additional installation package individually and adjust the resulting configuration (since each installation package can alter aspects of configuration objects, such as changing startup paths).

Operation

System Startup Procedure and Normal Operating Mode

1. Start the replication process.

> **Tip**
>
> For reference, use the Oracle® GoldenGate Windows and UNIX Administrator's Guide 11g Release 1 (11.1.1) E17341-01 (Ref 1.)

a. On the MAIN and SECONDARY systems, run the script that removes INSERT, UPDATE, and DELETE permissions to CFG_DB and MS_DB users.

b. Use the START MANAGER command to start manager processes at both sites.

c. At the SECONDARY site, using GGSCI, start REPLICATs in preparation to receive changes from the Configuration and Log Databases on the live MAIN system.

```
START REPLICAT CSP
START REPLICAT MSP
```

d. At the MAIN site, using GGSCI, start REPLICAT MSS in preparation to receive changes from the log Database on the live SECONDERY system.

```
START REPLICAT MSS
```

e. On the MAIN site system, alter the primary Extract to begin capturing data based on the current timestamp. Otherwise, Extract will spend unnecessary time looking for operations that date back to the time that the group was created using the `ADD EXTRACT` command.

```
ALTER EXTRACT CSP, BEGIN NOW
ALTER EXTRACT MSP, BEGIN NOW
```

f. On the SECONDARY site system, alter the secondary Extract to begin capturing data based on the current timestamp. Otherwise, Extract will spend unnecessary time looking for operations that date back to the time that the group was created with the `ADD EXTRACT` command.

```
ALTER EXTRACT MSS, BEGIN NOW
```

g. On the MAIN system, start the primary Extracts so they are ready to capture transactional changes.

```
START EXTRACT CSP
START EXTRACT MSP
```

h. On the SECONDARY system, start the secondary Extract so it is ready to capture transactional changes at the secondary Log Database.

```
START EXTRACT MSS
```

i. On the MAIN system, do the following:

- Run the script that grants INSERT, UPDATE, and DELETE permissions to CFG_DB and MS_DB users.
- Run the script that enables triggers and cascade delete constraints.

j. On the SECONDARY system, do the following:

- Run the script that grants INSERT, UPDATE, and DELETE permissions to MS_DB users.
- Run the script that enables triggers and cascade delete constraints.

At this point, the database system is ready for normal operation.

2. Start the system.

a. Run the scripts that switchover `cfgmaster` host name IP resolution to a MAIN live system.

b. Launch the MAIN live Master Configuration Server primary/backup pair at Site 1.

c. Launch the MAIN live Master Solution Control Server to control the main Master Configuration Server pair at Site 1.

d. Launch the MAIN Message server at Site 1 to support communications for Solution Control Servers controlling site components.

e. Launch Solution Control Server at Sites 1 and 2.

f. Using Solution Control Server, start the Configuration Server Proxy pair at Sites 1 and 2.

g. Using Solution Control Server, start the Framework site components.

Disaster Recovery Switchover

At this point, all system components residing at the MAIN site on Host 3 are lost and not running:

- MAIN live Master Configuration Server primary/backup pair
- MAIN live Master Solution Control Server
- MAIN Message server at Site 1
- Oracle database
- Oracle GoldenGate

Operations on other sites can be continued non-stop in limited mode without a configuration change using Configuration Server Proxies running in Read-Only mode until the SECONDARY Master Configuration Server is brought on-line.

Perform the following steps to move activity to the SECONDARY live Master Configuration Server primary/backup pair.

1. On the SECONDARY live standby system, using GGSCI, issue the `LAG REPLICAT` command until it returns `At EOF (end of file)` to confirm that REPLICAT applied all of the data from the trail to the database.
 	
`LAG REPLICAT CSP`
`LAG REPLICAT MSP`

2. Stop the REPLICAT processes.
 	
`STOP REPLICAT CSP`
`STOP REPLICAT MSP`

3. On the SECONDARY system, run the script that grants INSERT, UPDATE, and DELETE permissions to the CFG_DB and MS_DB users.
4. Run the script that enables triggers and cascade delete constraints.
5. Launch the SECONDARY live Master Configuration Server primary/backup pair at Site 2.
6. Launch the SECONDARY live Master Solution Control Server to control the MAIN Master Configuration server pair at Site 2.
7. Launch the SECONDARY Message Server at Site 2 to support communication for Solution Control Servers controlling site components.
8. Run the `dnscmd` script that switches over `cfgmaster` host name IP resolution to a MAIN live system.
9. On the host running Configuration Server Proxies, run the switch over script to flush the DNS cache.

Communication Server Proxies reconnect to the SECONDARY live Master Configuration Server primary/backup pair and resume normal operation.

> ### Warning
> Do not start the data EXTRACTOR on the SECONDARY system. The user transactions must accumulate in the SECONDARY system database until the MAIN system is to be

restored. Use the Secondary CSS replication group for database replication from SECONDARY to MAIN system before switching back to the MAIN system.

Configuration Example

Configuration for Oracle GoldenGate Replication Processes

1. At the MAIN and SECONDARY Oracle databases, create a user CFG_DB for the Configuration Server database, and user MS_DB for the Log Message Server database.

2. Using the initialization scripts in the Installation Package, create the database objects for the Configuration and Log Message Server Databases.

3. Use Oracle® GoldenGate Oracle Installation and Setup Guide11g Release 1 (11.1.1) E17799-01 (Ref 1.) and the examples of Parameter files below to configure the EXTRACT and REPLICAT processes.

Process	Table (EXTRACT) / Map (REPLICAT) Parameters
EXTRACT CSP	SEQUENCE CFG_DB.*;
EXTRACT CSS	TABLE CFG_DB.*; TABLEEXCLUDE CFG_DB.cfg_refresh;
EXTRACT MSP	SEQUENCE MS_DB_1.*; TABLE MS_DB_1.*; TABLEEXCLUDE MS_DB_1.G_LOG_SCHEMA_INFO;
EXTRACT MSS	SEQUENCE MS_DB_2.*; TABLE MS_DB_2.*; TABLEEXCLUDE MS_DB_2.G_LOG_SCHEMA_INFO;
REPLICAT CSS	MAP CFG_DB.*,TARGET CFG_DB.*;
REPLICAT CSP	MAPEXCLUDE CFG_DB.cfg_refresh;
REPLICAT MSP	MAP MS_DB_1.*,TARGET MS_DB_1.*; MAPEXCLUDE MS_DB_1.G_LOG_SCHEMA_INFO;
REPLICAT MSS	MAP MS_DB_2.*,TARGET MS_DB_2.*; MAPEXCLUDE MS_DB_2.G_LOG_SCHEMA_INFO;

4. Register Oracle GoldenGate EXTRACT and REPLICAT using GGSCI.

On the MAIN live system:

```
dblogin userid gg_user, password gg_password
register extract CSP, LOGRETENTION
register extract MSP, LOGRETENTION
```

On the SECONDARY live standby system:

```
dblogin userid gg_user, password gg_password
register extract CSS, LOGRETENTION
register extract MSS, LOGRETENTION
```

Extract Group CSP at Primary Site Configuration Example

1. At MAIN system, start GGSCI.
2. Use the ADD EXTRACT command to create an Extract group CSP.

```
ADD EXTRACT CSP , TRANLOG, BEGIN NOW
```

Use TRANLOG as the data source option.

3. Use the ADD RMTTRAIL command to specify a remote trail to be created on the target system.

```
ADD RMTTRAIL ./CS, EXTRACT CSP
```

Use the EXTRACT argument to link this trail to the Extract group.

4. Use the EDIT PARAMS command to create a parameter file for the Extract group. Include the following parameters plus any others that apply to your database environment.

```
EDIT PARAMS CSP
```

CSP EXTRACT Parameters File Example:

```
EXTRACT CSP
RMTHOST <Secondary host name>, MGRPORT 7809
RMTTRAIL ./dirdat/CP
USERID gg_user PASSWORD gg_password
TRACE ./trace/cfg_db.trc
--Only use if DDL is configured
WILDCARDRESOLVE DYNAMIC
DDL INCLUDE MAPPED OBJNAME cfg_db.*
DDLOPTIONS ADDTRANDATA RETRYOP RETRYDELAY 20 MAXRETRIES 60 REPORT
-- TRANLOGOPTIONS DBLOGREADER, DBLOGREADERBUFSIZE 1024000
-- TRANLOGOPTIONS  DBLOGREADERBUFSIZE 1024000

STATOPTIONS RESETREPORTSTATS
REPORT AT 00:01
REPORTROLLOVER AT 00:01
REPORTCOUNT EVERY 60 SECONDS, RATE
--
SEQUENCE cfg_db.*;
TABLE cfg_db.*;
TABLEEXCLUDE cfg_db.cfg_refresh;
```

REPLICAT Group CSP at Secondary Site Configuration Example

1. At the SECONDARY system, start GGSCI.
2. Use the ADD REPLICAT command to create a Replicat group CSP.

```
   ADD REPLICAT CSP, EXTTRAIL CSP, BEGIN NOW
```

Use the EXTTRAIL argument to link the Replicat group to the remote trail.

3. Use the EDIT PARAMS command to create a parameter file for the Replicat group. Include the following parameters plus any others that apply to your database environment:

```
   EDIT PARAMS CSP
```

CSP REPLICAT Parameters File Example:

```
MACRO #exception_handler
BEGIN
, TARGET ggate.exceptions
, COLMAP ( rep_name = "rep"
, table_name = @GETENV ("GGHEADER", "TABLENAME")
, errno = @GETENV ("LASTERR", "DBERRNUM")
, dberrmsg = @GETENV ("LASTERR", "DBERRMSG")
, optype = @GETENV ("LASTERR", "OPTYPE")
, errtype = @GETENV ("LASTERR", "ERRTYPE")
, logrba = @GETENV ("GGHEADER", "LOGRBA")
, logposition = @GETENV ("GGHEADER", "LOGPOSITION")
, committimestamp = @GETENV ("GGHEADER", "COMMITTIMESTAMP"))
, INSERTALLRECORDS
, EXCEPTIONSONLY;
END;
-- This ends the macro
REPLICAT CSP
HANDLECOLLISIONS
--END RUNTIME
USERID gg_user, PASSWORD gg_password
ASSUMETARGETDEFS
DISCARDFILE ./dirrpt/CP.dsc, purge
TRACE ./trace/CSP.trc
-- INCLUDE ALL -- &
-- STATOPTIONS RESETREPORTSTATS
DDL INCLUDE  ALL
--INCLUDE MAPPED  --  &

-- DBOPTIONS SUPPRESSTRIGGERS, DEFERREFCONST
DBOPTIONS  DEFERREFCONST
REPORT AT 00:01
REPORTROLLOVER AT 00:01
REPORTCOUNT EVERY 60 SECONDS, RATE
---
DDLOPTIONS REPORT
DDLERROR DEFAULT IGNORE
REPERROR (DEFAULT, EXCEPTION)
REPERROR (DEFAULT2, ABEND)
REPERROR (-1, EXCEPTION)
MAP CFG_DB.*,TARGET CFG_DB.*;
MAPEXCLUDE CFG_DB.cfg_refresh;
--MAP CFG_DB.* #exception_handler();
```

Configuration of Genesys Components

> **Important**
> LCA must be installed on every host on which Genesys components are running.

1. Start the replication process (P.3.3.1.1).
2. Run the `dnscmd` script that switches over `cfgmaster` host name IP resolution to a MAIN live system.
3. Install the MAIN live Master Configuration Server primary/backup pair at Site 1.
4. Launch the MAIN live Master Configuration Server primary/backup pair at Site 1.
5. Using Genesys Administrator connected to the Primary Master Configuration Server at Site 1, configure Master Solution Control Server, Message Server for distributed SCS, and Backup Master Configuration Server.
6. Install Master Solution Control Server, Message Server for distributed SCS, and Master Backup Configuration Server on Site 1.
7. Copy all backup and primary instances of Master Solution Control Server, Message Server for distributed SCS, and Master Configuration Server working directories to Site 2. **Warning** Never launch any instance of Master Configuration Server at Site 2 if it can access the Configuration Database that is also used by a running Configuration Server at Site 1.
8. Launch Master Solution Control Server and Message Server for distributed SCS.
9. Using Genesys Administrator connected to the Primary Master Configuration Server at Site 1, configure and install Configuration Server Proxies, Solution Control Servers, and Message Servers for network logging for Sites 1 and 2.
10. Start Configuration Server Proxies at Sites 1 and 2.
11. Start Solution Control Servers at Sites 1 and 2.
12. Start Message Servers for network logging at Sites 1 and 2.
13. Install Framework Components at Site 1 using the Configuration Server Proxy host and port at Site 1.
14. Install Framework Components at Site 2 using the Configuration Server Proxy host and port at Site 2.

File and Script Examples

EXTRACT Parameters File

```
EXTRACT <extract_name>
RMTHOST <target database host name>, MGRPORT 7809
RMTTRAIL ./dirdat/<rmttrail_name>
USERID <golden_gate_user> PASSWORD <golden_gate_password>
TRACE ./trace/<oracle_user_name>.trc
--Only use if DDL is configured
WILDCARDRESOLVE DYNAMIC
```

```
DDL INCLUDE MAPPED OBJNAME <oracle_user_name>.*
DDLOPTIONS ADDTRANDATA RETRYOP RETRYDELAY 20 MAXRETRIES 60 REPORT
-- TRANLOGOPTIONS DBLOGREADER, DBLOGREADERBUFSIZE 1024000
-- TRANLOGOPTIONS  DBLOGREADERBUFSIZE 1024000

STATOPTIONS RESETREPORTSTATS
REPORT AT 00:01
REPORTROLLOVER AT 00:01
REPORTCOUNT EVERY 60 SECONDS, RATE
--
SEQUENCE <oracle_user_name>.*;
TABLE <oracle_user_name>.*;
TABLEEXCLUDE <exclude_filter>;
```

REPLICAT Parameters File

```
MACRO #exception_handler
BEGIN
, TARGET ggate.exceptions
, COLMAP ( rep_name = "rep"
, table_name = @GETENV ("GGHEADER", "TABLENAME")
, errno = @GETENV ("LASTERR", "DBERRNUM")
, dberrmsg = @GETENV ("LASTERR", "DBERRMSG")
, optype = @GETENV ("LASTERR", "OPTYPE")
, errtype = @GETENV ("LASTERR", "ERRTYPE")
, logrba = @GETENV ("GGHEADER", "LOGRBA")
, logposition = @GETENV ("GGHEADER", "LOGPOSITION")
, committimestamp = @GETENV ("GGHEADER", "COMMITTIMESTAMP"))
, INSERTALLRECORDS
, EXCEPTIONSONLY;
END;
-- This ends the macro
REPLICAT <replicat_name>
HANDLECOLLISIONS
--END RUNTIME
USERID <golden_gate_user>, PASSWORD <golden_gate_password>
ASSUMETARGETDEFS
DISCARDFILE ./dirrpt/<discard_file_name>.dsc, purge
TRACE ./trace/<traice_file_name>.trc
-- INCLUDE ALL -- &
-- STATOPTIONS RESETREPORTSTATS
DDL INCLUDE  ALL
--INCLUDE MAPPED  --  &

-- DBOPTIONS SUPPRESSTRIGGERS, DEFERREFCONST
DBOPTIONS  DEFERREFCONST
REPORT AT 00:01
REPORTROLLOVER AT 00:01
REPORTCOUNT EVERY 60 SECONDS, RATE
---
DDLOPTIONS REPORT
DDLERROR DEFAULT IGNORE
REPERROR (DEFAULT, EXCEPTION)
REPERROR (DEFAULT2, ABEND)
REPERROR (-1, EXCEPTION)
MAP <oracle_user_name>.*,TARGET <oracle_user_name>.*;
MAPEXCLUDE <exclude_filter>;
--MAP CFG_DB.* #exception_handler();
```

dnscmd Scripts

Switch to SECONDARY Master Server

```
rem DNSCMD DELETE command
dnscmd 135.17.36.102 /RecordDelete mst.lab cfgmaster A /f
rem DNSCMD ADD command
dnscmd 135.17.36.102 /RecordAdd mst.lab cfgmaster A 135.17.36.140
```

Switch to MAIN Master Server

```
rem DNSCMD DELETE command
dnscmd 135.17.36.102 /RecordDelete mst.lab cfgmaster A /f
rem DNSCMD ADD command
dnscmd 135.17.36.102 /RecordAdd mst.lab cfgmaster A 135.17.36.139
```

Switch over Script

```
ipconfig /flushdns
ping cfgmaster.mst.lab
```

Framework Database Replication for Database Recovery Using MS SQL Cluster with AlwaysOn

This section describes the configuration of a Disaster Recovery / Business Continuity architecture, as described in Disaster Recovery / Business Continuity. The configuration is based on the MS Failover Cluster and MS SQL AlwaysOn software.

> **Important**
>
> This page is not a step-by-step guide. It is a description of reference architecture; it tells you what components are needed and where. A customer can copy a file (on Linux), create an image of their entire Virtual Machine and clone it (this is the recommended way, keeping in mind the complexity of the Disaster Recovery architecture and the number of components involved), or deploy each additional installation package individually and adjust the resulting configuration (since each installation package can alter aspects of configuration objects, such as changing startup paths).

Operation

System Startup Procedure and Normal Operating Mode

If Configuration and Log databases have not been added to the AlwaysOn High Availability Group, complete step 2 of the procedure Configuration for MS SQL Replication Processes, “Add Configuration and Log Databases to AlwaysOn High Availability Group”.

1. Check database status.
   - All databases at the primary node have status “Synchronized”.
   - All databases at the secondary node have status “Synchronizing”.
   - The Availability Replicas folder shows status at Node 1 “Primary” and Node 2 “Secondary”.
   - The Availability Databases folder shows the data moving icon as a “green arrow” for all databases.

   At this point, the database system is ready for normal operation.
2. Start the system.

   a. Launch the main live Master Configuration Server primary/backup pair at Site 1.

b. Launch the main live Master Solution Control Server to control the main Master Configuration Server pair at Site 1.

c. Launch the main Message Server at Site 1 to support communications for Solution Control Servers controlling site components.

d. Launch Solution Control Server at Sites 1 and 2.

e. Using Solution Control Server, start the Configuration Server Proxy pair at Sites 1 and 2.

f. Using Solution Control Server, start the Framework site components.

At this point Site 1 and Site 2 systems are ready for normal operation.

Disaster Recovery Switchover

At this point, all system components residing at Host 2 are lost and not running:

- Main live Master Configuration Server primary/backup pair
- Main live Master Solution Control Server
- Main Message Server at Site 1
- MS SQL database

Perform the following steps to move activity to the secondary live Master Configuration Server primary/backup pair.

1. Switch MSSQL replication processes over to the secondary site.

a. Check the Availability Groups status: Availability Replicas Node 1 has the question mark and Node 2 shows status “Resolving”. Databases shows status “Not Synchronizing”.

b. Start the Failover wizard by right-clicking the Availability Group object at Node 2 (Primary) in the Availability Groups folder.

c. Select Node 2 as new Primary replica.

d. Confirm potential data loss.

e. Observe Failover wizard completion results to ensure there are no errors. Check the Availability Replicas Nodes status: Node 2 is Primary and Node 1 is Secondary.

At this point, the Secondary database is ready for operation in failover mode.

2. Start the system in failover mode.

a. Launch the secondary live Master Configuration Server primary/backup pair at Site 2.

b. Launch the secondary live Master Solution Control Server to control the main Master Configuration Server pair at Site 2.

c. Launch the secondary Message Server at Site 2 to support communication for Solution Control Servers controlling site components.

Configuration Server Proxies have reconnected to the secondary live Master Configuration Server primary/backup pair and resume

normal operation.

Switching Back to Normal Operating Mode

1. Stop the secondary Message Server at Site 2.
2. Stop the secondary live Master Solution Control Server at Site 2.
3. Stop the secondary live Master Configuration Server primary/backup pair at Site 2.
4. Back up the Configuration and Log databases at Node 2.
5. Start MS SQL Server at Node 1.
6. Switch the MS SQL replication processes back to the original primary.

   At this point we assume that MS SQL server at Node 1 is restored and running with existing Configuration and Log Message Server databases that have outdated configuration data. If databases are lost, complete the procedure Configuration of MS SQL Replication Processes.

   a. Check if the Availability Replicas folder has listed Node 1 running as Secondary. Configuration and Log databases at Node 2 have “Synchronized” status, and Configuration and Log databases at Node 1 have “Not Synchronizing” status.

   b. Start the Resume Data Movement procedure (right-click and select it from the context menu) for Configuration and Log databases at Node 1.

   c. Wait until data move starts and the databases statuses are changed to “Synchronized” at Node 2 and “Synchronizing” at Node 1. You might have to refresh your display to see that the database status has changed. Then, start the Failover Wizard (right-click the Availability Groups object and select it from the context menu).

   d. Change Availability Mode to “Synchronous” in the Availability Group object property page.

   e. Check if Configuration and Log databases statuses are “Synchronized” at Node 2 and “Synchronizing” at Node 1. Then, start the Failover Wizard.

   f. Connect to the Availability Replica at Node 1.

   g. Ensure that the Summary contains the correct information. Click **Finish** and watch the Progress. Check to ensure there are no errors in the results.

   h. Change the Availability Mode to “Asynchronous” on the Availability Group object property page. Wait until the Configuration database at Primary node 1 has status “Synchronized” and the database at secondary Node 2 has status “Synchronizing”.

   i. Start the system in normal operating mode (Step 1, above).

Configuration Example

At this point we assume that AlwaysOn High Availability Group is configured. AlwaysOn High Availability Group listener has no port range set.

Configuration for MS SQL Replication Processes

1. Using the initialization scripts in the Installation Package, create the database objects for the Configuration and Log Message Server databases at Node 1.

2. Add the Configuration and Log Message Server Databases to the AlwaysOn High Availability Group.

a. Start the Add Database Wizard to add the Configuration Database to the AlwaysOn High Availability Group.

b. Select the Configuration Database from the database list.

c. Select the initial data synchronization mode “Full” and point to the shared location accessible from both nodes.

d. Connect to the second node using the Domain user account.

e. Check if the Validation pass and Summary have accurate information. Click Next on Validation and Summary pages. Check the Results page to ensure there are no errors and the Configuration Database shows the “Restoring…” status.

f. Wait until the Configuration Database at the Primary node shows status “Synchronized” and the database at the secondary node has status “Synchronizing”. You might have to refresh your display to see the change in database status.

g. Repeat Steps a-f for Log Message Server databases for Sites 1 and 2.

Configuration of Genesys Components

1. Install the main live Master Configuration Server primary/backup pair at Site 1.

2. Install the secondary dormant Configuration Server primary/backup pair at Site 2.

3. Launch the main live Master Configuration Server primary/backup pair at Site 1.

4. Using Genesys Administrator connected to the Primary Master Configuration Server at Site 1, configure Master Solution Control Server, Message Server for distributed SCS, and Backup Master Configuration Server.

5. Install Master Solution Control Server, Message Server for distributed SCS, and Master Backup Configuration Server on Site 1.

6. Copy Master Solution Control Server, Message Server for distributed SCS, and Master Backup Configuration Server working directories to Site 2.

7. Launch Master Solution Control Server and Message Server for distributed SCS.

8. Using Genesys Administrator connected to the Primary Master Configuration Server at Site 1, configure and install Configuration Server Proxies, Solution Control Servers, and Message Servers for network logging for Sites 1 and 2.

9. Start Configuration Server Proxies at Sites 1 and 2.

10. Start Solution Control Servers at Sites 1 and 2.

11. Start Message Servers for network logging at Sites 1 and 2.

12. Install Framework Components at Site 1 using the Configuration Server Proxy host and port at Site 1.

13. Install Framework Components at Site 2 using the Configuration Server Proxy host and port at Site 2.

Internet Protocol version 6 (IPv6)

IPv6 is a network layer for packet-switched inter-networks. It is designated as the successor of IPv4, the current version of the Internet Protocol, for general use on the Internet.

> **Important**
>
> - This section contains a detailed description of IPv6 and deployment considerations associated with it. See IPv6 vs. IPv4 Overview for information about activating support for IPv6 for a Genesys component. For a list of Framework connections that support IPv6, see IPv6 Support.
> - This section includes material that is freely available on the Internet and from other public sources.

Addressing

The primary change from IPv4 to IPv6 is the length of network addresses. IPv6 addresses are 128 bits long (as defined by RFC 4291), whereas IPv4 addresses are 32 bits. This amounts to an address space for IPv4 of approximately 4 billion addresses, compared to 3.4 x 10^{38} unique addresses for IPv6.

IPv6 addresses are typically composed of two logical parts: a 64-bit network or subnetwork prefix, and a 64-bit host part. This host part is either generated automatically from the MAC address of the interface, or assigned sequentially. Because globally unique MAC addresses offer an opportunity to track user equipment (and therefore users) across IPv6 address changes, RFC 3041 was developed to reduce the chance of user identity being permanently tied to an IPv6 address, thus restoring some of the anonymity existing with IPv4. RFC 3041 specifies a mechanism by which time-varying random bit strings can be used as interface circuit identifiers, replacing unchanging and traceable MAC addresses.

Notation

IPv6 addresses are normally written as eight groups of four hexadecimal digits separated by colons (:). For example:

```
2001:0db8:85a3:08d3:1319:8a2e:0370:7334
```

If one or more four-digit groups is `0000`, the zeros can be omitted and replaced with two colons (::). For example:

```
2001:0db8:0000:0000:0000:0000:1428:57ab
```

can be shortened to

```
2001:0db8::1428:57ab
```

Following this rule, any number of consecutive `0000` groups can be reduced to two colons, as long as

there is only one double colon used in an address. Leading zeros in a group can also be omitted (as in `::1` for a `localhost` address). Therefore, the following addresses are all valid and are equivalent:

```
2001:0db8:0000:0000:0000:0000:1428:57ab
2001:0db8:0000:0000:0000::1428:57ab
2001:0db8:0:0:0:0:1428:57ab
2001:0db8:0:0::1428:57ab
2001:0db8::1428:57ab
2001:db8::1428:57ab
```

Note that having more than one double-colon syntax element in an address is invalid, as it would make the notation ambiguous. For example, the following address:

```
2001:0000:0000:FFD3:0000:0000:0000:57ab
```

abbreviated to

```
2001::FFD3::57ab
```

could imply any of the following:

```
2001:0000:0000:0000:0000:FFD3:0000:57ab
2001:0000:FFD3:0000:0000:0000:0000:57ab
```

or any other similar permutation.

For more information about IPV6 addressing, refer to RFC 4291.

Literal IPv6 Addresses in URLs

In a URL, the IPv6 address is enclosed in brackets. For example:

```
http://[2001:0db8:85a3:08d3:1319:8a2e:0370:7344]/
```

This notation enables the parsing of a URL without confusing the IPv6 address and port number, such as in:

```
https://[2001:0db8:85a3:08d3:1319:8a2e:0370:7344]:443/
```

Additional information can be found in RFC 2732 and RFC 3986.

Network Notation

IPv6 networks are written using Classless Inter-Domain Routing (CIDR) notation.

An IPv6 network (or subnet) is a contiguous group of IPv6 addresses, the size of which must be a power of two. The initial bits of any address in the network are called the prefix, and are identical for all hosts in the network.

A network is denoted by the first address in the network, and the size (in bits) of the prefix (in decimal), separated with a forward-slash (`/`). For example:

```
2001:0db8:1234::/48
```

stands for the network with addresses

```
2001:0db8:1234:0000:0000:0000:0000:0000
```

through

```
2001:0db8:1234:ffff:ffff:ffff:ffff:ffff
```

Because a single host can be seen as a network with a 128-bit prefix, host addresses are often followed with `/128`.

Kinds of IPv6 addresses

IPv6 addresses are divided into the following categories (see RFC 4291 - IP Version 6 Addressing Architecture):

- unicast addresses
- multicast addresses
- anycast addresses

Unicast Addresses

A unicast address identifies a single network interface. A packet sent to a unicast address is delivered to that specific computer. The following types of addresses are unicast IPv6 addresses:

- Global unicast addresses
- Link-local addresses (prefix `fe80::/10`): Valid only on a single link; analogous to `169.254.0.0/16` in IPv4
- Unique local IPv6 unicast addresses
- Special addresses (see examples in the following table)

`::/128`	The address with all zeros is an unspecified address, and is to be used only in software.
`::1/128`	The loopback address is a localhost address. It corresponds to `127.0.0.1` in IPv4.
`::ffff:0:0/96`	This prefix is used for IPv4-mapped addresses (see Transition Mechanisms).
`2002::/16`	This prefix is used for 6to4 addressing.
`2001:db8::/32`	This prefix is used in documentation (RFC 3849). Anywhere where an example of an IPv6 address is given, addresses from this prefix should be used.

Multicast Addresses

Multicast addresses are used to define a set of interfaces that typically belong to different nodes instead of just one. When a packet is sent to a multicast address, the protocol delivers the packet to all interfaces identified by that address. Multicast addresses begin with the prefix `FF00::/8`. The second octet identifies the scope of the addresses, that is, the range over which the multicast address is propagated. Commonly used scopes include link-local (`0x2`), site-local (`0x5`) and global (`0xE`).

Anycast Addresses

Anycast addresses are also assigned to more than one interface belonging to different nodes. However, a packet sent to an anycast address is delivered to just one of the member interfaces, typically the closest as defined by the routing protocol. Anycast addresses cannot be easily identified. They have the structure of normal unicast addresses, and differ only by being injected into the routing protocol at multiple points in the network.

Broadcast Addresses

There are no address ranges reserved for broadcast in IPv6. Applications use multicast to the all-hosts group instead. The Internet Assigned Numbers Authority (IANA) maintains the official list of the IPv6 address space. Global unicast assignments can be found on the various Regional Internet Registries (RIR) or on the Ghost Route Hunter Default Free Prefixes (GRH DFP) pages.

Transition Mechanisms

Until IPv6 completely supplants IPv4, which is not expected to occur in the foreseeable future, a number of transition mechanisms are needed to enable IPv6-only hosts to reach IPv4 services and to allow isolated IPv6 hosts and networks to reach the IPv6 Internet over the IPv4 infrastructure. An overview of some of the various IPv6 transitions that currently exist is provided at: https://www.sixxs.net/faq/connectivity/?faq=comparison.

Dual Stack

Because IPv6 is a conservative extension of IPv4, it is relatively easy to write a network stack that supports both IPv4 and IPv6 while sharing most of the source code. Such an implementation is called a *dual stack*, and a host implementing a dual stack is called a *dual-stack host*. This approach is described in RFC 4213}}.

Most current implementations of IPv6 use a dual stack. Some early experimental implementations used independent IPv4 and IPv6 stacks. There are no known implementations that implement IPv6 only.

Tunneling

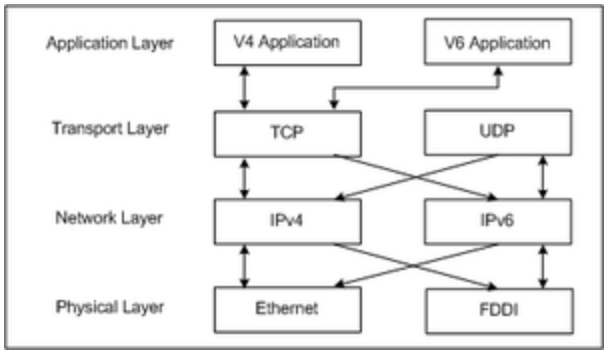
To reach the IPv6 Internet, an isolated host or network must be able to use the existing IPv4 infrastructure to carry IPv6 packets. This is done using a technique referred to as *tunneling*. Tunneling consists of encapsulating IPv6 packets within IPv4, in effect using IPv4 as a link layer for IPv6.

IPv6 packets can be directly encapsulated within IPv4 packets using Protocol 41. They can also be encapsulated within User Datagram Protocol (UDP) packets, for example, to cross a router or Network Address Translation (NAT) device that blocks Protocol 41 traffic. They can also use generic encapsulation schemes, such as Anything In Anything (AYIYA) or Generic Routing Encapsulation (GRE).

Architecture

Dual-Stack IPv6 Implementation

Genesys support for IPv6 relies on true dual-stack IPv6 implementation of the operating system as specified in RFC 3493. Conceptually, the configuration of a dual-stack machine with a v4 TCP and a v6 TCP application is shown in the following figure.



Dual-Stack Architecture

Using this approach, you can write an application that can operate with both IPv4 and IPv6 peers using just one socket. In addition, an application that uses a properly designed Transport Layer library and does not have to operate directly with IP addresses (and other Network Layer elements) may not be aware of the IP version used at all.

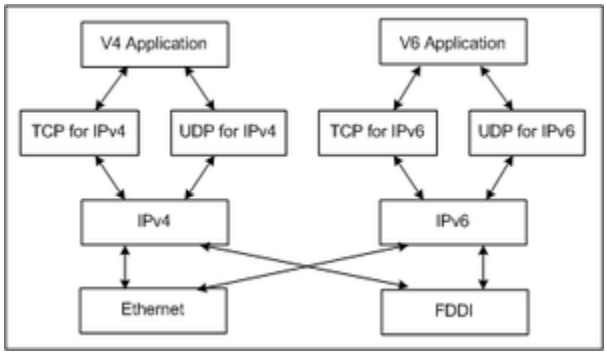
Microsoft Windows Implementation

Microsoft uses slightly different terminology when describing IPv6 architecture. For Microsoft, dual-layer refers to dual-network layers sharing a single transport layer. Dual-stack refers to dual-network layers and dual transport layers, that is, two separate stacks. Only a dual-layer architecture is compliant with RFC 3493.

> **Important**
>
> In this document dual-stack always refers to RFC 3493-compliant implementations, not the Microsoft definition.

Windows Server 2000/2003 and Windows XP

The following figure illustrates Microsoft Windows IPv6 implementation prior to Windows Vista. Microsoft calls this a dual-stack architecture, but it is actually implemented as two separate stacks with separate TCP and UDP paths. This implementation forces an application to open separate sockets to talk to IPv4 and IPv6 peers.



Microsoft IPv6 Stack Prior to Windows Vista

Windows Vista

In Windows Vista, Microsoft calls its next generation IP stack dual-layer architecture, but it is actually a correct dual-stack implementation as described above, where there is only a single transport layer component for TCP and UDP.

Operating Systems Supporting Dual-Stack Architecture for IPv6

Operating system support of dual-stack IPv6 implementation (RFC 3493-compliant) by different operating system platforms is provided in the following table. Refer to platform-specific documentation (including web sites) for additional information about supporting and implementing IPv6.

Operating System	Supporting Releases
AIX	AIX 4.3.3 and later
Linux	kernel 2.6 (Red Hat Enterprise Linux release 4) and later
Mac OS X	Mac OS 10.3 Panther and later
Solaris	Solaris 8 and later
Windows	Windows Vista and later, Windows Server 2008 and later

DNS

Genesys products are using Domain Name System (DNS) resolution of hostnames specified in configuration, and require that the DNS is operating according to the AAAA schema. IPv6 addresses are represented in the Domain Name System by AAAA records (so-called `quad-A records`) for forward lookups; reverse lookups take place under `ip6.arpa` (previously `ip6.int`), where the address space is delegated on nibble boundaries. This scheme, which is a straightforward adaptation of the familiar A record and in-addr.arpa schemes, is defined in RFC 3596. The following table describes the fields in an AAAA record.

Field Name	Description
NAME	Domain name
TYPE	AAAA (28)
CLASS	Internet (1)
TTL	Time to live (seconds)
RDLENGTH	Length of RDATA field
RDATA	String form of the IPv6 address as described in RFC 3513

RFC 3484 specifies how applications should select an IPv6 or IPv4 address for use, including addresses retrieved from DNS. For mixed networks, the DNS must provide both A and AAAA records.

On a historical note, the AAAA schema was one of two proposals at the time the IPv6 architecture

was being designed. The other proposal, designed to facilitate network renumbering, would have had A6 records for the forward lookup and a number of other innovations such as bit-string labels and DNAME records. It is defined in the experimental RFC 2874 and its references (with further discussion of the advantages and disadvantages of both schemes in RFC 3364).

Virtualization

There are no known restrictions from the Genesys side for deploying IPv6 in a virtual operating environment. Check with the documentation specific to the virtual environment you are using for more information and any limitations.

License Control

> **Important**
> The information in this section is based on information provided in Flexera documentation, and may be specific to their products. For information about IPv6 support and implementation for other licensing products, consult documentation specific to the product.

Genesys uses FlexLM 9.5 and FlexNet Publisher 11.9-based license control, but only the FlexNet Publisher Licensing toolkit 11.9 supports IPv6. Genesys License Server 8.1 uses FlexNet Publisher 11.9 for all platforms.

The following table summarizes the addressing compatibility of a FlexNet License Server Machine and a Flex-enabled Application Server, as described in this section.

<table>
<tr><th colspan="2" rowspan="2"></th><th colspan="4">FlexNet License Server Machine</th></tr>
<tr><th>IPv4-only</th><th>Dual IPv4/ IPv6 Stack</th><th>IPv6-only</th><th>No Server</th></tr>
<tr><td rowspan="5">Flex-enabled Application Server</td><td>IPv4-only</td><td rowspan="3">Use IPv4 only</td><td>Use IPv4 only</td><td rowspan="2">Not supported</td><td rowspan="2">Use IPv4 only</td></tr>
<tr><td>Dual Stack using IPv4 only</td><td rowspan="2">Use IPv4, IPv6, or both</td></tr>
<tr><td>Dual Stack using IPv4 and IPv6</td><td rowspan="3">Use IPv6 only</td><td>Use IPv4, IPv6, or both</td></tr>
<tr><td>Dual Stack using IPv6 only</td><td rowspan="2">Not supported</td><td rowspan="2">Use IPv6 only</td><td rowspan="2">Use IPv6 only</td></tr>
<tr><td>IPv6 only[a]</td><td>Use IPv6 only</td></tr>
</table>

[a] Genesys does not recommend or support IPv6 environments.

In the license file, an IPv6 address should be defined as the `host` value in the SERVER line. Entries in the license search path that use the `port@host` convention to identify the license server can also specify an IPv6 address as the `host` value.

Deploying License Servers in Mixed Protocol Environments

For FlexNet Publisher components to work properly using IPv6 addresses, all systems in an enterprise (including the network hardware and software) must be configured properly to support communication using IPv6 addresses.

Before testing or deploying a FlexEnabled application that supports IPv6 or IPv4/IPv6 dual communication, make sure that all systems on the network can communicate successfully. If the license server can run under any of the following operating systems:

- Any supported edition of Windows Vista
- Any supported Linux platform
- Any supported Unix platform

it can communicate with FlexEnabled clients using either IPv4 or IPv6, so long as the network is configured properly.

Because these operating systems support dual-layer communication, both IPv4 and IPv6 FlexEnabled clients can communicate with an IPv6 license server. In addition, IPv6 clients can communicate with an IPv4 license server using the IPv4 address.

The FlexNet Publisher license server lmadmin supports both IPv4 and IPv6 clients. If you are using it, you must rename one of your vendor daemon executable files, because separate IPv4 and IPv6 vendor daemons are required.

If the license server runs on Windows XP or Windows Server 2003, there are certain limitations because of the limited dual-layer support on these operating systems (see Windows Server 2000/ 2003 and Windows XP). IPv4 FlexEnabled clients cannot communicate with an IPv6 license server running on these operating systems. However, IPv6 FlexEnabled clients can communicate with an IPv4 license server running on these operating systems.

If an enterprise runs license servers on Windows 2003 or Windows XP, the license administrators should create and maintain two separate networks - one for IPv6 FlexEnabled clients that will use the IPv6 license server, and one for IPv4 FlexEnabled clients that will use the IPv4 license server.

Using Wildcards in an IPv6 Address

An asterisk (*) can be used as a wildcard character in place of an entire field or on a byte-by-byte basis to specify a range of addresses without having to list them all.

For example, the following feature definition line is locked to four specific addresses:

```
FEATURE f1 myvendor 1.0 1-jan-2010 uncounted \
HOSTID="INTERNET=127.17.0.1,\
INTERNET=2001:0db8:0000:0000:ff8f:effa:13da:0001,\
INTERNET=127.17.0.4,\
NTERNET=2001:0db8:0000:0000:ff8f:effa:13da:0004" \
SIGN="<...>"
```

The following feature definition line specifies an entire range of addresses, including the four specific ones from the line above:

```
FEATURE f1 myvendor 1.0 1-jan-2010 uncounted \
HOSTID="INTERNET=127.17.0.*,\
INTERNET=2001:0db8:0000:0000:*:*:*:000*"\
SIGN="<...>"
```

Genesys IPv6 Support

Genesys supports IPv6 as described in this section.

Common Principles

The implementation of IPv6 in Genesys is based on the following assumptions:

- Dual-stack requirement and backward compatibility
- Dual IPv4/IPv6 server sockets
- IPv4 preference for DNS

Dual-Stack Requirement and Backward Compatibility

Only dual-stack IPv6 implementations are supported. Support of IPv6 on Windows 2002/2003 and XP is not required, while all recent versions of UNIX have dual-stack support already. However, the connection layer must still operate on all other platforms in IPv4 mode only.

On the platforms where IPv6 support is available, the default mode of operation is IPv4 for backward compatibility. IPv6 support must be turned on explicitly by each application using one of the following methods:

- Set the environment variable `GCTI_CONN_IPV6_ON` to 1.
- In the `common` section of the `Application` object's options, set `enable-ipv6` to 1.

Refer to IPv6 vs. IPv4 Overview for more details about enabling IPv6 in Genesys software.

> **Important**
> IPv6 is, by default, not enabled. But once it is enabled using one of the methods described above, it can only be disabled by turning it off in both places-the environment variable and the option. That is, turning it off in one location only disables it if it is not enabled in the other.

Dual IPv4/IPv6 Server Sockets

By default, a server socket opened by a standard method should accept both IPv4 and IPv6 client connections. That is, unless IPv6 is disabled on a particular node, unbound server sockets are opened

with the AF_INET6 family and use the AI_V4MAPPED flag to interact with IPv4 clients. However, a server socket bound to a particular IP address (either IPv4 or IPv6) only accepts a connection of the same IP family.

IPv4 Preference for DNS

Within an application, a name service should be used whenever possible. An AAAA record may return both a IPv4 and IPv6 address for dual stack nodes. For backward compatibility reasons, client connections in this case should prefer IPv4 over IPv6. That preference can be set using the configuration option `ip-version`.

However, a client connection bound to a particular IP address (either IPv4 or IPv6) can only interact with the server using a connection of the same IP family.

Implementation Characteristics

Individual Genesys components support the following features related to IPv6:

- Full IPv6 support in DNS lookup: Support both AAAA records and DNS over IPv6.
- Transparent server-side socket handling: The existing server-side interface allows IPv6 connections whenever possible using the AI_V4MAPPED flag.
- Transparent client-side connection: The existing client-side connection interface allows IPv6 connections by host name or explicit IP address in text format.
- DNS Lookup modes: Full DNS support using the synchronous method (name lookup using standard system calls) and asynchronous DNS (enabled by the `enable-async-dns` option in the `common` section of an `Application` object's options). Server and client side IPv6 sockets and connections are supported transparently, including hosts being addressed either by name, or by textual IP address in either IPv4 or IPv6 format.
- IPv6-related changes in the configuration environment: Configuration Server keeps IP addresses for all configured hosts, but it is not a replacement for DNS. However it is expected to be affected very little. In particular, a new field for the IPv6 address is not added to the CfgHost structure; while the new configuration option `ip-version` set at the connection level determines whether the connection uses IPv4 first (4,6; the default), or IPv6 first (6,4). To achieve compatibility with legacy servers (that is, a server without IPv6 support running on a dual-stack host, while IPv6-enabled clients try to connect), the suggested solution is to create an IPv4-only hostname alias for that host.

For more information about the two configuration options, refer to IPv6 vs. IPv4 Overview.

IPv6 Support by Genesys Products

To determine if a Genesys product supports IPv6, refer to the documentation for that product. Framework connections that support IPv6 are listed in IPv6 Support.

Deployment Considerations

When deploying IPv6 in your Genesys environment, you must take into consideration the factors discussed in this section.

Security

Preparation for IPv6 utilization will require careful planning of security measures, because IPv6 presents new challenges compared to IPv4. Some, but not all, of the challenges are discussed in this section.

TLS

In some deployments, multiple hostnames are assigned to a given computer, and are resolved to different IP versions. In this case, the TLS certificate of the given computer will have to be generated for all assigned hostnames.

Refer to the *Genesys Security Deployment Guide* for information about generating certificates.

Firewall and Client-Side Port

Genesys supports fine-grain firewall configuration at the port-level and applied both to incoming client connections and their target server destinations.

In IPv6 deployments, this might become even more valuable, for example, as a countermeasure against Network Discovery (ND) attacks. ND in IPv6 utilizes five different types of ICMPv6 messages for several purposes. ND attacks in IPv6 will likely replace ARP spoofing in IPv4.

Internet Protocol Security

Internet Protocol Security (IPSec) is an optional feature in IPv4, but is mandatory in IPv6. In certain deployments, it could make the use of TLS unnecessary.

DNS Security Extensions

Genesys recommends the use of DNS Security Extensions (DNSSEC), but it is not mandatory. There are no dependencies from the Genesys side.

IP Tunneling

When connecting sites, you may want to use IP tunneling. For example, two sites could be operating in IPv4 mode while the interconnection requires IPv6. In this case, one could consider embedding the IPv4 protocol into an IPv6 connection between sites.

Licensing

The G8.1 License Server (the Genesys vendor daemon) is based on FlexNet Publisher 11.9, and is IPv6 enabled.

However, within G8.1 the IPv6-enabled licensing client libraries (FlexNet Publisher 11.9) are implemented for only the RHEL 5 64-bit, Windows 2008 64-bit, and HP-UX Integrity (Itanium) operating systems. For all other platforms, the G8.1 applications are still using the older client libraries, which are not IPv6 enabled. This is done to provide backward compatibility; otherwise, the deployment of a G8.1 application in an existing environment would have required a complete upgrade of the licensing system.

Therefore, Genesys recommends that IPv4 be used for licensing.

SIP

The SIP protocol can contain explicit IP address values. This creates additional challenges, for example at the NAT level, but also if the same SIP Server instance has to concurrently support multiple SIP interfaces where one is operating in IPv4 mode and another in IPv6 mode.

It is recommended to address those scenarios by using a dedicated SIP Server for IPv4 only and another one for IPv6 only.

You could also consider using available NAT solutions that perform configurable SIP protocol inspection and conversion. One example is F5 Networks Big-IP LTM.

Thin Clients

Some Genesys client applications offer a web browser interface with an HTTP connection to a web server. These connections are under control of the given web technology, and all modern browsers already support IPv6. However, IPv6 must be enabled at both the client computer and server computer sides, and the DNS involved must also support IPv6.

External Interfaces

IPv4 dependencies at external interfaces must be considered. This includes, for example, interfaces to Session Border Controllers (SBC), media gateways, switches, and databases.

Dynamic Runtime Changes

Changes in the IPv4/IPv6 configuration should be performed during maintenance windows, as they will require a restart of impacted processes. These changes will include setting the following:

- Transport parameter `ip-version`
- DNS entries for hostnames
- Local computing node settings

Third-Party Dependencies

Genesys uses several third-party products as part of the suite. The IP capabilities of those products must be considered.

IPv6 vs. IPv4 Overview

> **Important**
>
> This section provides a high-level view of IPv6, and how to enable it in a Genesys component. For detailed information about IPv6, the operating systems that support it, and things to consider when deploying it, see Internet Protocol version 6 (IPv6). For a list of Framework connections that support IPv6, see IPv6 Support.

Internet Protocol version 6, commonly known as IPv6, is a network layer protocol for packet-switched inter-networks. It is designated as the successor of IPv4, the current version of the Internet Protocol, for general use on the Internet.

Configuring IPv6

IPv6 must be configured on each component that is going to support it. You can do this using an environment variable or a configuration option, depending on the situation:

You must use an environment variable if:

- An IPv6 connection is to be established before an application is able to, or must, read information from Configuration Server.
- You want all Genesys applications on the same host to support IPv6. You only have to configure this on the host once, rather than configure each application on that host individually. The host-level setting will override any application-level setting.

Otherwise, you can use either an environment variable or a configuration option.

Set Environment Variable

Set the environment variable **GCTI_CONN_IPV6_ON** to `true` (represented by any non-zero integer value) to enable IPv6; or to `false` (represented by zero (`0`)) to disable IPv6. The default value of this environment variable is `false` (`0`), indicating that IPv6 support is disabled. This default value ensures backward compatibility.

Set Configuration Option

Using Genesys Administrator, set the **enable-ipv6** option in the **[common]** section of the options of the component's Application object. Refer to the *Framework Configuration Options Reference Manual*

for more information about this option.

Refer to component-specific documentation and Internet Protocol version 6 (IPv6) for more information about IPv6 and any specific considerations for deploying IPv6 in your situation.

Mixed IPv4 and IPv6 Environments

You can configure IPv6 and IPv4 in the same environment, as described in Internet Protocol version 6 (IPv6). In this mixed environment, you can configure connections with servers that support IPv4, IPv6, and both. For connections with servers that support both IPv4 and IPv6, you can specify which version you prefer to use. For example, if you are setting up a connection to a DB Server that supports both IP4 and IPv6, you can choose to use IPv4 or IPv6 for that connection. There is no universal rule that determines what version should be used, so the choice is up to you.

To configure this choice, you can use either an environment variable or a transport option.

Set Environment Variable

Set the environment variable **GCTI_CONN_IP_VERSION** to either `4,6` to indicate a preference for IPv4; or to `6,4` to indicate a preference for IPv6. The default value of this environment variable is `4,6`, indicating that IPv4 is preferred. This default value ensures backward compatibility. The value of this environment variable is overridden if the transport parameter **ip-version** is configured at the Application level.

Set Transport Option

Using Genesys Administrator, set the Transport parameter **ip-version** on the **Advanced** tab of the **Connection Info** dialog box for the connection. Refer to the *Framework Configuration Options Reference Manual* for more information about this parameter.

> ### Important
> This option has no affect on connections to Configuration Server that are established before the option value can be read from the Configuration Database.

The same option can also be set in the transport section of the Host's annex to set the IP version used on the connection between Solution Control Server and LCA. Otherwise, Management Framework components do not support this option. When the same transport option is set on the application level, it overrides the value of the environment variable **GCTI_CONN_IP_VERSION**.

Refer to the *Framework Configuration Options Reference Manual* for more information about setting this option in an application or in a host.

The following table summarizes how this environment variable or option affects the connection for which it is configured.

Connecting Server	4,6	6,4
Supports only IPv4	ipv4 is used	ipv4 is used
Supports both IPv4 and IPv6	ipv4 is used	ipv6 is used
Supports only IPv6	ipv6 is used	ipv6 is used

Genesys Implementation of Secure Protocol Connections

This topic describes how to use the Genesys Security Pack to implement secure connections in Management Framework. More detailed information about OpenSSL, and how to use it to secure connections between Genesys components, is contained in the Secure Connections (TLS) section of the *Genesys Security Deployment Guide*.

Deployment Steps

To deploy Security Pack on UNIX or Linux, run the installation package. After the files have been copied into the target folders, make sure that you set up required environment variables to allow Genesys applications to load shared modules from those locations. For example, on Linux, you might want to include the installation path to the LD_LIBRARY_PATH environment variable and restart affected applications. Note that if you are using LCA to start your applications from the Management Layer, LCA must be restarted first to pick up the changes to the environment variables before you can (re)start the applications.

On Windows operating systems, you do not need to deploy any additional software; secure connections are available for use by any Genesys application that supports them.

Backward Compatibility

The new Security Pack is a drop-in replacement of the existing Security Pack. To upgrade to the OpenSSL version, you replace the binary modules. You do not have to make any change to the configuration of existing deployments.

> **Important**
>
> You must restart those applications using secured connections after upgrading to the new Security Pack.

To ensure backward compatibility, the new Security Pack includes a new mode (referred to as compatibility mode) that restores some behavior of the old Security Pack. This mode is disabled by default.

> **Warning**

- Compatibility mode should be enabled only as a last resort if the new Security Pack is encountering compatibility errors in the customer environment.
- When in compatibility mode, Genesys strongly recommends that you take the necessary actions to avoid long-term usage of this mode.

To enable the Security Pack compatibility mode, set the environment variable GCTI_SECPACK_COMPAT_MODE to 1 before starting the application. Once started, you cannot disable the mode during application runtime.

The following compatibility issue workarounds are enabled by compatibility mode:

- When verifying a peer certificate chain, a chain entry certificate revocation status will be ignored if the certificate is explicitly trusted as a CA in the local configuration (that is, listed in the ca certificate list).
- The peer certificate chain verification process will ignore any non-compatible "Key usage" extension value. For example, a peer certificate without "authentication" usage will be accepted in compatibility mode. RSA did not verify the "Key usage" extension values; OpenSSL does.

If you want to continue using the RSA BSafe implementation instead of OpenSSL, make sure you set up your environment so that shared modules from the **<Security Pack root>/legacy** folder have being loaded instead of the default ones (located in **<Security Pack root>**).

OpenSSL

OpenSSL is the industry standard SSL implementation. It is widely used in both open source and commercial products, and so existing vulnerabilities and issues are promptly discovered and fixed.

OpenSSL is developed in parallel with all the new features introduced into SSL, such as TLSv1.2 protocol version support.

Version Information

The OpenSSL website contains release strategy information, describing the version naming, release schedule and more. See the Links section for additional information regarding version information.

To determine the version of OpenSSL that the Security Pack is using, refer to the most recent Security Pack on UNIX Release Note.

FIPS mode information

OpenSSL provides native support for FIPS mode. Unlike RSA, OpenSSL does not require a special version capable of performing FIPS mode operations. A FIPS-capable version of OpenSSL is used by the Security Pack. For more info on FIPS mode and OpenSSL support of FIPS, please refer to OpenSSL documentation and the FIPS User Guide.

Links

OpenSSL site	https://www.openssl.org/
OpenSSL backwards compatibility analysis dashboard	http://upstream.rosalinux.ru/versions/openssl.html
OpenSSL FIPS User Guide	https://www.openssl.org/docs/fips/UserGuide-2.0.pdf

Protocol Versions Compatibility

The **sec-protocol** option supports the following modes: SSLv23 (the default), SSLv3, TLSv1, TLSv11, and TLSv12.

The availability of a particular protocol setting in **sec-protocol** strongly depends on the actual component version. Older components may not support this option at all. No components except the most recent Management Framework servers support the TLSv12 value. For other components, refer to documentation specific to the component to determine protocols are supported (if any), and for additional information.

Generally, the protocol versions currently available are as follows:

- On UNIX and Linux, TLS 1.2 is the highest available protocol with OpenSSL SecPack; TLS 1.1 with RSA SecPack.
- On Windows, TLS 1.1 and TLS 1.2 are supported starting with Microsoft Vista / Server 2008. However, in most cases these must be enabled in the registry to become available. Genesys recommends that you explicitly enable the desired protocol version in the Windows registry; refer to the following Windows document for more information about enabling and disabling protocols in the Windows registry: TLS/SSL Settings. Note that Genesys components use the Windows implementation of TLS on Windows platforms, and hence Windows settings take precedence over the **sec-protocol** settings. Genesys software is unable to use a protocol version if it is disabled on the Windows operating system level.

The supported protocol version modes can be categorized as one of two types: strict or compatibility. These are described below.

Strict protocol version modes

SSLv3, TLSv1, TLSv11, and TLSv12 are the strict protocol version modes. These settings can be used to enforce a specific protocol version. The connection will not be established if the remote server does not accept the enforced protocol version.

Compatibility protocol version modes

SSLv23, the default mode, is compatible with all modes from SSLv2 up to and including TLSv12, and will connect with the highest mode offered by the other server. If SSL 2 ciphers are explicitly specified, the SSL 2 client can connect only to servers running in SSLv23 mode. Otherwise, the SSL 2 mode is deprecated; it is highly vulnerable and is not to be used.

Windows Authentication with MS SQL Server for Framework Applications

Windows Authentication provides a more secure way for an Application to access an MS SQL database without storing the database password in the Genesys configuration.

Full details about how to enable Windows Authentication with MS SQL Server are described in the "Microsoft SQL Server Databases" section of the *Framework Database Connectivity Guide*.

This topic describes how to enable and configure Framework Applications to use Windows Authentication with MS SQL Server.

Prerequisites

Before you can configure individual applications to use Windows Authentication, you must first create Windows processes on the MS SQL Server for each component that will be accessing the database. Specifically, you need to create a Windows process for the following components in the scenarios that applies to your configuration:

- Configuration Server with a direct connection to the Configuration Database—None
- Configuration Server with an indirect connection to the Configuration Database—DB Server 8.1.3
- Message Server with a direct connection to the Log Database—Message Server
- Message Server with an indirect connection to the Log Database—DB Server 8.1.3

Follow the instructions here to configure the Windows processes required.

> **Important**
> If you are using DB Server 8.1.3, it must be using DB Client 8.5.1 or higher.

Configuring Applications to use Windows Authentication when Accessing MS SQL Server

How you configure a Framework Application to use Windows Authentication depends on how it accesses its database, that is, whether it uses DB Server or not. These scenarios are described below.

> **Important**
>
> For a full description of Windows Authentication, see the "Windows Authentication with MS SQL Server" section of the Microsoft SQL Server Databases topic in the "*Framework Database Connectivity Guide*. These scenarios assume that you are using Windows Authentication with Domain Source Name (DSN), not with a Trusted User. See "Configuring Applications to use Windows Authentication when Accessing MS SQL Server" on the same topic above for details about these two kinds of Windows Authentication.

Configuration Server

Using DB Server

If Configuration Server is using DB Server, you must:

1. Set up a Windows process on MS SQL Server for DB Server, as described here.
2. In Configuration Server, configure the options that describe the Configuration Database and set **dbthread=false** in the configuration file **confserv.cfg**, as shown in the example below.

Using a DSN named "sqldsn", the entries would look something like this:

```
...
[confserv]
...
dbthread=false
...

[dbserver]
host=<name of host on which DB Server is running>
port =<host port through which to communicate with DB Server>
dbengine=mssql
dbserver=dsn
dbname=sqldsn
username=<this field can be filled with a dummy username>
password=<this field can be left empty or filled with a dummy password>
...
```

Not Using DB Server

If Configuration Server is not using DB Server, you must set **dbthread=true** and the options that describe the Configuration Database in the configuration file **confserv.cfg**, as shown in the example below.

Using a DSN named "sqldsn", the entries would look something like this:

```
...
[confserv]
...
dbthread=true
...

[dbserver]
```

```
host=<name of host on which DBMS is running>
port =<host port through which to communicate with the DBMS>
dbengine=mssql
dbserver=dsn
dbname=sqldsn
username=<this field can be filled with a dummy username>
password=<this field can be left empty or filled with a dummy password>
...
```

Message Server

If Message Server is using DB Server, you must set **dbthread=false** in the **[messages]** section of the Message Server Application object. Configure a DAP in which DB Server is set. DB Server must be using DB Client 8.5.1 or higher. You must also configure a Windows process for DB Server, as described here.

If Message Server is accessing the Log Database directly (without DB Server), you must still configure a DAP but it will not point to DB Server. Set **dbthread=true** in the **[messages]** section of the Message Server Application object. You must also set up a Windows process for Message Server, as described here.

In both cases, set the Username field to a dummy username.

SQL Server Authentication with MS SQL Server for Framework Applications

Genesys applications support SQL Server Authentication to connect to an MS SQL Database by using Data Source Name (DSN). For more details of SQL Server Authentication, see the SQL Server Authentication with MS SQL Server section in the *Framework Database Connectivity Guide*.

Document Change History

This section provides a summary of changes that have been made in this document in 9.0.

June 8, 2018

The following pages have been added or updated:

- The single new Disaster Recovery / Business Continuity page replaces the former Disaster Recovery / Business Continuity pages for Oracle Golden Gate and MS SQL Always On. The two configuration pages have been moved under this page from the Additional Information section.
- The new MS SQL Cluster Database with MultiSubnet Listener for Framework Applications page describes how to implement MS SQL Always On MultiSubnetFailover, as described in the Framework Database Connectivity Guide.

May 1, 2018

The following pages have been added or updated:

- The Permission Prerequisites page has been renamed Minimum Required Permissions and Privileges, and moved to the Additional Information section. Minimum database privileges for both deploying and accessing each of the Configuration Server and the Centralized Log Database has been added. Information for DB2 databases has been added because it is no longer available.

March 27, 2018

The following pages have been added or updated:

- Information has been added to Configuration Server Proxy about limiting the number of Configuration Server Proxies that can upload data simultaneously.
- The new Transaction Serialization page describes how Configuration Server (or a master Configuration Server, if in a distributed environment) can defer data change requests to ensure that each request is completely processed before another request starts to be processed.

January 26, 2018

The following pages have been added or updated:

- Instructions for starting Solution Control Server using its configuration file has been added to the

Solution Control Server section of the Starting Manually page.

- A note clarifying the scope of the information has been added to the pages Configuring Disaster Recovery Using DNS Failover and Oracle GoldenGate and Configuring Disaster Recovery Using MS Failover Cluster and MS SQL AlwaysOn.
- The steps of Configuring Genesys Components have been updated on the page Configuring Disaster Recovery Using DNS Failover and Oracle GoldenGate.

September 26, 2017

The following pages have been added or updated:

- Starting and stopping Management Framework Using the Management Layer has been updated with important information about starting Applications on Linux.

August 22, 2017

The following pages have been added or updated:

- When creating hosts, it is recommended that you select the generic Windows operating system if you are running Windows 2016 or later.
- The optional `-f` parameter for starting SCS has been added to Manually Starting SCS.