Framework 8.0

Stat Server

Deployment Guide

About Genesys

Genesys Telecommunications Laboratories, Inc., a subsidiary of Alcatel-Lucent, is 100% focused on software for contact centers. Genesys recognizes that better interactions drive better business and build company reputations. Customer service solutions from Genesys deliver on this promise for Global 2000 enterprises, government organizations, and telecommunications service providers across 80 countries, directing more than 100 million customer interactions every day. Sophisticated routing and reporting across voice, e-mail, and Web channels ensure that customers are quickly connected to the best available resource—the first time. Genesys offers solutions for customer service, help desks, order desks, collections, outbound telesales and service, and workforce management. Visit www.genesyslab.com for more information.

Each product has its own documentation for online viewing at the Genesys Technical Support website or on the Documentation Library DVD, which is available from Genesys upon request. For more information, contact your sales representative.

Notice

Although reasonable effort is made to ensure that the information in this document is complete and accurate at the time of release, Genesys Telecommunications Laboratories, Inc., cannot assume responsibility for any existing errors. Changes and/or corrections to the information contained in this document may be incorporated in future versions.

Your Responsibility for Your System's Security

You are responsible for the security of your system. Product administration to prevent unauthorized use is your responsibility. Your system administrator should read all documents provided with this product to fully understand the features available that reduce your risk of incurring charges for unlicensed use of Genesys products.

Trademarks

Genesys, the Genesys logo, and T-Server are registered trademarks of Genesys Telecommunications Laboratories, Inc. All other trademarks and trade names referred to in this document are the property of other companies. The Crystal monospace font is used by permission of Software Renovation Corporation, www.SoftwareRenovation.com.

Technical Support from VARs

If you have purchased support from a value-added reseller (VAR), please contact the VAR for technical support.

Technical Support from Genesys

If you have purchased support directly from Genesys, please contact Genesys Technical Support at the regional numbers provided on page 9. For complete contact information and procedures, refer to the ***Genesys Technical Support Guide***.

Ordering and Licensing Information

Complete information on ordering and licensing Genesys products can be found in the ***Genesys Licensing Guide.***

Released by

Genesys Telecommunications Laboratories, Inc. www.genesyslab.com

Document Version: 80fr_dep_statserver_11-2009_v8.0.001.00

Table of Contents

Preface

Welcome to the *Framework 8.0 Stat Server Deployment Guide*. This document introduces you to the configuration, installation, and start procedures relevant to Stat Server. This guide is valid only for the 8.0.*x* releases of Stat Server.

Note: For releases of this document created for other releases of this product, please visit the Genesys Technical Support website, or request the Documentation Library DVD, which you can order by e-mail from Genesys Order Management at orderman@genesyslab.com.

This preface provides an overview of this guide, identifies the primary audience, introduces document conventions, and lists related reference information:

About Stat Server

Stat Server is part of the Services Layer of the Genesys Framework. This key component is used by other Genesys solutions and Solution Reporting to track the real-time states of interaction management resources and to calculate basic measurements about the performance of contact center events and activities.

Intended Audience

This guide, primarily intended for network, IT, and contact center administrators, assumes that you have a basic understanding of:

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

You should also be familiar with Genesys Framework and Genesys solutions architecture and functions.

Chapter Summaries

In addition to this preface, this guide contains the following chapters and appendixes:

- Chapter 1, “Overview of Stat Server,” on page 13, describes where Stat Server fits within the Genesys Management Framework, introduces the Statistics Call Model and application programming interface (API), and references other documents where these topics are more fully described.
- Chapter 2, “Stat Server Wizard,” on page 15, describes the different modes of operation of the Stat Server Wizard and how to install and invoke it.
- Chapter 3, “Optimizing Performance,” on page 19, lists some recommendations for optimizing Stat Server performance.
- Chapter 4, “Configuring a Stat Server Application,” on page 21, describes how to set up a Stat Server `Application` object both manually and using the Stat Server Wizard.
- Chapter 5, “Fine-Tuning Stat Server Configuration,” on page 29, describes the configuration options you can set within the Stat Server `Application` object to affect how Stat Server operates.
- Chapter 6, “Other Factors Affecting Stat Server,” on page 57, describes some of the attributes in other Genesys applications that affect how Stat Server operates.
- Chapter 7, “Common Log Options,” on page 61, describes log configuration options that are common to all Genesys server applications, including Stat Server.
- Chapter 8, “Installing Stat Server,” on page 83, describes how to install the application following manual or wizard configuration. This chapter also demonstrates how to install Stat Server silently—without interactive dialog during deployment.

- Chapter 9, "Starting and Stopping Stat Server," on page 95, describes how to start and stop Stat Server on all supported platforms as well as via the Genesys Solution Control Interface.
- Chapter 10, "Application Files," on page 101, describes the files that are deployed during installation.
- Appendix A, "Physical Data Models for Stat Server Tables," on page 105, describes the structure of the database tables to which Stat Server writes data, if configured to do so.
- Appendix B, "Manually Purging Data from the Stat Server Database," on page 115, provides some guidelines for performing a manual purge of Stat Server tables.

Making Comments on This Document

If you especially like or dislike anything about this document, feel free to e-mail your comments to Techpubs.webadmin@genesyslab.com.

You can comment on what you regard as specific errors or omissions, and on the accuracy, organization, subject matter, or completeness of this document. Please limit your comments to the scope of this document only and to the way in which the information is presented. Contact your Genesys Account Representative or Genesys Technical Support if you have suggestions about the product itself.

When you send us comments, you grant Genesys a nonexclusive right to use or distribute your comments in any way it believes appropriate, without incurring any obligation to you.

Contacting Genesys Technical Support

If you have purchased support directly from Genesys, contact Genesys Technical Support at the following regional numbers:

Region	Telephone	E-Mail
North America and Latin America	+888-369-5555 (toll-free) +506-674-6767	support@genesyslab.com
Europe, Middle East, and Africa	+44-(0)-1276-45-7002	support@genesyslab.co.uk
Asia Pacific	+61-7-3368-6868	support@genesyslab.com.au
Before contacting technical support, refer to the *Genesys Technical Support Guide* for complete contact information and procedures.		

Region	Telephone	E-Mail
Malaysia	1-800-814-472 (toll-free) +61-7-3368-6868	support@genesyslab.com.au
India	1-800-407-436379 (toll-free) +91-(022)-3918-0537	support@genesyslab.com.au
Japan	+81-3-6361-8950	support@genesyslab.co.jp
Before contacting technical support, refer to the *Genesys Technical Support Guide* for complete contact information and procedures.		

New in this Release

This *Framework 8.0 Stat Server Deployment Guide* includes the following new features over the 7.6.1 release:

- Support for secure Transport Layer Security connections between Stat Server and the following Genesys servers:
  - Configuration Server
  - T-Server
  - DB Server
  - Outbound Contact Server
  - Universal Router Server

  Note: Local Control Agent (LCA) connections between the Solution Control Server and Stat Server are not secure. LCA enables one local process to connect to another in order to start and stop it. No sensitive customer information passes through LCA connections.

  Part 4 of the *Genesys Security Deployment Guide* describes how to configure secure ports for server components as well as how to configure secure connections between components.
- Information for configuring a Stat Server application using Genesys Administrator, described on page 26.
- Several new options to direct Stat Server behavior:
  - `allow-vq-orig-dns-from-environment`, described on page 31.
  - `check-vq-stuck-calls-frequency`, described on page 33.
  - `load-balance-aht`, described on page 38.
  - `queue-use-pseudo-actions`, described on page 40.
  - `vq-ignore-third-party-dn`, described on page 43.
  - `vq-treat-unknown-third-party-dn-as-agent-dn`, described on page 43.
  - `ixn-id-in-status-table`, described on page 44.
  - `suppress-user-data`, described on page 58.
- Two new log sections: `sml` and `common`, described beginning on page 81.

- The re-inclusion of instructions for silently installing Stat Server, beginning on page 88.
- Two additional SQL scripts (`status_ixnid.sql` and `status_ltime_ixnid.sql`, described on page 102) for creating the `STATUS` table containing the `IxnID` field (described on page 112).
- Tips for manually purging the Stat Server database, beginning on page 115.

The *Framework 8.0 Stat Server User's Guide* describes other new features, such as filter improvements, support for additional ASM callflows, and a new statistic option (`UseSourceTimeStamps`) for configuring Stat Server to reference the timestamps of its client (instead of the time that Stat Server detects interactions) to better synchronize its recording of interactions with Interaction Concentrator. The *Supported Operating Environment Reference Manual* lists the operating systems (OSs) and RDBMSs on which Stat Server 8.0 can operate. And, the *Genesys Migration Guide* describes the high-level changes that were implemented with each release. And, *Genesys Administrator Help* describes the new XML template that you can use to configure a Stat Server 8.0 application.

Chapter

1 Overview of Stat Server

The Genesys Statistics Server (Stat Server) is part of the Genesys Management Framework Services Layer shown in Figure 1. Stat Server is the component responsible for converting single interactions into statistical data that is useful for interaction processing and contact center reporting. This data is then used by various Genesys solutions and applications to determine the availability of resources and to generate statistical indicators of contact center performance.

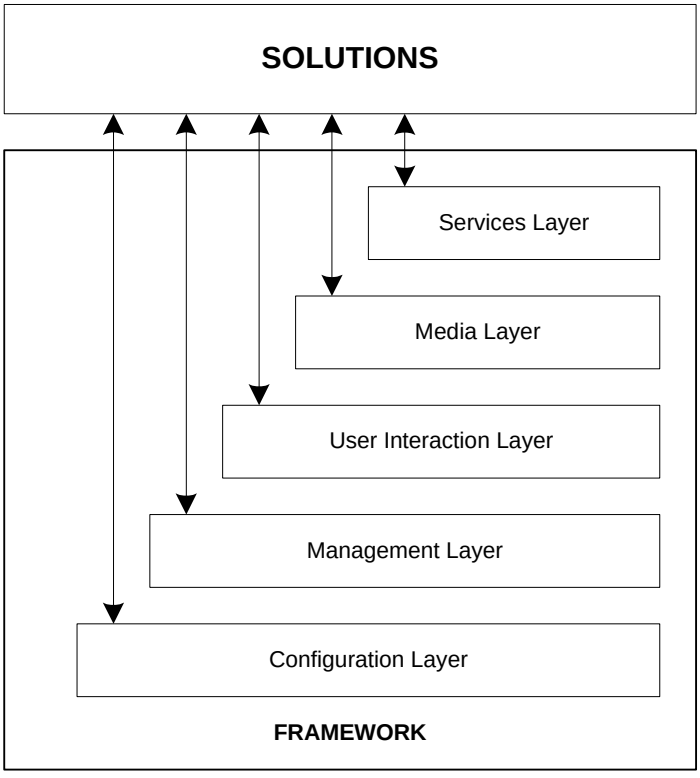


Figure 1: Management Framework Architecture

Refer to the *Framework 8.0 Deployment Guide* for more information about the Services Layer and Stat Server's role within the Genesys Management Framework.

As a client of T-Server and Interaction Server, Stat Server processes raw information received from these applications. As a client of Configuration Server, Stat Server retrieves information about the following contact center objects:

- Regular DNs
- Agents
- Places
- Queues
- Group of agents, places, and queues
- Routing Points
- Staging areas
- Tenants
- Workbins
- Switches
- Campaigns
- Calling lists

In short, Stat Server reconstructs the behavior of contact center objects in order to provide its clients with more elaborate and statistically useful reporting data.

To receive statistical information, Stat Server clients must specify the kind of data they need, following the Genesys Statistical Model described in the *Reporting Technical Reference Guide*. This specification consists of a request for statistics retrieval, via stat types, from the Stat Server application programming interface (API)—the Statistics Library, or Stat Lib for short. (This API is not documented.) For instance, the Universal Routing Server requests information for the purpose of monitoring virtual queues and determining agents' availability to process additional simultaneous interactions. And, the Genesys Outbound Contact Solution requests statistical information from Stat Server about the performance of its outbound campaigns and calling lists.

This *Framework 8.0 Stat Server Deployment Guide* describes the configuration and installation of the Stat Server application (and supporting topics) whereas the *Framework 8.0 Stat Server User's Guide* describes the configuration of statistics—their stat type definition. The *Reporting Technical Reference Guide* describes the application of Stat Server statistical types employed by the Solution Reporting applications—CCPulse+ and CC Analyzer. You can also reference other Genesys solution user guides for information about how the various solutions rely on Stat Server to provide statistical information.

Chapter

2 Stat Server Wizard

You use the Stat Server Wizard to guide you through the process of configuring a Stat Server application. You can also manually configure a Stat Server application without the benefit of using the Wizard. Manual configuration instructions are provided in Chapter 4 beginning on page 26. This chapter describes the following:

Installing the Stat Server Wizard

The Stat Server Wizard is one component of the Genesys Wizard Manager. You do not need to upgrade the Genesys Wizard Manager prior to installing the Stat Server Wizard. Installing the latest Stat Server Wizard automatically upgrades the common wizards set. You must, however, have a 7.0.1 or later release of the Genesys Wizard Manager already installed, and you must uninstall any previous releases of the Stat Server Wizard that may reside on your machine. If the Genesys Wizard Manager is not installed on your machine, you can install it from the `configuration_wizard` directory of the Management Framework CD.

`Note:` You can install Genesys wizards only on Windows platforms.

Before you operate other Genesys wizards that rely on the Stat Server component, install the Stat Server Wizard following these steps:

1. From the *Real-Time Metrics Engine* CD, go to either the root directory or the `configuration_wizard` subdirectory and run `Setup.exe`.
2. On the `Welcome` page, click `Next`.

3. On the Security Banner Configuration page, decide whether to configure a security banner for user entry to the Stat Server Wizard. If configured, this security banner will prompt the user to accept or reject conditions of use before the wizard is invoked.

   If you select the Enable Security Banner box, additional options appear on this page prompting you to apply this feature either to all applications that support it (such as other Genesys wizard applications) or only to the current application. Click Next.

4. If you opted to display a security banner in Step 3, configure the following on the Security Banner Parameters dialog box:
   - When to display the security banner during the next launch of the Resource Capacity Wizard (every time the application starts or at user's discretion).
   - How the application should process a login if the banner cannot be displayed (proceed or exit).
   - The title to display in the browser.
   - The amount of time that should elapse before the application displays a temporary message in the event the security banner has not yet loaded.
   - The height and width of the security banner.

   Tip: If you leave the height and width fields blank, the resulting browser window for the security banner will automatically resize to fit the content. If, however, you do specify values and later want to change them, you can edit the Height and Width entries directly in the Windows registry under:
   HKEY_LOCAL_MACHINE\SOFTWARE\GCTI\Unilogin\Banner

   For the first two configuration parameters, you can select from among predefined values in list boxes.

   If you opted not to display a security banner, skip to Step 7.

5. Near the bottom of the Security Banner Documents page, input the Universal Resource Locator (URL) of your security message, and add it to the list. Repeat this step for each subsequent security message that should be displayed. Then, click Next.

   Note: The installation routine does not validate URLs.

6. If you opted not to have the wizard invoked when the security banner cannot be displayed (Step 4, second bullet), input the URL of each of your error pages and add it to the list, or leave this field blank. Then, click Next.

   If you opted to proceed (Step 4, second bullet), this page, requesting the URL of your error pages, does not appear. Proceed to Step 7.

7. On the Ready to Install page, click Install.

   The InstallShield Wizard copies the wizard to the genesyscfgwizards directory.

8. On the Installation Complete page, specify how the host will be restarted, and then click Finish.

Modes of the Stat Server Wizard

At various points in the configuration of a Genesys solution or Solution Reporting, you can invoke the Stat Server Wizard to configure a Stat Server Application object. Or, you can invoke this Wizard directly within Configuration Manager.

The Stat Server Wizard is comprised of four modes:

- Stat Server Installation mode
- Stat Server Configuration mode
- Stat Server Upgrade mode
- Stat Server Uninstall mode

Configuring a backup Stat Server is also possible in all three modes. The manner in which you invoke the Stat Server Wizard, and the place from which you invoke it, determine the mode under which it operates. All modes modify or create a Stat Server Application object in Configuration Manager with defined T-Server connection(s), startup information, configuration options, and tenant and server identification.

Uninstalling the Stat Server Wizard

You must uninstall previous releases of the Stat Server Wizard prior to installing a new release. To uninstall the Stat Server Wizard:

1. Open the Control Panel and double-click Add/Remove Programs.
2. Select Genesys Stat Server Wizard from the list box and click Add/Remove.
3. Click Yes to confirm deletion.
4. Click OK.

The Uninstallation Wizard removes the program registry entries for the Stat Server Wizard. After reboot, all traces of the Stat Server Wizard are removed from the host.

Chapter

3 Optimizing Performance

Review the recommendations provided in this chapter to optimize Stat Server performance. This chapter contains these sections:

Hardware-Related Recommendations

When planning to deploy Stat Server to your environment, follow these recommendations:

- Consider the following formula, which approximates Stat Server memory, in megabytes, for a typical large contact center:

  MemoryReqd = 100 + (*NStatistics* x 0.0012)

  where *NStatistics* represents the number of open statistics and 0.0012 refers to approximately 1.2 KB of memory per statistic.

  For example, Stat Server on a box with 1.5 GB of memory should be more than ample to handle CC Analyzer requests of 30,000 active `Agent` or `Place` objects originating from the Genesys-provided Agent and Place reports):

NStatistics	= 28 statistics/report layout x 30,000 objects = 840,000 statistics
MemoryReqd	= 100 + (840,000 x 0.0012) = 1,108 MB

  For smaller contact centers, you can reduce the constant (100) to a smaller value.

  Install Stat Server on a computer with sufficient physical memory to avoid swapping.

- Consider distributing the total number of required statistics for Solution Reporting and real-time interaction processing for all solutions over a number of Stat Server applications.
- Install Stat Server and T-Server on the same machine or connect them through a fast LAN. If you are using several T-Server applications, position Stat Server nearest to the busiest T-Server.
- Do not install Stat Server on the same box as Configuration Server.
- Do not install real-time, third-party applications on the machine running Stat Server.
- If you want to store Stat Server data, consider dedicating a separate Stat Server application whose sole purpose is to write data to the Stat Server database.
- For large contact centers, consider allocating approximately 5 MB of space for each day Stat Server writes data to a database. This recommendation applies only if you configure Stat Server with a database access point and enable your Stat Server application to write data to a database by setting corresponding configuration options.

Software-Related Recommendations

You can improve Stat Server performance further by tuning Stat Server configuration options:

- Specify only the debugging log level that you need by setting the `debug-level` configuration option appropriately.
- For Stat Server applications that write to the Stat Server database, configure options only for the tables that you need by setting these configuration options:
  - `login-table`
  - `qinfo-table`
  - `status-table`
  - `voice-reasons-table`
- Review the configuration options that are related to write operations to this database:
  - For Oracle, Microsoft SQL, Sybase, and DB2 relational database management systems (RDBMSs), set the `enable-binding` option to `Yes`.
  - Set the `local-time-in-status-table` configuration option to `No` if you do not need a translation of UTC time to the time zone of the host on which Stat Server is deployed.
  - Set `ixn-id-in-status-table` to `No` for Solution Reporting and other clients that employ only an interaction's connection ID.

Chapter

4 Configuring a Stat Server Application

You must configure a Stat Server `Application` object before installing it. To configure Stat Server using the Stat Server Wizard (or manually), Configuration Server must be running. This chapter describes four methods for configuring a Stat Server `Application` object. It contains these sections:

`Note:` The security improvements offered in the release are not provided via the Stat Server Wizard. To use secure Transport Layer Security (TLS) connections between Stat Server and it clients, or between Stat Server and Configuration Server, you must configure such connections manually following the procedures described in the *Genesys Security Deployment Guide.*

Stat Server Wizard—Installation Mode

At various points during the wizard-driven configuration of Solution Reporting or of other Genesys solutions that rely on Stat Server data, you are presented with the opportunity to create and configure a new Stat Server `Application` object. Figure 2, for example, shows this starting point in the Data Sourcer and Universal Routing wizards. When you invoke the Stat Server Wizard in this manner, it operates in installation mode. You can also start the installation mode of the Stat Server Wizard by right-clicking an application folder in Configuration Manager and selecting `Wizard > Install > Stat Server`.

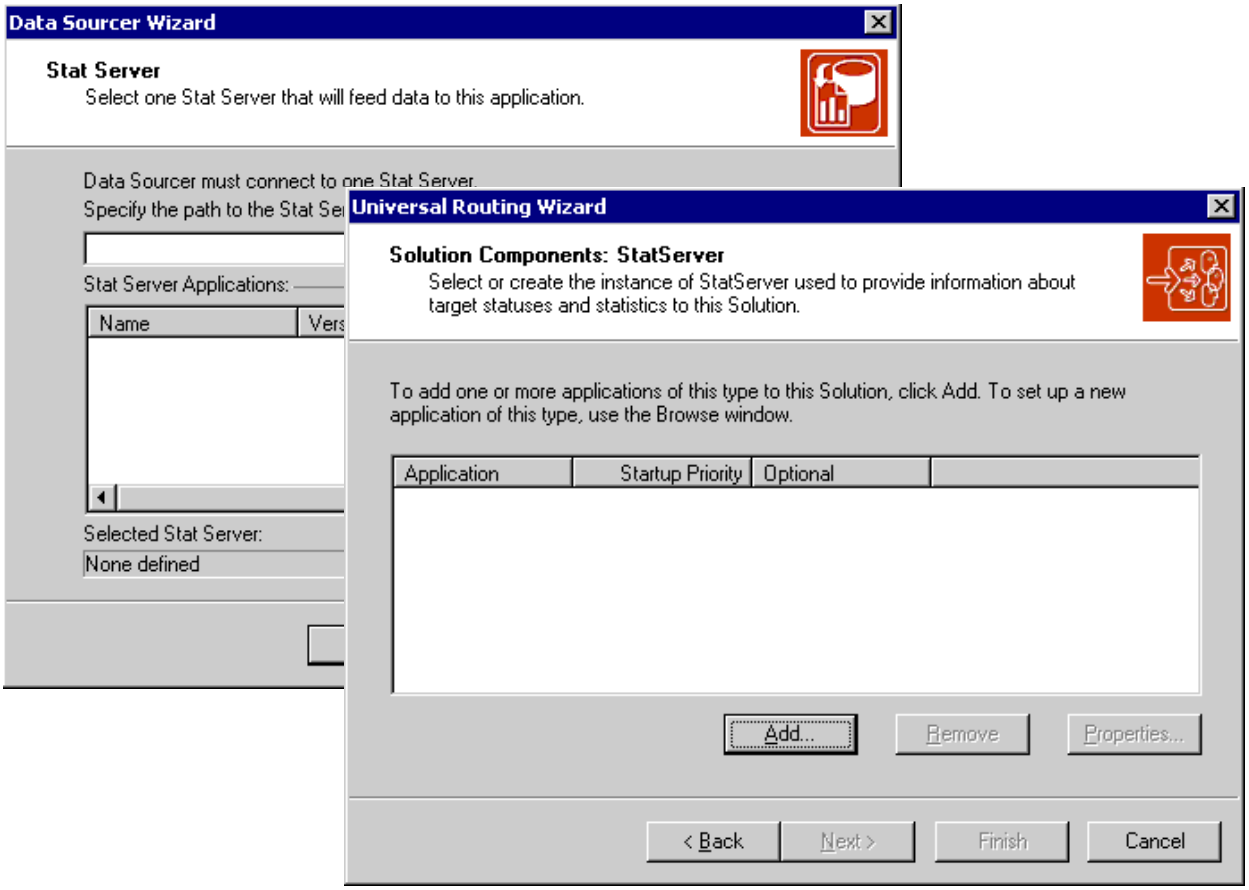


Figure 2: Invoking the Stat Server Wizard in Installation Mode

To operate the Stat Server Wizard in installation mode:

1. Click `Add` from the Stat Server page of the solution wizard (pictured in Figure 2 above).
2. On the `Welcome` page, click `Next`.
3. On the `Stat Server Name` page, type a unique name for your Stat Server `Application` object, and select a Configuration Server folder to store it in if you do not want to use the default. Click `Next`.
4. On the `Tenants Monitored by this Stat Server` page, select which tenants this Stat Server application is to monitor, and click `Next`.

   Note: This page does not appear in single-tenant environments.

5. On the `Switch Selection` page, select one or more T-Server applications that you want Stat Server to monitor, and click `Next`.
6. On the `Interaction Server selection` page, select the Interaction Server applications that you want Stat Server to monitor, and click `Next`.

7. On the `Log configurations` page, specify what information Stat Server is to log. If desired, click `Run Log Wizard...` to change the default options and/or the log output. Then, click `Next`.
8. On the `Java Configuration Options` page, specify whether Stat Server should support Java functionality. If so, select the `Load Java at startup` check box and then configure JVM (Java Virtual Machine) and Stat Server Java Extension parameters, and then click `Next`.

   Note: You need only configure a Java Runtime Environment if you intend to use Stat Server for Reporting in Outbound Contact 7.2+, eServices (formerly known as Multimedia) 7.0+, or VCB 7.1+.

9. On the `Server Info` page, select the host and port where you want Stat Server installed, and click `Next`. The wizard verifies that the indicated port is not already in use by another application.
10. On the `Installation Package` page:
    a. Specify the root of the *Real-Time Metrics Engine* CD.
    b. Specify where you want the installation package copied, and then click `Next`.

    Note: Remember the destination path you specify in this step. You will need it later (on page 83) to install Stat Server.

    The screen changes to display a progress-of-completion bar as the wizard deploys the installation package.
11. On the `Installation Ready` page, click `Next`.
12. On the `Backup Server Information` page, select the check box if you want to specify a backup for your primary Stat Server application. You do not configure the backup Stat Server `Application` object on this page—you merely identify it.

    After doing so, additional fields appear on this page prompting you to specify a name, host, port, folder, and redundancy type for your backup Stat Server application. Stat Server supports only the `Warm Standby` redundancy type. A backup Stat Server application operating in *warm standby* mode processes client requests only after its role changes from backup to primary. Stat Server does not support backup applications in *hot standby* mode, in which the client application connects to both servers upon startup and in which the primary and backup servers synchronize data between each other.

    Click `Next` to proceed.

> Note: Your backup Stat Server application must be created using the same application template as the primary Stat Server.

The Summary page appears, showing configuration highlights for the Stat Server Application object you just created.

13. On the final Wizard page, click Finish.

You have now configured a Stat Server Application object. The wizard creates a file called GCTISetup.ini containing the configuration parameters you specified, and places this file in the destination path indicated in Step 10. If you configured Java sections for your Stat Server Application object, be sure to specify this Stat Server application on the Connections tab of Application objects for the components of your Outbound Contact, eServices (formerly known as Multimedia) and/or VCB. Now, install the deployed installation package as described on page 83. Be sure to review the recommendations in the "Deploying with Configuration Wizards" chapter of the *Framework 8.0 Deployment Guide* if you plan to install Stat Server on a host other than the computer running the Stat Server Wizard.

Stat Server Wizard—Upgrade Mode

To benefit from improvements made from release to release, you should run the Stat Server Wizard in Upgrade mode to upgrade Stat Server Application objects. You need not first uninstall the Stat Server application to upgrade it but you do need to stop the application from running.

Before upgrading, Genesys recommends that you export your current configuration to a *.cfg or *.conf file and save this file to a secure location for use in the event you want to perform a rollback. Refer to *Framework 8.0 Configuration Manager Help* for information about using this utility.

1. In Configuration Manager, open the Properties page of a Stat Server Application object and, on either the General or Server Info tab, click the Upgrade button to invoke the Stat Server Wizard in Upgrade mode.

> Note: There are other ways to invoke this wizard in Upgrade mode including invoking it from the various Genesys solution wizards.

2. On the Welcome page, click Next. The Upgrading Application page appears, listing high-level information about the selected application.
3. Click Next.
4. If necessary, on the Server Information page, change the host and port information for the Configuration Server on which this application is located. Then, click Next.

5. On the Installation Package page:
   a. Specify the root of the *Real-Time Metrics Engine* CD.
   b. Specify where you want the installation package copied, and then click Next.

   Note: Remember the destination path you specify in this step. You will need it later (on page 83) to install Stat Server.

   A bar displays the progress of completion.
6. On the Upgrade Ready page, click Next.

   The Summary screen appears, showing configuration highlights for the Stat Server Application object that you just created, including any changes that you made to configuration options.
7. Click Finish.

You have now upgraded your Stat Server Application object. The wizard creates a file called GCTISetup.ini containing the configuration parameters you specified. If you configured Java sections for your Stat Server Application object, be sure to specify this Stat Server application on the Connections tab of the Application objects for the components of your Outbound Contact, eServices (formerly known as Multimedia), and/or VCB. Next, install the deployed installation package as described on page 83. Be sure to review the recommendations in the "Deploying with Configuration Wizard Framework" chapter of the *Framework 8.0 Deployment Guide* if you plan to install Stat Server on a host other than the computer running the Stat Server Wizard.

Stat Server Wizard—Configuration Mode

When you open the Properties page of an existing Stat Server Application object from a solution wizard or a Solution Reporting Wizard and then click Configure..., you invoke the Stat Server Wizard in Configuration mode. This Properties page has either six or nine tabs (depending on the location from which you invoke it), which you can visit in any order to alter the configuration of your Application object:

- General—Allows you to rename, upgrade, enable, and/or relocate your Application object.
- Tenants—Specifies the tenants that your Application object will monitor.
- Start Info—Specifies the working directory, executable, command-line arguments, startup and shutdown timeouts, and whether the application is to automatically restart upon shutdown.
- Server Info—Specifies the host and port of the machine where Stat Server resides. From this tab you can also specify a timeout value in seconds for unsuccessful attempts to connect to the server.

- Backup Server—Enables you to change the designated backup Stat Server application, disassociate the existing backup Stat Server application from the primary application, or invoke the Backup Stat Server Installation Wizard to associate a backup application with the primary application.
- Switches—Specifies one or more T-Server applications from which Stat Server gathers source data.
- Interaction Servers—Specifies one or more Interaction Server applications from which Stat Server gathers source data.
- Log Options—Specifies Stat Server log options.
- Java Options—Specifies the options that Stat Server uses and the Java extensions that Stat Server loads upon startup.

When you click the Apply or OK buttons on the Properties page, your changes are saved to the Configuration Server Database.

Note: Opening the Properties page of a Stat Server Application object in Configuration Manager displays the same tabs; however, this method does not invoke the Stat Server Wizard.

Manual Configuration

You can manually configure a Stat Server 8.0 Application object either within Configuration Manager or within Genesys Administrator. You use the Stat Server application template to accomplish this. This template is located in the templates directory of the *Real-Time Metrics Engine* CD and is named Stat_Server_800.apd. You should import this file into your configuration environment before configuring Stat Server applications.

Configuring a Stat Server Application Using Genesys Administrator

Another file in the templates directory of the *Real-Time Metrics Engine* CD, Stat_Server_800.xml, contains the metadata that defines the default and valid values for most of the Stat Server configuration options that are available to you in the 8.*x* release. (The complete listing of configuration options is located in this document and accompanying release notes.) To use this metadata, you import it into the Stat Server application template. As you configure a Stat Server application within Genesys Administrator, Genesys Administrator validates the values that you specify for configuration options based on this metadata.

Refer to the "Application Templates" book within *Framework 8.0 Genesys Administrator Help* for instructions on how to import and use metadata and for instructions on how to configure applications within Genesys Administrator.

Note: Metadata cannot be used in conjunction with the Stat Server application template in Configuration Manager.

Configuring a Stat Server Application Using Configuration Manager

To manually configure a Stat Server application within Configuration Manager:

1. From the main window, open the Applications folder.
2. From the File menu, select New > Application, which opens the Browse page with the available application templates.
3. Choose the template that corresponds to the Stat Server release you are installing—for example, Stat_Server_800.apd.

   If you cannot locate this template, import it from the *Real-Time Metrics Engine* CD.
4. On the General tab of the Properties page, enter a name for your Stat Server application. The application template provides information about the Stat Server type and version.
5. For multi-tenant configurations, on the Tenants tab, specify the tenants that you plan to have Stat Server monitor. This tab does not appear in single-tenant environments.
6. On the Server Info tab of the Properties page, specify:
   a. The host on which you want to install this server.
   b. The communication port that applications will use to connect to this server.
   c. The reconnection timeout, in seconds, for which the client application waits before attempting reconnection to this server following a failed connection.
   d. (Optional) The name of the backup Stat Server application.
   e. (Optional) Warm standby, in the Redundancy Type field, to indicate the state of readiness of the backup Stat Server application. A backup Stat Server in *warm standby* mode processes client requests only after its role changes from backup to primary. Stat Server does not support backup applications in *hot standby* mode, in which the client application connects to both servers upon startup and in which the primary and backup servers synchronize data between each other.

7. On the Start Info tab, provide the working directory and command-line parameters. For information about command-line parameters, refer to Chapter 9.
8. On the Connections tab, add the following Application objects:
   - One or more T-Server and/or Interaction Server application(s).
   - Message Server, if you plan to use the Management Layer for alarm-signaling and centralized logging capabilities.
   - Database access point, if you plan to use Stat Server to populate Stat Server–related database tables.

   Note: If you specify more than one database access point, Stat Server attempts to connect DB Server using only the first one listed, even if the first is unavailable.
9. On the Options tab, specify the values of the desired configuration options in appropriate sections. For option descriptions, refer to
   - Chapter 5 on page 29 of this guide for a listing of configuration options that pertain to operation of the Stat Server application.
   - The *Framework 8.0 Stat Server User's Guide* for configuration options pertaining to statistics.
10. If you configured Java sections for your Stat Server Application object, be sure to specify this Stat Server application on the Connections tab of your Outbound Contact, eServices, and/or VCB Application object.

Stat Server Wizard—Uninstall Mode

Operating the Stat Server Wizard in Uninstall mode removes the Application object from the Configuration Server Database. This wizard does not uninstall any application files, logs, directories, or the application executable itself. Refer to "Uninstalling the Stat Server Application" on page 93 for those instructions.

To remove a Stat Server Application object from Configuration Server:

1. From Configuration Manager, select the Stat Server Application object you want to uninstall.
2. Right-click and select Wizard > Uninstall from the shortcut menu that appears.

   The Stat Server Uninstall Wizard opens.
3. On the Welcome page, click Next.
4. On the Uninstalling Application page, verify the application you are about to uninstall and click Next.
5. On the Uninstall Complete page, click Finish.

Chapter

5 Fine-Tuning Stat Server Configuration

This chapter describes the options you can use to configure a Stat Server application. Refer to the next chapter, "Other Factors Affecting Stat Server" on page 57 for descriptions of options in other Genesys applications that impact Stat Server behavior and Chapter 7, "Common Log Options," on page 61, for descriptions of log configuration options common to most Genesys 8.0 server applications. Finally, to learn about the options you can use to configure statistics for your Stat Server application, refer to the "Statistics Configuration Options" chapter in the *Framework 8.0 Stat Server User's Guide.*

The information in this chapter is divided among the following topics:

Note: The configuration options related to Stat Server logging are described separately in Chapter 7.

Mandatory Options

You are not required to configure any options to start Stat Server.

StatServer Section

You must name this section `statserver`. Table 1 lists some options available for configuring Stat Server `Application` objects. Use the options listed in Table 2 on page 44 to configure a Stat Server application to write data to a database. And use the options listed in Tables 4 and 5, beginning on page 50, to enable Java functionality. You are directed to read your operating-system and/or JRE vendor documentation to learn of any peculiarities regarding JVM installation or the setup of a Java Runtime Environment.

For those configuration options that indicate `true` or `false` values, any of the following additional values are also valid:

- `t/f`
- `yes/no`
- `y/n`
- `1/0`
- `on/off`

These alternates may not be indicated within the following tables. Also, the default values listed in the tables refer to those that are either inherent to Stat Server or in the Stat Server application template (`Stat_Server_800.apd`, for example), which is provided on the *Real-Time Metrics Engine* CD. These default values may differ from what is documented here if you configure a new Stat Server `Application` object using the Stat Server Wizard.

`Note:` Where the name of a configuration option changed between releases, Stat Server continues to support the former name.

Table 1: Configuration Options for the statserver Section

Option	Description
accept-clients-in-backup-mode	Specifies whether Stat Server accepts client connections when operating in backup mode. With this option set to `yes`, Stat Server notifies the clients about its redundancy mode after a client's registration and after a change in mode. Moreover, when its redundancy mode is changed to backup, Stat Server does not close the communication port and accepts clients' connections and requests. Default Value: `no` Valid Values: `yes, no` Changes Take Effect: When Stat Server is restarted

Table 1: Configuration Options for the statserver Section (Continued)

Option	Description
allow-vq-orig-dns-from-environment	Specifies whether Stat Server will generate retrospective actions, reflecting regular DNs, to virtual queue objects that belong to the `Environment` tenant. If set to `yes` and the `Environment` tenant is listed among those assigned to Stat Server (in a multi-tenant environment), Stat Server will generate such actions when these virtual queues are also assigned as origination DNs to `GroupAgents` and `GroupPlaces` objects belonging to a particular tenant (that is, to a tenant that is not the `Environment` tenant) as a result of call activity on that tenant. For this functionality to work properly, you must also set the `vq-treat-unknown-third-party-dn-as-agent-dn` option to `true` and the `vq-ignore-third-party-dn` option to `false`. Default Value: `yes` Valid Values: `yes, no` Changes Take Effect: When Stat Server is restarted
auto-backup-interval	Sets the time, in minutes, for checking persistent statistics and storing them in the file specified by the `backup-file-name` option. This option was previously named `AutoBackupInterval`. Default Value: `15` Valid Values: Integers `0` through `2147483647` ($2^{31}-1$) Changes Take Effect: When Stat Server is restarted
backup-file-name	Specifies the name of the backup file that stores persistent statistics for synchronization. Stat Server memorizes all parameters for statistics in demand, initiating their collection immediately after restart. If a particular statistic has not been requested for a long time period (three days by default as specified in the `old-stats-remove-interval` option), the statistics are removed from both the cache and the backup file. **Note:** Stat Server ignores backup files that were generated by a different version of Stat Server. This option was previously named `BackupFileName`. Default Value: `ssbackup.000` Valid Value: Any valid file name Changes Take Effect: When Stat Server is restarted

Table 1: Configuration Options for the statserver Section (Continued)

Option	Description
capacity-treat-acw-as-interaction	Determines whether Stat Server treats interactions initiated or received as interactions while the associated DN is in after-call work (ACW) status. The routability of additional, simultaneous interactions to a device is dependent on the number of interactions currently occurring at that device. Setting this option to `yes` instructs Stat Server to treat any ACW activity as an interaction *for the purpose of determining capacity*—synonymous to any other type of voice interaction, such as handling customer-initiated (inbound) calls, internal calls among agents, and so forth. *For the purpose of reporting current activity*, this treatment does not increment the count of `CurrentNumber` or `TotalNumber` statistics. The presence of ACW on a device also impacts the routability of interactions of other media types, as defined in the capacity model for your environment. For information about defining capacity rules, refer to the *Genesys 8.0 Resource Capacity Planning Guide*. If this option is set to `no`, Stat Server does not consider ACW-related activities occurring at a device in its calculation of the `current_number` component of the capacity vector. In fact, Stat Server may allow additional, simultaneous interactions to be routed to that device per the capacity rules defined in your environment. Default Value: `no` Valid Value: `yes, no` Changes Take Effect: When Stat Server is restarted
check-stuck-calls	When you set the value of this option to `yes`, Stat Server checks DNs of the `Extension, ACD Position, VTO` (IVR), `ACD Queue, Service Number,` and `Routing Point` types for calls with no activity during the last 10 minutes. When detecting such calls, Stat Server queries T-Server on the current DN status. If T-Server indicates that the call has been cleared from the DN in question, Stat Server deletes the call from memory. The checkup interval is 600 seconds (10 minutes). Stat Server does not check Internet DNs or DNs of `Virtual Routing Point` type. This option was previously named `CheckStuckCalls`. **Note:** Calls can be stuck in T-Server and/or Stat Server. The `check-stuck-calls` configuration option enables Stat Server to clear calls that it determines to be stuck in Stat Server. Default Value: `no` Valid Values: `yes, no` Changes Take Effect: Immediately upon notification

Table 1: Configuration Options for the statserver Section (Continued)

Option	Description
check-vq-stuck-calls-frequency	Works in conjunction with the frequency of `EventReserved_2` heartbeats that accompany live calls from URS, specifying the frequency, in seconds, with which Stat Server checks virtual queues for stuck calls. Stat Server suspends checks for stuck calls if no calls are queued at any virtual queue that Stat Server monitors. Stuck calls result if a URS connection breaks when a live calls is completed. In this situation, Stat Server does not receive the `EventReleased` TEvent to indicate the end of the call, and Stat Server views the interaction as continuing. Setting this option to a high value for large environments alleviates CPU load and helps to avoid situations where Stat Server inadvertently clears live calls due to network latency. In such situations, you should also consider resetting the `call_kpl_time` URS option, which measures the frequency of `EventReserved_2` heartbeats, to a higher value. Small environments can set both options to relatively lower values. Also, you should be aware that setting `call_kpl_time` to `0` (zero) means that Stat Server will receive no `EventReserved_2` events from URS. In this case, Stat Server considers all calls currently residing at the virtual queue as stuck and eliminates them from processing following the period of time specified by the `check-vq-stuck-calls-frequency` option. Refer to the *Universal Routing 8.0 Reference Manual* for additional information about the `call_kpl_time` configuration option. Prior to release 8.0, this functionality was not configurable. Stat Server's checks for stuck calls was internally hard-coded at `60` seconds. Beginning with the 8.0 release, the default for this option, whether explicitly set or not, is `600` seconds. Default Value: `600` Valid Values: `30` to `2147483` (just over 24 days; `2147483=`2^{31}`/1,000`) Changes Take Effect: When Stat Server is restarted

Table 1: Configuration Options for the statserver Section (Continued)

Option	Description
debug-level	A comma-separated list of debug levels that you can see in the Stat Server log. This option is enabled only if you have set the `verbose` common log option to `all`. In graphical environments, log output often takes more than half of a server's execution time. To maintain performance, use only the debug levels that you need and run Stat Server in the background. Also, minimize the Stat Server window or redirect log output to a different device, such as a file. Be very careful, however, when directing log output to a file and consider the available free disk space, directory and file permissions, and possible conflicts with different software trying to use the log file at the same time. This option was previously named `DebugLevel`. Default Value: `Init, Client, ClientX, Server, Action, Status` Valid Values: `all` Synonymous with `Init, Server, Client, ClientX, Status, Action, SQL, Mngmnt, Java, Reset`. `Action` Logs changes to the internal Stat Server object model and provides a significant source of troubleshooting data, which includes entries following every TEvent. `Client` Logs all Stat Server communication with its clients, such as the opening of statistics and all statistical values sent to the client. This value generates a large amount of data, and should be sparingly used for troubleshooting reproducible problems with statistics. **Note:** Specifying this value alone no longer logs `StatValid` or `StatInvalid` messages as was done in previous releases. For Stat Server to log equivalent messages, you must specify the `ClientX` value in conjunction with this option. `ClientX` Logs `StatValid` and `StatInvalid` messages to indicate whether the named statistic is valid. Specifying this value alone excludes Stat Server's logging of other statistic-related communications, such as the opening of statistics and statistical values. `Ext` For internal use only.

Table 1: Configuration Options for the statserver Section (Continued)

Option	Description	
debug-level (continued)	`Init`	Used for capturing data related to Configuration Server that affects Stat Server, including dynamic Configuration Server changes made as Stat Server starts—such as the addition, deletion, and/or change of objects or their properties having an affect on Stat Server. This value is useful for tracking initial configuration and dynamic changes and is much more compact than the information provided in the Configuration Server log. Genesys recommends that you always include this value in this option.
	`Java`	Displays information related to Java extension functionality. Use this value only for statistics in the Outbound Contact 7.2.0+, MCR 7.0.1+, or VCB 7.1.0+. **Note:** MCR has been renamed to eServices in release 8.0.
	`Mngmnt`	Displays profiling information, including the number of currently connected clients, statistics being computed at the moment, and statistics to be reported to clients.
	`Reset`	Enables the log messages Stat Server sends to clients while sending statistics requested with a reset-based notification mode.
	`Server`	Logs T-Server events pertaining to Stat Server. Genesys recommends that you not include this value if you maintain logs for the related T-Server(s).
	`SPT`	Logs events related to Stat Server startup. This value is provided to maintain backward compatibility and may be eliminated in future releases.
	`SQL`	Displays the SQL statements issued if you have configured a database for Stat Server.
	`Status`	Logs events related to the current state of objects and is useful for troubleshooting Stat Server–Router problems.
	Changes Take Effect: Immediately upon notification	

Table 1: Configuration Options for the statserver Section (Continued)

Option	Description
DefaultAgentSPT, DefaultDNSPT, DefaultRPSPT	These options create a precedence list of actions, which Stat Server uses to assign status to DNs, agents, places, or routing points, when there is more than one action occurring at each point. The `DefaultGroupSPT` and `DefaultGroupRPSPT` options available in the initial 7.0 release and prior releases are no longer required. For information on the operating mechanism of Status Priority tables, refer to the "Object Statuses" chapter in the *Framework 8.0 Stat Server User's Guide.* **Warning!** Do not change these options without consulting a Genesys technical representative. Default Value: . . . (an ellipsis) Valid Value: A list of actions separated by a comma or an ellipsis. If you specify a list, it overrides the list hard-coded in the Stat Server Status Priority tables. Changes Take Effect: When Stat Server is restarted
do-backup-in-background	Specifies whether Stat Server spawns a separate thread to store statistic definitions in its backup file. If set to `yes` (the default), Stat Server spawns a separate thread. If set to `no`, Stat Server writes to its backup file using the main thread. Default Value: `yes` Valid Values: `yes, no` Changes Take Effect: When Stat Server is restarted
emulate-acw-for-mlink	Enables Stat Server support for the `AfterCallWork` (ACW) implementation in T-Server for the Nortel Meridian 1 switch. This option has been provided to maintain backward compatibility and is necessary only if you are using a version of Mlink 6.5. You do not have to set this option for Mlink 7.0 and subsequent versions. This option was previously named `EmulateACWForMLink`. The default template does not include this option nor does the Stat Server Wizard set it. Default Value: `no` Valid Values: `yes, no` Changes Take Effect: When Stat Server is restarted

Table 1: Configuration Options for the statserver Section (Continued)

Option	Description
filters-allow-wildcards-in-values	Specifies whether Stat Server accepts the wild-card characters * and ? in the <value> argument of PairExist functions in filters. If set to yes, Stat Server interprets these characters as wildcards. If set to no, Stat Server interprets these as literal characters. Prior to release 7.5, Stat Server interpreted a <value> argument of "*" as *any string* and "*", embedded within a string, as a literal character. For example, Stat Server interprets the PairExist(KY1, "Mr.*") function in one of two ways depending on the value of the filters-allow-wildcards-in-values option: • As a function whose filter returns any statistic where the values for KY1 begin with Mr., if the value of this option is set to yes. • As a function whose filter returns only those statistics where the value for KY1 is equivalent to the four characters Mr.* if the value of this option is set to no. Stat Server interprets the PairExist(KY2, "*") filter as one where KY2 is equal to any number of characters regardless of the value of this option. Default Value: no Valid Values: no, yes Changes Take Effect: When Stat Server is restarted
ignore-disabled-objects-in-group-statistics	Specifies whether Stat Server takes into account the Person and Place objects that are disabled in the Configuration Layer when calculating statistics for corresponding groups of objects. Setting this option to yes excludes agents and places in the calculation of group status for CurrentState statistics as long as the relevant Person and Place configuration objects are disabled in the Configuration Layer. This option also affects any number-related group statistics in the same manner. Genesys recommends setting this option to yes for a Stat Server application serving Universal Routing Server, and no (the default) for a Stat Server application serving CCPulse+ and/or CC Analyzer. Default Value: no Valid Values: yes, no Changes Take Effect: When Stat Server is restarted

Table 1: Configuration Options for the statserver Section (Continued)

Option	Description
ignore-disabled-objects-in-queue-statistics	Specifies whether Stat Server takes into account disabled `Person` and `Place` objects when calculating certain queue statistics. When this option is set to `yes`, Stat Server abstains from updating queue statistics having any of the `ActionLogin`, `AgentReady`, and `AgentActive` actions in the main mask while `Person` and `Place` objects continue to be disabled. When set to `no`, Stat Server considers all `Person` and `Place` objects—disabled or enabled—and all masks in computations of queue statistics. **Note:** The aforementioned statistics are such that actions are propagated from a place to a queue only when an agent is logged on to the place and either of the following: • Both the agent and place are enabled, in which case the value of this configuration option is irrelevant. • The agent and/or place is disabled and the value of this option is set to `no`. Default Value: `no` Valid Values: `yes, no` Changes Take Effect: When Stat Server is restarted
ignore-off-hook-on-position	Specifies whether to ignore `On-Hook/Off-Hook` events on `Position` DNs. If set to `yes`, `On-Hook/Off-Hook` events are ignored on `Position` DNs. Use this option if your version of T-Server does not properly propagate `On-Hook` or `Off-Hook` TEvents. This option was previously named `IgnoreOffHookOnPosition`. Default Value: `no` Valid Values: `yes, no` Changes Take Effect: When Stat Server is restarted
load-balance-aht	Specifies the initial value, in seconds, for handling time. Stat Server uses this figure in the operand of the formula for calculating load-balancing of all mediation DNs that Stat Server monitors. Refer to the `LoadBalance` statistical category in the *Stat Server User's Guide* for more information about this formula. You can also configure this value within the application options of mediation DN objects. Values specified at the mediation DN level supersede the value specified within the Stat Server `Application` object for that mediation DN. Refer to page 60 of this document for information about configuring this option for mediation DNs. Prior to release 8.0, this value was hard-coded at `90` seconds. Default Value: `90` Valid Values: Positive integers less than `4294967296` (2^{32}) Changes Take Effect: When Stat Server is restarted (for the global `load-balance-aht` option).

Table 1: Configuration Options for the statserver Section (Continued)

Option	Description
management-port	Specifies the TCP/IP port that Stat Server reserves for connections that its SNMP Option Management Client establishes. **Warning!** You must specify a value for this option if you are using an SNMP connection. Do not change the value for this option while Stat Server is running. Default Value: `3031` Valid Values: Any available TCP port (Integers `0` through `65535`) Changes Take Effect: When Stat Server is restarted
max-client-connections	Specifies the maximum number of clients that can be connected to Stat Server at any given time. The default value, `-1`, or `0` (zero) indicates that an unlimited number of clients can be connected to Stat Server. Default Value: `-1` Valid Values: `-1`, `0`, or any positive integer less than `2147483648` (2^{31}) Changes Take Effect: When Stat Server is restarted
nec-position-extension-linked	Specifies whether Stat Server applies a special model when processing after-call work (ACW) notifications from NEC T-Server. This model, normally used with Meridian T-Server, consists of Position and Extension DNs linked together in Stat Server logic when they belong to the same phone. Refer to the *Framework 8.0 Stat Server User's Guide* for a description of the `AfterCallWork` action and models for its generation. The option name is case-insensitive. Default Value: `no` Valid Values: `yes`, `no` Changes Take Effect: When Stat Server is restarted
old-stats-remove-interval	Sets the amount of time, in minutes, that unused statistics should continue to calculate on Stat Server. A value of `0` causes Stat Server to close a statistic as soon as the application requesting it closes its request or disconnects. This option was previously named `OldStatsRemoveInterval`. The default template does not include this option nor does the Stat Server Wizard set it. The internal, hard-coded default value is `4320` (three days). Default Value: `4320` Valid Values: Integers `0` through `2147483647` ($2^{31}-1$) Changes Take Effect: When Stat Server is restarted

Table 1: Configuration Options for the statserver Section (Continued)

Option	Description
position-extension-linked	Specifies how Stat Server interprets the status of a place and an agent when the place contains a position and an extension that belong to the same switch. By default (`yes`), the status of a DN of the `Extension` type affects the place status under these conditions: • An agent is logged in at the DN of the `Position` type that belongs to the same place. • An agent may or may not be logged in at the DN of the `Extension` type. With the option set to `no`, the status of the DN of the `Extension` type affects the place status under these conditions: • An agent may or may not be logged in at the DN of the `Position` type that belongs to the place. • An agent *must* be logged in at the DN of the `Extension` type. Default Value: `yes` Valid Values: `yes, no` Changes Take Effect: When Stat Server is restarted
queue-use-pseudo-actions	Restricts Stat Server's use of the following mediation DN actions to unfiltered statistics defined using the `CurrentNumber` or `CurrentRelative NumberPercentage` statistical categories: • `DNLogin` • `DNActive` • `DNReady` • `AgentLogin` • `AgentActive` • `AgentReady` If set to `true`, Stat Server enables this restriction and minimizes the possible overhead that could results in environments that contain a large number of origination DNs assigned to `GroupPlaces` or `GroupAgents` objects. If set to `false`, all statistical categories and filters will be applicable to the listed actions. Default value: `false` Valid values: `true, false` Changes take effect: Upon restart of Stat Server

Table 1: Configuration Options for the statserver Section (Continued)

Option	Description
reconnect-timeout	Indicates the time interval, in seconds, between Stat Server attempts to reconnect to a T-Server or DB Server if either is disconnected or not running. This option was previously named `reconnect_timeout` (spelled with an underscore). Default Value: `10` Valid Values: Positive integers less than `4294967296` (2^{32}); Stat Server sets any negative or `0` values that you might configure to `1`. Changes Take Effect: When Stat Server is restarted
reg-delay	Causes Stat Server to wait the specified number of seconds before registering DNs that have been added in Configuration Manager. This option was previously named `reg_delay` (spelled with an underscore). Default Value: `3` Valid Value: `0` (zero) or positive integers less than `4294967296` (2^{32}) Changes Take Effect: When Stat Server is restarted
reg-dns-chunk-delay	Specifies the interval, in seconds, between two subsequent DN registration requests. Stat Server waits for the specified interval before sending a request to T-Server to register a subsequent set of DNs, thus allowing T-Server to process the previous request. In a large configuration environment, use this option in conjunction with `reg-dns-chunk-volume` to optimize DN registration at Stat Server startup. Default Value: `10` Valid Values: Positive integers less than `4294967296` (2^{32}) Changes Take Effect: When Stat Server is restarted
reg-dns-chunk-volume	Specifies the number of DNs that Stat Server submits in a single registration request to T-Server. Instead of trying to register for all configured DNs at once, Stat Server divides the DN registration among several requests, each for the specified number of DNs. In a large configuration environment, use this option in conjunction with `reg-dns-chunk-delay` to optimize DN registration at Stat Server startup. Default Value: `1000` Valid Values: Any positive integer less than `2147483648` (2^{31}) Changes Take Effect: When Stat Server is restarted

Table 1: Configuration Options for the statserver Section (Continued)

Option	Description
send-timeout	Specifies the interval, in seconds, that Stat Server keeps client requests in the output queue. When this timeout expires for a given client request, Stat Server disconnects this client as being "too slow." Consider increasing this option's value in an environment with a slow network or where client disconnects are frequent. Default Value: `300` Valid Values: `0-3600` (1 hour) Changes Take Effect: When Stat Server is restarted
show-attached-data	If set to `yes`, Stat Server outputs call-extracted `UserData` to the Stat Server log. If set to `no`, Stat Server stops outputting attached data to its log—regardless of the log-level setting (`trace`, `debug`, and so forth). T-Server propagates attached data (`UserData`) by way of TEvents; this data is used for internal computations. To output `UserData` to the log, in the `Filters` section of the Stat Server application object, add a `PairExist("key","value")` filter where `key` is the name of the `UserData` key; `value` may denote a specific value or `"*"`. Setting this option does not affect Stat Server's processing of `UserData`. For memory, performance, and security reasons, however, Stat Server strips away any attached data that is not directly used for internal computations. Refer to the `UserData` property in the "Call Properties" table of the *Stat Server User's Guide* for more information. Default Value: `no` Valid Values: `no, yes` Changes Take Effect: Immediately upon notification
suppress-agent-status-updates-for-ixn-server	Enables suppression of `EventCurrentAgentStatus` notifications by Stat Server in environments that deploy multiple Stat Server applications. Disabling this statistic request from select clients avoids situations in which Stat Server clients receive multiple and identical notifications about current status for the same agent. Default Value: `no` Valid Values: `no, yes` Changes Take Effect: Immediately upon notification
suppress-user-data	You configure this option in the `[statserver]` section on the `Annex` tab of switch and/or DN objects. Refer to "To Suppress the Transmission of Attached Data" on page 58 for the description and permissible values of this option.

Table 1: Configuration Options for the statserver Section (Continued)

Option	Description
vag-statistics-active-agents-only	Limits the membership of virtual agent groups to only those active agents satisfying a particular script condition. (An active agent is `Person` object that has been enabled in Configuration Server.) Default Value: `no` Valid Values: `yes, no` Changes Take Effect: When Stat Server is restarted
vq-ignore-third-party-dn	Controls whether Stat Server relies on the `ThirdPartyDN` attribute of `EventDiverted` TEvents to determine the DN to which a call was diverted from a given virtual queue. Default Value: `true` Valid Values: `true, false` Changes Take Effect: When Stat Server is restarted
vq-treat-unknown-third-party-dn-as-agent-dn	Indicates whether Stat Server generates the `CallAnswered` action for virtual queue objects in the following scenario: 1. Stat Server receives an `EventDiverted` TEvent for the virtual queue. 2. The `ThirdPartyDN` attribute value of this TEvent contains the ID of an unknown DN—one that is monitored by a switch other than that to which the virtual queue belongs. 3. The call is subsequently routed to an agent. If this option is set to `true`, Stat Server generates the `CallAnswered` action under the preceding circumstances. If this option is set to `false`, Stat Server does not generate this action under the same circumstances. If the `ThirdPartyDN` attribute value is null or contains an ID that coincides with that of the answering DN, Stat Server generates the `CallAnswered` action on virtual queue objects, regardless of this option's setting. **Note:** The `vq-ignore-third-party-dn` option must be set to `false` in order for Stat Server to consider the value of this option. Default Value: `yes` Valid Values: `yes, no` Changes Take Effect: When Stat Server is restarted

Table 2: Configuration Options for Operating Stat Server with a Database

Option	Description
binding-threshold	Specifies the number of records in a binding block—that is, the number of records to be sent to the DBMS simultaneously. This option is enabled only if you have set the value of the `enable-binding` configuration option to `yes`. The default template does not include this option, nor does the Stat Server Wizard set it. The value of this option has no effect for Informix RDBMSs. Default Value: `10` Valid Values: Any positive integer less than `2147483648` (2^{31}) Changes Take Effect: When Stat Server is restarted
enable-binding	Specifies whether to enable binding functionality. By default, Stat Server uses a regular method of sending requests. If you set the value of this option to `yes`, Stat Server uses binding for sending requests. This option works in conjunction with the `binding-threshold` configuration option and is supported for Oracle, Microsoft SQL, Sybase, and DB2 relational database management systems. The default template does not include this option nor does the Stat Server Wizard set it. This option was previously named `OracleBinding`. Default Value: `no` Valid Values: `yes`, `no` Changes Take Effect: Immediately upon notification
ixn-id-in-status-table	Specifies whether Stat Server will populate the `IxnID` field for records written to the `STATUS` table. If you set this option to `off`, or if you do not configure this option, the `IxnID` field will be null. This field provides functionality, comparable to connection IDs for calls, for Multimedia interactions that rely predominantly on the number generated by Interaction Server for identification in the interaction flow. **Note:** If you set this option to `on`, consider also setting the `multimedia-activity-in-status-table` configuration option to `yes`, so that Stat Server will record information about the status of multimedia interactions in the other fields of the `STATUS` table. Default Value: `off` Valid Values: `on`, `off` Changes Take Effect: Immediately upon notification Refer to Appendix A, “Physical Data Models for Stat Server Tables” on page 105 for a complete description of the `STATUS` table. **Warning**! To avoid data loss, do not change the setting of this option in runtime if you have also set `enable-binding` to `yes`.

Table 2: Configuration Options for Operating Stat Server with a Database (Continued)

Option	Description
local-time-in-status-table	Specifies whether to populate the `StartLocalTime` and `EndLocalTime` fields in the `STATUS` table. If you set the value of this option to `off`, or if you do not specify a value, the `StartLocalTime` and `EndLocalTime` fields will contain no data. For Solution Reporting applications, set this option to `off`; such reports do not use the local time fields, and setting this option to `on` could impact performance. When setting this option to `on`, also set the `time-format` option to the desired format. Refer to page 110 for a complete description of the `STATUS` table. This option was previously named `LocalTimeInStatusTable`. Default Value: `off` Valid Values: `on`, `off` Changes Take Effect: Immediately upon notification **Warning!** To avoid data loss, do not change the value of this option in runtime if you have also set `enable-binding` to `yes`.
login-table	Specifies whether Stat Server writes records about login and logout TEvents directly to the `LOGIN` table in the Stat Server database. Refer to page 108 for a complete description of this table. This option was previously named `LoginTable`. Default Value: `off` Valid Values: `on`, `off` Changes Take Effect: Immediately upon notification
max-unsent-sql-statements	Specifies the maximum number of SQL statements that Stat Server is allowed to maintain in memory. As soon as Stat Server's connection to the RDBMS is broken, Stat Server starts storing SQL statements in memory. These statements will be issued against the Stat Server database once the connection is restored. If the number of SQL statements in memory exceeds the value specified by this option, data loss may result. To avoid data loss, Stat Server must remain connected to DB Server for the entire period that DB Server submits records to the RDBMS. Your addp timeout for connection from Stat Server to DB Server should be set as large as possible to prevent disconnection by addp. Refer to the *Framework 8.0 Configuration Options Reference Manual* for information about setting addp.

Table 2: Configuration Options for Operating Stat Server with a Database (Continued)

Option	Description
max-unsent-sql-statements (continued)	If the number of SQL statements in memory ever exceeds this option's value, data loss of the entire memory pool will result and the accumulation of SQL statements will begin anew. Please be aware that setting this option's value too high may cause your system to run out of memory. Configure this option in conformance with the amount of RAM installed on the machine where Stat Server operates. If you specify any value that is less than the default (`100000`), Stat Server resets it to `100000`. Default Value: `100000` (SQL statements) Valid Values: Integers greater than or equal to `100000` and less than `2147483648` (2^{31}). Changes Take Effect: When Stat Server is restarted
multimedia-activity-in-status-table	Specifies whether multimedia-related actions are counted while computing status values that are written to the `STATUS` table. (For a complete classification of actions, refer to the *Stat Server User's Guide*.) If this option is set to `no`, Stat Server ignores multimedia-related actions in its computation of place and agent status. Stat Server also reads the value of the `multimedia` configuration option in the `TServer` section of the monitored DN (whose type is `Extension`) to determine whether the corresponding DN is a multimedia DN, capable of processing interactions of different media types, such as those DNs that are controlled by a SIP-compliant T-Server. Refer to page 58 for more information. **Note:** If you set this option to `yes`, you might also consider setting the `ixn-id-in-status-table` configuration option to `yes` so that Stat Server populates the `IxnID` field for multimedia interactions. Default Value: `yes` Valid Values: `no, yes` Changes Take Effect: When Stat Server is restarted
qinfo-table	Specifies whether Stat Server writes records about queue statuses directly to the `QINFO` table. Refer to page 109 for a complete description of this table. This option was previously named `QInfoTable`. Default Value: `off` Valid Values: `on, off` Changes Take Effect: Immediately upon notification

Table 2: Configuration Options for Operating Stat Server with a Database (Continued)

Option	Description
status-table	Specifies whether Stat Server writes records about agent statuses directly to the `STATUS` table. Refer to page 110 for a complete description of this table. This option was previously named `StatusTable`. Default Value: `off` Valid Values: `on, off` Changes Take Effect: Immediately upon notification
status-table-update-end-time-at-end-only	Setting this option to `on` enables Stat Server to set the `EndTime` and `EndLocalTime` fields of the `STATUS` table to `0` (zero) during updates, provided that the corresponding status has not yet ended. A zero value implies `0` for integer fields and "" (empty string) for character fields. As soon as the statuses complete, Stat Server updates those fields with the time when the statuses ended. Default Value: `no` Valid Values: `yes, no` Changes Take Effect: When Stat Server is restarted
time-format	Specifies the time format of data stored in the `StartLocalTime` and `EndLocalTime` fields in the `STATUS` table. You must set the `local-time-in-status-table` option (see page 45) to `yes` to use the `time-format` option. The format string consists of one or more codes preceded by a percent sign (%). Character strings that do not begin with % are copied unchanged to `strDest`. This option was previously named `TimeFormat`. Default Value: `%m/%d/%Y %H:%M:%S` Valid Values: See Table 3 on page 49 for a complete listing and description of valid time formats. Changes Take Effect: When Stat Server is restarted This option was previously named `TimeFormat`. **Example** Suppose you are using the default time format `%m/%d/%Y %H:%M:%S`. If the start time for a particular state is Tuesday, January 1, 1999, at 3 PM and 10 seconds, character data stored in the `STARTLOCALTIME` field in the `STATUS` table is stored as `01/01/1999 15:00:10`. Changing the format codes for the date in the `time-format` option to `%Y/%m/%d` means the date is stored in the international date format as `1999/01/01`. Spaces can also be used. For example, `%Y %m %d` would be stored as `1999 01 01`.

Table 2: Configuration Options for Operating Stat Server with a Database (Continued)

Option	Description
use-server-id	This option prevents constraint-violation errors from occurring in a database when more than one Stat Server application attempts to write to the same database. If only one Stat Server application writes to the same database table or you have set the value of the `status-table` option (see page 47) to `no`, you do not have to specify a value for this option. The default template does not include this option nor does the Stat Server Wizard set it. To set this option, enter any number from `0` to `63`. Use a different value for each Stat Server application that writes to the same database table. Each Stat Server application uses its assigned value to generate internally stored IDs. **Note:** Configure this option only for those Stat Server applications writing to the same database and monitoring different switches. Do not configure Stat Server applications to write to the same database if they monitor the same switches. This option was previously named `UseServerID`. Default Value: No default value Valid Value: Any integer from `0` (zero) to `63` Changes Take Effect: Immediately upon notification
voice-reasons-table	Specifies whether Stat Server stores the reasons for agents to change or continue `Ready` and `NotReady` states and `AfterCallWork` work mode. If this option is set to `yes`, Stat Server writes the reasons records directly to the `VOICE_REASONS` table. Refer to page 112 for a description of this table. Default Value: `no` Valid Values: `yes`, `no` Changes Take Effect: Immediately upon notification
warn-unsent-sql-statements	Defines the threshold upon which Stat Server begins logging warning messages about the number of unsent SQL statements. To avoid data loss, Stat Server must remain connected to DB Server for the entire period that DB Server submits records to the RDBMS. Your addp timeout for connection from Stat Server to DB Server should be set as large as possible to prevent disconnection by addp. Default Value: `5000` (SQL statements) Valid Values: Any positive value, both less than `2147483648` (2^{31}) and less than the value specified by the `max-unsent-sql-statements (continued)` configuration option. Changes Take Effect: When Stat Server is restarted

Table 3 lists the valid time-format codes you can use with the `time-format` configuration option.

Table 3: Valid Time-Format Codes

Format Code	Description
%a	Abbreviated weekday name
%A	Full weekday name
%b	Abbreviated month name
%B	Full month name
%c	Date and time representation appropriate for locale
%d	Day of month as a two-digit number (`01`–`31`)
%H	Hour in 24-hour format (`00`–`23`)
%I	Hour in 12-hour format (`01`–`12`)
%j	Day of year as a three-digit number (`001`–`366`)
%m	Month as a two-digit number (`01`–`12`)
%M	Minute as a two-digit number (`00`–`59`)
%p	Current locale's AM/PM indicator for 12-hour clock
%S	Second as a two-digit number (`00`–`59`)
%U	Week of year as a two-digit number, with Sunday as the first day of week (`00`–`51`)
%w	Weekday as a one-digit number (`0`–`6`; Sunday is `0`)
%W	Week of year as a two-digit number, with Monday as first day of week (`00`–`51`)
%x	Date representation for current locale
%X	Time representation for current locale
%y	Year without century, as a two-digit number (`00`–`99`)
%Y	Year with century, as a four-digit number (`1970`–*x*)
%z, %Z	Time-zone name or abbreviation; no characters, if time zone is unknown
%%	Percent sign

Table 3: Valid Time-Format Codes (Continued)

Format Code	Description
%#c	Long date and time representation, appropriate for current locale; for example, `Wednesday, March 14, 2001, 12:41:29`
%#x	Long date representation, appropriate to current locale; for example, `Wednesday, March 14, 2001`
#	The pound sign (#) can precede any formatting code. This changes the meaning of the format code as shown in entries with the pound sign in this table. **Notes:** • The pound sign is ignored in these format codes: `%#a, %#A, %#b, %#B, %#p, %#X, %#z, %#Z, %#%` • The pound sign in these format codes removes any leading zeroes: `%#d, %#H, %#I, %#j, %#m, %#M, %#S, %#U, %#w, %#W, %#y, %#Y`

Table 4: Java-Related Options in the [statserver] Section

Option	Description
debug-level	Adding `Java` to the value of this option enables Stat Server to log messages that are related to Java extension functionality. For the complete description of this option, see page 34.
enable-java	When you set the value of this option to `true`, Stat Server tries to load JVM at startup.The `jvm-path` configuration option described on page 53 defines the location of JVM. If you set this value to `false` at Stat Server startup, but later set it to `true`, Stat Server attempts to load JVM at runtime. **Note:** Stat Server ignores the change in setting from `true` to `false`. To unload JVM, you must stop Stat Server. Default Value: `false` Valid Values: `true, false` Changes take effect: When Stat Server is restarted

Java Sections

Note: For this Stat Server release, Java functionality is reserved for use in conjunction with Genesys-provided reports for Outbound Contact, eServices (formerly known as Multimedia), and the Voice Callback option of Enterprise Routing.

Upon startup, Stat Server reads the `enable-java` configuration option to determine whether SSJE (Stat Server Java Extension) functionality is enabled. If the value of this option is `true`, Stat Server processes the information specified in the following Stat Server sections:

- `[java-config]`
- `[jvm-options]`
- `[java-extensions]`

using the following high-level procedure:

1. Stat Server verifies that the `[java-config]` section exists.
2. Stat Server verifies that the `jvm-path` option within that section has been specified.
3. Stat Server verifies that the `[jvm-options]` section exists.
4. If all three are true, Stat Server loads Java Virtual Machine (JVM) from the path specified by `jvm-path` using any options you may have specified within the `[jvm-options]` section.

   For Stat Server to be able to load JVM, environment variables should be set appropriately on the host for your operating system, including:
   - `LD_LIBRARY_PATH` for all UNIX platforms (except AIX).
   - `LIBPATH` for AIX.
   - `PATH` for Microsoft Windows.

   The path to `<JDK HOME>/jre/bin/` should be included within the value of the preceding environment variables.
5. Stat Server loads Java classes from the Genesys Platform SDK (`kv65_adapter.jar` and `kvlists.jar`) and from the Stat Server Java SDK (`statserver.jar` and `statserver_impl.jar`).
6. Stat Server loads the Java libraries indicated by the `java-libraries-dir` configuration option of the `[java-config]` section.
7. If Stat Server successfully loads the Java host environment, Stat Server next tries to load Java Extensions (specified by the `java-extensions-dir` configuration option of the `[java-config]` section) from archives specified in the `[java-extensions]` section.

8. Stat Server takes the initial parameters for each `<extension.jar>` extension from the section where `java-extension-jar=<extension.jar>`, and uses them for this extension execution.

Each Java configuration section is further described in Table 5. For those configuration options for which you specify `true/false` values, any of the following additional values are also valid:

- `yes/no`
- `1/0`
- `y/n`
- `on/off`

These alternates may not be indicated in the table.

Table 5: Java Sections and Pertinent Configuration Options

Option	Description
java-config Section	
java-extensions-dir	The value of this option must contain the path to the directory where all Java Extensions are stored. Default Value: `./java/ext` Valid Value: Any valid, fully specified directory path Changes Take Effect: Upon Stat Server restart, or upon setting the `enable-java` configuration option to `true`.
java-libraries-dir	The value of this option must contain the path to the directory where all Java libraries are stored. Default Value: `./java/lib` Valid Value: Any valid, fully specified directory path Changes Take Effect: Upon Stat Server restart, or upon setting the `enable-java` configuration option to `true`.
java-extension-loading-timeout	Specifies the length of time, in seconds, that Stat Server allocates for loading Java Extensions. If an Extension does not load within this timeout, Stat Server sends a message to its log indicating this. Stat Server makes no further attempts to load the Extension during runtime. Default Value: `20` Valid Values: Any positive integer less than `2147483648` (2^{31}) Changes Take Effect: Upon Stat Server restart, or upon setting the `enable-java` configuration option to `true`. Only under rare circumstances should you change this option, such as if your particular Java Extension is very large or if its execution is very time consuming.

Table 5: Java Sections and Pertinent Configuration Options (Continued)

Option	Description
jvm-path	Contains the path to Java Virtual Machine: Stat Server supports JDK version 1.4 or 1.5. • `jvm.dll` on Windows • `libjava.so`, `libjvm.so`, `libjvm.a`, or `libjvm.sl` on Unix Default Value: "" (Empty string) Valid Value: Any valid, fully specified path (including file name) to the particular file Changes Take Effect: Upon Stat Server restart, or upon setting the `enable-java` configuration option to `true`.
jvm-options Section The configuration options you specify for this section correspond to the Java executable (`java.exe` on Windows, `java` on UNIX), and command-line options specific to your branch and version of JVM. Refer to your JVM documentation to find out its applicable configuration options. For Solaris platforms, set the stack space to at least 4,096K. For example, for HotSpot JVM, configuring the following would accomplish this: `-XX:ThreadStackSize=4096` Note that Genesys neither recommends nor endorses any particular JVM. Configuration options follow the `Name/Value` format used in other Stat Server sections, where `Name` is the name of the Java command-line option. If you specify a value for a named configuration option in this section, Stat Server converts the two to `Name=Value` before passing the option to JVM. If you do not specify a value, Stat Server passes the name only.	

Example 1	Assume that `foo` is a valid option requiring a value for your Java application. To specify a value of `some string`, create the following configuration option within the `[jvm-options]` section of your Stat Server application.
	`Name = -Dfoo`
	`Value = "some string"`
	Note: Include quotes in the value's definition if JVM requires them on the command line.

Example 2	This example demonstrates how to configure an option—the Java HotSpot Client VM—which does not require a value.
	`Name = -Client`
	Note: You must include the hyphen if JVM requires it.
	`Value = <null>`

Table 5: Java Sections and Pertinent Configuration Options (Continued)

Option	Description
java-extensions Section Default behavior creates neither this section nor any of its configuration options. You must manually add the section to the Stat Server `Application` object and provide an arbitrary section name. Use this section to pass initialization parameters to the Java Extension.	
<filename>.jar	The name of this Java configuration option is the relative path of the Java Extension jar archive with respect to the SSJE installation directory described with [`java-config`]/`java-extensions-dir`. The resulting combined path should point inside the SSJE installation directory (note that on UNIX systems, all symbolic links are resolved). Otherwise, Stat Server logs a security violation message and does not load the corresponding SSJE. Furthermore, if Stat Server cannot match the resulting path to any existing Java Extension configured to be loaded, Stat Server ignores the content of this entire section. The corresponding value is either `false` (indicating that Stat Server is not to consider this particular Java Extension) or `true` (indicating that it is). The path is relative to that specified by the `java-extensions-dir` configuration option described on page 52; for example, `ext1.jar` or `subdir3/ext3.jar`. If you initially do not set this option when Stat Server first starts, but later set it, Stat Server attempts to dynamically load the extension at runtime. Refer to How to Configure a Particular Java Extension below for additional information. Default Value: No default value Valid Values: `false, true` Changes Take Effect: during Java Extension initialization phase
<filename>.jar (continued)	If, during the configuration of your Stat Server `Application` object using the Stat Server Wizard, you select the `eServiceContactStat` check box, the wizard adds the `eServiceContactStat.jar` option to this section with a `yes` value. If you clear the check box, the wizard adds the option with a `no` value. Likewise, the Stat Server Wizard adds the `eServiceInteractionStat.jar` option to this section with either a `yes` or `no` value, depending on whether you select or clear the `eServiceInteractionStat` check box.

Table 5: Java Sections and Pertinent Configuration Options (Continued)

Option	Description
<Name>	<Value> You can specify additional configuration options following the `Name/Value` format used in other Stat Server sections, where `Name` is name of the parameter to be passed to SSJE and `Value` is the parameter's value. If you do specify a value for a parameter in this section, Stat Server converts the `Name/Value` pair to `Name=Value` before passing it to SSJE. If you do not specify a value, Stat Server passes only the name. Default Value: No default value Changes Take Effect: during Java Extension initialization phase

How to Configure a Particular Java Extension

When Stat Server loads SSJE, Stat Server passes a set of parameters during the initialization phase. To specify those parameters in Stat Server, follow these steps:

1. Create a new configuration section, with an arbitrary name, on the Stat Server `Options` tab in Configuration Server.
2. Within this section, create the `java-extension-jar` option and, as its value, specify the relative path of the corresponding SSJE jar archive with respect to the SSJE installation directory; for example, `MySSJE.jar`.
3. Add any other options to this section. Stat Server passes the corresponding `name:value` pairs to SSJE during the initialization phase.

How to Configure a Particular Java Extension Stat Type

Some Stat Server clients (such as CC Analyzer) require an explicit statistical type (stat type) configuration in Configuration Server. Java stat types are configured slightly differently than regular stat types. To configure a particular stat type defined in a Java Extension:

1. Create a new section, with an arbitrary name, on the Stat Server `Options` tab in Configuration Server.
2. Within the newly created section, create these new mandatory options:
   - `Category`
   - `Objects`
   - `JavaSubCategory`

   The first two are standard for all stat types. (Refer to the "Statistical Type Sections" section in the *Framework 8.0 Stat Server User's Guide* for a description of these and other options.)

The value of the third option must have the format *extension-jar-path:stat-type-name,* where:

- *extension-jar-path* is the relative path of the Java Extension jar archive with respect to the SSJE installation directory described by [jvm-options]/java-extensions-dir.
- *stat-type-name* is the name of the stat type residing in SSJE.

3. Add any other options to the newly created section. Stat Server will pass the corresponding name:value pairs to SSJE whenever the statistics associated with this corresponding stat type are requested.

Chapter

6 Other Factors Affecting Stat Server

Stat Server receives events from the Genesys applications that are configured in Stat Server's application connections and processes them within the confines of Stat Server's configuration. In addition, Stat Server directly reads general information about the switch underlying these applications. Stat Server uses this information, in part, to determine which action(s) to generate and report to its clients. Though Stat Server does not read the values of the configuration options of such applications, Stat Server does consider certain attributes about these applications (such as their type and version) in its handling of events that originate from these applications.

Note: For the purpose of this chapter, DNs and switches are not considered to be applications. However, Stat Server does read the configuration options of these objects to provide certain functionality.

This chapter describes factors other than Stat Server's own configuration that have an impact on Stat Server output. This chapter contains the following sections:

For information about manipulating Stat Server behavior via the configuration of the Stat Server `Application` object, refer to the previous chapter "Fine-Tuning Stat Server Configuration" on page 29.

Stat Server Reads Switch and DN Attributes

To Determine Capacity and Impact Routing of Interactions to Multimedia DNs

In support of reporting for multimedia DNs, whenever Stat Server detects a multimedia DN, Stat Server now reads the DN's attributes, and those of its switch, to determine whether the DN is capable of handling multiple, simultaneous interactions of differing media types. Stat Server looks for the following:

- A DN switch type of either of the following:
  - `VoIPSMCPSwitch` (`Voice over IP SMCP Switch` in Configuration Manager).
  - `SIPSwitch` (`SIP Switch` in Configuration Manager)
- A DN type of `CFGExtension` (`Extension` in Configuration Manager)
- A value of `yes` in the `[TServer]/multimedia` configuration option for the DN. (This option is defined on the `Annex` tab of the DN object in Configuration Manager.)
- Version 7.6.*x* or greater of T-Server, if the switch type is `SIPSwitch`.

Stat Server uses the switch's media attributes as the default for all `Extension` DNs that belong to it.

If these criteria are met, Stat Server supports routing of interactions with `chat` or `voice` media types to multimedia DNs. (For more information on this subject, refer to "Capacity Planning for Multimedia DNs" in the *Genesys Resource Capacity Planning Guide*.) Prior to release 7.6, Stat Server supported routing of `voice` interactions only to such DN types.

To Suppress the Transmission of Attached Data

For the switches and DNs that Stat Server monitors, Stat Server checks the `[statserver]` section of the `Annex` tab for the value of the `suppress-user-data` configuration option. The value of this option determines whether Stat Server should transmit call-extracted attached data to Stat Server clients for the particular DN on which the option was set or for all DNs registered on a switch. Setting this option is useful for reducing network traffic in environments where many Stat Server applications are connected to a single T-Server, for example, and where each Stat Server application in such a scenario serves a different business purpose.

A value of `no` (the default value) indicates that Stat Server will continue to receive attached data (and transmit attached data to its clients). If the option value is set to `yes`, however, T-Server will not send any `EventAttachedData` TEvents or `AttributeUserData` attributes of any other TEvent to Stat Server;

and, as a result, Stat Server will not transmit userdata, for the associated DN or switch object, to its clients.

If this option is defined for a particular DN, its value overrides any value that may be specified at the switch. Dynamic changes to this option take effect upon DN re-registration.

Note: The selective suppression of attached data is possible only with T-Server release 7.6 and later.

For Processing Stuck Calls and ACW Notifications

In addition, Stat Server regularly references a switch's type and a DN's type to perform many other operations, such as checking for stuck calls or processing ACW notifications.

Stat Server Reads Resource Attributes

To Determine Which Objects Are Enabled

To calculate group- and queue-related statistics, Stat Server considers whether member `Person` and `Place` objects have been enabled in Configuration Server, depending on the values of the `ignore-disabled-objects-in-group-statistics` and `ignore-disabled-objects-in-queue-statistics` Stat Server configuration options (described on page 37). This property of contact center resources is but one attribute that Stat Server directly reads about configuration objects.

To Determine if Origination DNs Are Configured

Stat Server, also reads the properties of agent group and place group objects to determine if origination DNs have been configured therein (on the `Advanced` tab of the object's properties in Configuration Manager). If configured, Stat Server reflects the events occurring at these origination DNs for agent group and place group statistics computations—Stat Server also generates retrospective, interaction-related actions reflecting regular DNs onto these origination DNs.

Stat Server Reads Virtual Agent Group Definitions

To Determine Group Membership

For agent group objects, Stat Server also reads the `script` configuration option (located in the `virtual` section of the `Annex` tab) to determine the objects to which actions apply. Refer to the "Virtual Agent Groups" chapter of the *Framework 8.0 Stat Server User's Guide* for more information about how to define this object.

Stat Server Reads Mediation DN Attributes

To Determine Average Handling Time

When calculating statistics for URS so that it can balance call loads over several mediation DNs, Stat Server reviews each mediation DN's setting of average handling time, which is configured through use of the `load-balance-aht` option in the `[statserver]` section on the `Annex` tab of mediation DN objects. Values specified at the mediation DN–level supersede the global value, which is controlled and set within the Stat Server `Application` object, and the same range of values apply. See the description of the `load-balance-aht` Stat Server configuration option on page 38.

Dynamic changes to this option, at the mediation DN level, take effect immediately upon notification of mediation DN re-registration.

Chapter

7 Common Log Options

This chapter describes log configuration options that are common to all Genesys server applications and applicable to any Framework server component. This chapter includes the following sections:

Mandatory Options

You do not have to configure any common log options to start Server applications.

Log Section

You must name this section `log`. Table 6 lists the log configuration options available to you. Note that to use these options, you must actively set them, either using the Stat Server Wizard or manually on the `Options` tab of the Stat Server `Application` object within Configuration Manager. The default Stat Server application template includes only the `verbose` option.

Table 6: Log Options

Option	Description
verbose	Determines whether a log output is created. If it is, this option specifies the minimum level of log events generated. The log events levels, starting with the highest-priority level, are `standard`, `interaction`, `trace`, and `debug`. Refer to "Log Output Options" on page 70 for more information. Default Value: `all` Valid Values: `all` — All log events (that is, log events of `standard`, `trace`, `interaction`, `alarm`, and `debug` levels) are generated if you set the `debug-level` option in the `statserver` section to `all`. `debug` — The same as `all`. `trace` — Log events of `trace` and higher levels (that is, log events of `standard` and `interaction` levels) are generated, while log events of the `debug` level are not. `interaction` — Log events of the `interaction` and higher levels (that is, log events of `standard` level) are generated, while log events of the `trace` and `debug` levels are not generated. `standard` — Log events of the `standard` level are generated, while log events of the `interaction`, `trace`, and `debug` levels are not generated. `none` — Produces no output. Changes Take Effect: Immediately Refer to the *Framework 8.0 Management Layer User's Guide* or to *Framework 8.0 Solution Control Interface Help* for more information on the `standard`, `trace`, `interaction`, and `debug` log levels.
buffering	Turns operating system file buffering on or off. This option applies only to `stderr` and `stdout` output (see page 70). Setting this option to `true` increases output performance. **Note:** When you enable buffering, messages may appear at the console with delay. Default Value: `true` Valid Values: `true` — Enables buffering `false` — Disables buffering. Changes Take Effect: Immediately

Table 6: Log Options (Continued)

Option	Description
segment	Specifies whether there is a segmentation limit for a log file. If there is, this option sets the mode of measurement along with the maximum size. If the current log segment exceeds the size set by this option, the current file is closed and a new file is created. Stat Server ignores this option if log output is not configured to be sent to a log. Default Value: `false` Valid Values: `false` — No segmentation allowed. `<number> KB` or `<number>` — Sets the maximum segment size in kilobytes. The minimum segment size is 100 KB. `<number> MB` — Sets the maximum segment size in megabytes. `<number> hr` — Sets the number of hours for the segment to stay open. The minimum number is 1 hour. Changes Take Effect: Immediately
keep-startup-file	Specifies whether a startup segment of the log, containing the initial T-Server configuration, is to be kept. If it is, you can set this option to `true` or to a specific file size. A `true` setting means that the size of the initial segment will be equal to the size of the regular log segment defined by the `segment` option (defined on page 63). Stat Server ignores this option if you set the `segment` option to `false`. Default Value: `false` Valid Values: `false` — No startup segment of the log is kept. `true` — A startup segment of the log is kept. The size of the segment equals the value of the `segment` option. `<number> KB` — Sets the maximum size, in kilobytes, for a startup segment of the log. `<number> MB` — Sets the maximum size, in megabytes, for a startup segment of the log. Changes Take Effect: After restart

Table 6: Log Options (Continued)

Option	Description
expire	Determines whether log files expire. If they do, this option sets the measurement for determining when they expire, along with the maximum number of files (segments) or days before the files are removed. Default Value: `false` Valid Values: `false` — No expiration; all generated segments are stored. `<number>` `file` or `<number>` — Sets maximum number of log files to store. Specify a number from `1-1000`. `<number> day` — Sets the maximum number of days before log files are deleted. Specify a number from `1-100`. Changes Take Effect: Immediately **Note:** If you incorrectly set an option's value—such as setting values outside the range of valid values—Stat Server automatically resets the value to `10`.
messagefile	Specifies the file name for Stat Server log events. The name must be valid for the operating system on which Stat Server is running. The option value can also contain the absolute path to the `statserver.lms` file. Otherwise, Stat Server looks for the file in its working directory. Default Value: `statserver.lms` Valid Values: `<string>.lms` Changes Take Effect: After Stat Server restarts if Stat Server locates `statserv.lms` at startup or immediately if Stat Server cannot locate this file at startup **Warning!** If Stat Server cannot locate its message file upon startup, Stat Server will not be able to generate log events or send them to Message Server.
message_format	Specifies the format of log record headers that Stat Server uses when writing to the log file. Using compressed log record headers improves Stat Server performance and reduces the log's file size. Default Value: `short` Valid Values: `short` — Stat Server uses compressed headers when writing log records to its log file. `full` — Stat Server uses complete headers when writing log records to its log file.

Table 6: Log Options (Continued)

Option	Description

message_format (continued)	Changes Take Effect: Immediately
	With the value set to `short`:
	• A header of the log file or the log file segment contains information about the application (such as the application name, application type, host type, and time zone) while single log records within the file or segment omit this information.
	• A log message priority is abbreviated to `std`, `Int`, `Trc`, or `Dbg`, for `standard`, `interaction`, `trace`, or `debug` messages respectively.
	• `Message ID` does not contain the prefix GCTI or the application type ID.
	A log record in the `short` format looks like this:
	`2007-05-07T18:15:33.952 Std 05060 Application started`
	A log record in the `full` format looks like this:
	`2007-05-07T18:11:38.196 Standard localhost cfg_dbserver GCTI-00-05060 Application started`
	Note: Whether the `full` or `short` format is used, time is printed as specified by the `time_format` option.

time_convert	Specifies the system by which Stat Server calculates the log record time when generating a log file. The time is converted from the time in seconds since the Epoch (00:00:00 UTC, January 1, 1970). Default Value: `Local` Valid Values: `local` — Time of log-record generation expressed as a local time, based on the time zone and any seasonal adjustments. Time zone information of the Stat Server host computer is used. `utc` — Time of log-record generation expressed as Coordinated Universal Time (UTC). Changes Take Effect: Immediately

Table 6: Log Options (Continued)

Option	Description
time_format	Specifies how to represent the time in a log file when Stat Server generates log records. Default Value: `time` Valid Values: `time` — Time string is formatted according to the HH:MM:SS.sss (hours, minutes, seconds, and milliseconds) format. `locale` — Time string is formatted according to the system's locale. `ISO8601` — Date in the time string is formatted according to ISO 8601 format. Fractional seconds are given in milliseconds. Changes Take Effect: Immediately A log record's time field in ISO 8601 format looks like this: `2007-07-24T04:58:10.123`
print-attributes	This log option has no effect on Stat Server. Default Value: `false` Valid Values: `true, false` Changes Take Effect: Immediately
check-point	Specifies how often, in hours, Stat Server generates a `check-point` log event to divide the log into sections of equal time. By default, Stat Server generates this log event every hour. Setting the option to `0` prevents generation of `check-point` events. Default Value: `1` Valid Values: `0-24` Changes Take Effect: Immediately
memory	Specifies the name of the file to which Stat Server regularly prints a snapshot of the memory output (see page 70). The new snapshot overwrites previously written data. If Stat Server terminates abnormally, this file contains the latest log messages. Memory output is not recommended for processors with a CPU frequency lower than 600 MHz. **Note:** If the file specified is located on a network drive, Stat Server does not create a snapshot file (with the extension `*.memory.log`). Default Value: No default value Valid Values: `<string>` (memory file name) Changes Take Effect: Immediately

Table 6: Log Options (Continued)

Option	Description
memory-storage-size	Specifies the buffer size for log output to the memory. Refer also to "Log Output Options" on page 70 for more information. Default Value: `2 MB` Valid Values: `<number> KB` or `<number>` — Size of the memory output, in kilobytes. The minimum value is `128 KB`. `<number> MB` — Size of the memory output, in megabytes. The maximum value is `64 MB`. Changes Take Effect: When memory output is created
spool	Specifies the folder, including full path to it, in which Stat Server creates temporary log-related files. If you change this value while Stat Server is running, the change does not affect the currently open network output. Default Value: The Stat Server working directory Valid Values: `<path>` — The full path of the folder Changes Take Effect: Immediately
compatible-output-priority	Specifies whether Stat Server uses 6.*x* output logic. Default Value: `false` Valid Values: `true` — The log of the level specified by one of the log output options described on page 71 is sent to the specified output. `false` — The log of the level specified by one of the log output options described on page 71 and higher levels is sent to the specified output.

Table 6: Log Options (Continued)

Option	Description

compatible-output-priority (continued)	Changes Take Effect: Immediately
	For example, you configure the following options in the `log` section for a 6.*x* application and for a 8.*x* application:
	`[log]`
	`verbose = all`
	`debug = file1`
	`standard = file2`
	Stat Server 6.*x* log file content is as follows:
	• `file1` contains `debug` messages only.
	• `file2` contains `standard` messages only.
	Stat Server 8.*x* log file content is as follows:
	• `file1` contains `debug`, `trace`, `interaction`, and `standard` messages.
	• `file2` contains `standard` messages only.
	Warning! Genesys does not recommend changing the default value of the `compatible-output-priority` option unless you have specific reasons to use the 6.*x* log output logic—that is, to mimic the output priority as implemented in releases 6.*x*. Setting this option to `true` affects log consistency.

The configuration options listed in Table 7 enable you to generate debug logs containing information about specific Stat Server operations. You designate these options in the `log` section of the Stat Server application.

`Warning!` Genesys advises you to used these options only when requested by Genesys Technical Support.

Table 7: Debug Log Options

Option	Description
x-conn-debug-open	Generates debug log records about “open connection” Stat Server operations. Default Value: `0` Valid Values: `0` Log records are not generated. `1` Log records are generated. Changes Take Effect: After restart

Table 7: Debug Log Options (Continued)

Option	Description
x-conn-debug-select	Generates debug log records about "socket select" Stat Server operations. Default Value: `0` Valid Values: `0` Log records are not generated. `1` Log records are generated. Changes Take Effect: After restart
x-conn-debug-timers	Generates debug log records about Stat Server operations involved in creating and deleting system timers. Default Value: `0` Valid Values: `0` Log records are not generated. `1` Log records are generated. Changes Take Effect: After restart
x-conn-debug-write	Generates debug log records about Stat Server write operations. Default Value: `0` Valid Values: `0` Log records are not generated. `1` Log records are generated. Changes Take Effect: After restart
x-conn-debug-security	Generates debug log records about security-related operations, such as Transport Layer Security (TLS) and security certificates. **Note:** This option has no effect on Stat Server 7.6 and earlier releases, which do not support TLS operations. Default Value: `0` Valid Values: `0` Log records are not generated. `1` Log records are generated. Changes Take Effect: After restart

Table 7: Debug Log Options (Continued)

Option	Description
x-conn-debug-api	Generates debug log records about connection library function calls. Default Value: `0` Valid Values: `0` Log records are not generated. `1` Log records are generated. Changes Take Effect: After restart
x-conn-debug-dns	Generates debug log records about DNS operations. Default Value: `0` Valid Values: `0` Log records are not generated. `1` Log records are generated. Changes Take Effect: After restart
x-conn-debug-all	Generates debug log records about open connection, socket select, timer creation and deletion, write, security-related, DNS operation, and connection library function calls. This option is the same as enabling or disabling all of the previous `x-conn-debug-<optype>` options. Default Value: `0` Valid Values: `0` Log records are not generated. `1` Log records are generated. Changes Take Effect: After restart

Log Output Options

To configure log outputs, set log level options (`all`, `alarm`, `standard`, `interaction`, `trace`, `memory`, and/or `debug`) to the desired types of log output (`stdout`, `stderr`, `network`, `memory`, and/or `[filename]` for log file output).

You can use:

- One log level option to specify different log outputs.
- One log output type for different log levels.
- Several log output types simultaneously for logging the events of the same or different log levels.

You must separate the log output types by a comma when you are configuring more than one output for the same log level. See "Examples" on page 75.

Note: The log output options are activated according to the setting of the `verbose` configuration option.

Warnings!
- If you direct log output to a file on the network drive, an application does not create a snapshot log file (with the extension `*.snapshot.log`) in case it terminates abnormally.
- Directing log output to the console (by using the `stdout` or `stderr` settings) can affect application performance. Avoid using these log output settings in a production environment.

Table 8: Log Output Options

Option	Description
all	Specifies the outputs to which Stat Server sends all log events. You must separate log output types with commas when you configure more than one output type. Default Value: `stdout` Valid Values: `stdout` — Log events are sent to the standard output (`stdout`). `stderr` — Log events are sent to the standard error output (`stderr`). `network` — Log events are sent to Message Server, which can reside anywhere on the network. Message Server stores log events in the Log Database. Setting the `all` log level option to `network` enables Stat Server to send log events of `standard`, `interaction`, and `trace` levels to Message Server. Log events of `debug` level are neither sent to Message Server nor stored in the Log Database. `memory` — Log events are sent to the memory output on the local disk. This output is the safest in terms of Stat Server performance. `[filename]` — Log events are stored in a file with the specified name. If you do not specify a path, the file is created in Stat Server's working directory. Changes Take Effect: Immediately For example, `all = stdout, logfile` **Note:** To ease the troubleshooting process, consider using unique names for log files that different Stat Server applications generate.

Table 8: Log Output Options (Continued)

Option	Description
alarm	Specifies the outputs to which Stat Server sends log events of `Alarm` level. You must separate log output types with commas when you configure more than one output type. For example, `alarm = stdout, logfile` Default Value: No default value Valid Values (log output types): `stdout` — Log events are sent to the standard output (`stdout`). `stderr` — Log events are sent to the standard error output (`stderr`). `network` — Log events are sent to Message Server, which can reside anywhere on the network. Message Server stores log events in the Log Database. Setting the `all` log level option to `network` enables Stat Server to send log events of `Standard`, `Interaction`, and `Trace` levels to Message Server. Log events of `debug` level are neither sent to Message Server nor stored in the Log Database. `memory` — Log events are sent to the memory output on the local disk. This output is the safest in terms of Stat Server performance. `[filename]` — Log events are stored in a file with the specified name. If you do not specify a path, the file is created in Stat Server's working directory. Changes Take Effect: Immediately

Table 8: Log Output Options (Continued)

Option	Description
standard	Specifies the outputs to which Stat Server sends log events of the `Standard` level. You must separate log output types with commas when you configure more than one output type. For example, `standard = stderr, network` Default Value: No default value Valid Values (log output types): `stdout` Log events are sent to the standard output (`stdout`). `stderr` Log events are sent to the standard error output (`stderr`). `network` Log events are sent to Message Server, which can reside anywhere on the network. Message Server stores the log events in the Log Database. `memory` Log events are sent to the memory output on the local disk. This is the safest output in terms of Stat Server performance. `[filename]` Log events are stored in a file with the specified name. If you do not specify a path, the file is created in Stat Server's working directory. Changes Take Effect: Immediately
interaction	Specifies the outputs to which Stat Server sends log events of the `Interaction` and higher levels (that is, log events of `Standard` and `Interaction` levels). You must separate log outputs with commas when you configure more than one output type. For example, `interaction = stderr, network` Default Value: No default value Valid Values (log output types): `stdout` Log events are sent to the standard output (`stdout`). `stderr` Log events are sent to the standard error output (`stderr`). `network` Log events are sent to Message Server, which can reside anywhere on the network. Message Server stores the log events in the Log Database. `memory` Log events are sent to the memory output on the local disk. This is the safest output in terms of Stat Server performance. `[filename]` Log events are stored in a file with the specified name. If you do not specify a path, the file is created in Stat Server's working directory. Changes Take Effect: Immediately

Table 8: Log Output Options (Continued)

Option	Description
trace	Specifies the outputs to which Stat Server sends log events of `Trace` and higher levels (that is, log events of `Standard`, `Interaction`, and `Trace` levels). You must separate log outputs with commas when you configure more than one output type. For example, `trace = stderr, network` Default Value: No default value Valid Values (log output types): `stdout` Log events are sent to the standard output (`stdout`). `stderr` Log events are sent to the standard error output (`stderr`). `network` Log events are sent to Message Server, which can reside anywhere on the network. Message Server stores the log events in the Log Database. `memory` Log events are sent to the memory output on the local disk. This output is the safest in terms of Stat Server performance. `[filename]` Log events are stored in a file with the specified name. If you do not specify a path, the file is created in Stat Server's working directory. Changes Take Effect: Immediately
debug	Specifies the outputs to which Stat Server sends log events of `debug` and higher levels (that is, log events of `standard`, `trace`, `interaction`, and `debug` levels). You must separate log output types with commas when you configure more than one output type. For example, `debug = stderr, /usr/local/genesys/logfile` Default Value: No default value Valid Values (log output types): `stdout` Log events are sent to the standard output (`stdout`). `stderr` Log events are sent to the standard error output (`stderr`). `memory` Log events are sent to the memory output on the local disk. This output is the safest in terms of Stat Server performance. `[filename]` Log events are stored in a file with the specified name. If you do not specify a path, the file is created in Stat Server's working directory. Changes Take Effect: Immediately **Note:** Log events of `debug` level are never sent to Message Server nor stored in the Log Database.

Log File Extensions

You can use the following file extensions to identify log files that Stat Server creates for various types of output:

- `*.log`—Assigned to log files when you configure output to a log file. For example, if you set `standard = statservlog`, Stat Server prints log messages into a text file called `statservlog.<time_stamp>.log`.
- `*.qsp`—Assigned to temporary (spool) files when you configure output to the network, but the network is temporarily unavailable. For example, if you set `standard = network`, Stat Server prints log messages into a file called `statserv.<time_stamp>.qsp` during the time the network is unavailable.
- `*.snapshot.log`—Assigned to files containing the output snapshot when you configure output to a log file. The file contains the last log messages that Stat Server generates before abnormal termination. For example, if you set `standard = statservlog`, Stat Server prints the last log message into a file called `statserv.<time_stamp>.snapshot.log` in case of failure.

  Note: Provide `*.snapshot.log` files to Genesys Technical Support when reporting a problem.

- `*.memory.log`—Assigned to log files that contain the memory output snapshot when you configure output to memory and redirect the most recent memory output to a file. For example, if you set `standard = memory` and `memory = statserv`, Stat Server prints the latest memory output to a file called `statserv.<time_stamp>.memory.log`.

Examples

This section presents examples of a `log` section you might configure for an application operating in production mode and in two lab modes, debugging and troubleshooting.

Production Mode Log Section

```
[log]
verbose = standard
standard = network, statservlogfile
```

With this configuration, Stat Server generates only log events of the `Standard` level and sends them to the standard output, to Message Server, and to a file named `statservlogfile`, which Stat Server creates in its working directory. Genesys recommends that you use this or a similar configuration in a production environment.

Warning! Directing log output to the console (by using the `stdout` or `stderr` settings) can affect application performance. Avoid using these log output settings in a production environment.

Lab Mode Log Section

```
[log]
verbose = all
all = stdout, /usr/local/genesys/statservlogfile
trace = network
```

With this configuration, Stat Server generates log events of the `standard`, `interaction`, `trace`, and `debug` levels, and sends them to the standard output and to a file named `statservlogfile`, which Stat Server creates in the `/usr/local/genesys/` directory. In addition, Stat Server sends log events of the `standard`, `interaction`, and `trace` levels to Message Server. Use this configuration to test new interaction scenarios in a laboratory environment. Be sure to appropriately set the `debug-level` option in the `statserver` section.

Failure-Troubleshooting Log Section

```
[log]
verbose = all
standard = network
all = memory
memory = statservlogfile
memory-storage-size = 32 MB
```

With this configuration, Stat Server generates log events of the `standard` level and sends them to Message Server. It also generates log events of the `standard`, `interaction`, `trace`, and `debug` levels, and sends them to the memory output. The most current log is stored to a file named `statservlogfile`, which the application creates in its working directory. An increased memory storage enables Stat Server to save more log information generated before a failure. Use this configuration when trying to reproduce an application failure. The memory log file would contain the snapshot of Stat Server's log at the moment of failure. This should help you and Genesys Technical Support identify the reason for the failure. Be sure to appropriately set the `debug-level` option in the `statserver` section.

Note: If you are operating Stat Server on Unix and do not specify any files in which to store the memory output snapshot, the core file that Stat Server produces before terminating contains the most current Stat Server log. Provide the Stat Server's core file to Genesys Technical Support when reporting problems.

Log-Extended Section

This section must be named `log-extended`.

Table 9: Extended Log Options

Option	Description
level-reassign-<eventID>	Specifies one of five log levels for log event <*eventID*>, which may differ from its default level, or disables logging of the named event altogether. This option is useful if you want to change the behavior of what Stat Server logs for the specified log event ID. If no value is specified, then the named log event retains its default level. You can deactivate these options with the `level-reassign-disable` configuration option, described below. Default Value: Default value of log event <*eventID*>. Refer to the *Common Log Events Help* or `statserver.lms` (located in the directory where Stat Server is installed) for a listing of each of Stat Server's the default levels. Valid Values: `alarm` The log level of log event <*eventID*> is set to `alarm`. `standard` The log level of log event <*eventID*> is set to `standard`. `interaction` The log level of log event <*eventID*> is set to `interaction`. trace The log level of log event <*eventID*> is set to `trace`. debug The log level of log event <*eventID*> is set to `debug`. none Log event <*eventID*> is not recorded in a log. Changes Take Effect: Immediately
level-reassign-disable	When this option is set to `true`, the original (default) log level of all log events in the `[log-extended]` section are restored. This option is useful when you want to use the default levels and keep the customizations. Default Value: `false` Valid Values: `true, false` Changes Take Effect: Immediately Defined: `Options` tab of `Application` object

Warning! Use caution when making these changes in a production environment.

Depending on the log configuration, changing the log level to a higher priority may cause the log event to be logged more often or to a greater number of outputs. This could affect system performance.

Likewise, changing the log level to a lower priority may cause the log event to be not logged at all, or not logged to specific outputs, thereby losing important information. The same applies to any alarms associated with that log event.

In addition to the precautionary message above, take note of the following:

- Logs can be customized only by release 7.6 or later applications.
- When the log level of a log event is changed to any level except none, it is subject to the other settings in the `[log]` section at its new level. If set to none, it is not logged and therefore not subject to any log configuration.
- Changing the log level of a log using this feature changes only its priority; it does not change how that log is treated by the system. For example, increasing the priority of a log to `Alarm` level does not mean that an alarm will be associated with it.
- Each application in an HA pair can define its own unique set of log customizations, but the two sets are not synchronized with each other. This can result in different log behavior depending on which application is currently in primary mode.
- This feature is not the same as a similar feature in Universal Routing Server, version 7.2 or later. In this Framework feature, the priority of log events are customized. In the URS feature, the priority of debug messages only are customized. Refer to the *Universal Routing Server 8.0 Reference Manual* for more information about the URS feature.
- You cannot customize any log event that is not in the unified log record format. Log events of the `Alarm`, `Standard`, `Interaction`, and `Trace` levels feature the same unified log record format.

Example

This is an example of using customized log level settings, subject to the following log configuration:

```
[log]
verbose=interaction
all=stderr
interaction=log_file
standard=network
```

Before the log levels of the log are changed:

- Log event 1020, with default level standard, is output to stderr and the log file, and sent to Message Server.
- Log event 2020, with default level standard, is output to stderr and the log file, and sent to Message Server.
- Log event 3020, with default level trace, is output to stderr.
- Log event 4020, with default level debug, is output to stderr.

Extended log configuration section:

```
[log-extended]
level-reassign-1020=none
level-reassign-2020=interaction
level-reassign-3020=interaction
level-reassign-4020=standard
```

After the log levels are changed:

- Log event 1020 is disabled and not logged.
- Log event 2020 is output to stderr and the log file.
- Log event 3020 is output to stderr and the log file.
- Log event 4020 is output to stderr and the log file, and sent to Message Server.

Log-Filter Section

This section must be called log-filter. Table 10 describes the option you configure in this section.

Table 10: Log-Filter Option

Option	Description
default-filter-type	Specifies the default manner in which KVList information (including UserData, Extensions, and Reasons) is presented in the Stat Server log. Stat Server applies this option to all KVList pairs, except the ones explicitly defined within the log-filter-data section (see page 80). Default Value: copy Valid Values: copy — The keys and values of the KVList pairs are copied to the log. hide — The keys of the KVList pairs are copied to the log; the values are replaced with strings of asterisks. skip — The KVList pairs are not copied to the log. Changes Take Effect: Immediately

Example

```
[log-filter]
default-filter-type=copy
```

Here is an example of a log using the default log filter settings:

```
message RequestSetCallInfo
    AttributeConsultType            3
    AttributeOriginalConnID         008b012ece62c8be
    AttributeUpdateRevision         2752651
    AttributeUserData               [111] 00 27 01 00..
           'DNIS'                   '8410'
           'PASSWORD'               '111111111'
           'RECORD_ID'              '8313427'
    AttributeConnID                 008b012ece62c922
```

Log-Filter-Data Section

This section must be called `log-filter-data`. Table 11 describes the options you configure in this section.

Table 11: Log-Filter-Data Options

Option	Description
<key name>	Specifies the manner in which the KVList pair, defined by the key name (this option's name), is presented in the Stat Server log. Setting this option supersedes the manner in which KVList information is presented as defined by the `default-filter-type` option in the `log-filter` section for the given KVList pair. Default Value: `copy` Valid Values: `copy` The key and value of the given KVList pair are copied to the log. `hide` The given KVList key is copied to the log; the KVList value is replaced with a string of asterisks. (See the example below.) `skip` The KVList pair is not copied to the log. Changes Take Effect: Immediately

Example

```
[log-filter-data]
PASSWORD=hide
```

Here is an example of the log with the PASSWORD option set to hide:

```
message RequestSetCallInfo
    AttributeConsultType            3
    AttributeOriginalConnID         008b012ece62c8be
    AttributeUpdateRevision         2752651
    AttributeUserData               [111] 00 27 01 00
           'DNIS'                   '8410'
           'PASSWORD'               '****'
           'RECORD_ID'              '8313427'
    AttributeConnID                 008b012ece62c922
```

SML Section

This section must be called sml, which stands for System Management Layer. Table 12 describes the options you configure in this section.

Table 12: SML Option

Option	Description
suspending-wait-timeout	Specifies a timeout, in seconds, after the Stop Graceful command is issued to an application within the Solution Control Interface, during which the status of the application should change to Suspending if the application supports graceful shutdown. If the status of the application does not change to Suspending before the timeout expires, it is assumed that the application does not support graceful shutdown, and it is stopped ungracefully. This option is defined on the Annex tab of the Stat Server Application object. **Note:** Stat Server does not support graceful shutdown. Default Value: 10 Valid Values: 5–600 Changes Take Effect: Immediately

Common Section

This section must be named `common`. Table 13 describes the options you configure in this section.

> **Warning!** Use these options only when requested by Genesys Technical Support.

Table 13: Common Option

Option	Description
rebind-delay	Specifies the delay, in seconds, between socket-bind operations that are being executed by Stat Server. Use this option if the Stat Server has not been able to successfully occupy a configured port. Default Value: `10` Valid Values: `0–600` Changes Take Effect: After restart

Chapter

8 Installing Stat Server

You must configure a Stat Server `Application` object before installing the Stat Server application. Read Chapter 4 for this and other important information. You need not uninstall prior releases of Stat Server to install a newer release. This chapter, nonetheless, provides uninstallation procedures, as well as installation procedures, to address the case where you want to permanently remove Stat Server from your machine.

This chapter contains the following topics:

Installing Stat Server Following Wizard Configuration

This section describes how to install Stat Server following wizard configuration.

On UNIX

1. In the directory to which you copied the Stat Server installation package, locate the `install.sh` shell script.
2. Run this script from the command line by typing: `sh install.sh`.
3. When prompted, specify the host name of the computer on which you want Stat Server installed.
4. Specify the destination directory into which you want Stat Server installed.
5. If prompted for which version of the product to install (32- or 64-bit), choose the version appropriate for your environment.

As soon as the installation process completes, a message announces that installation was successful. The process creates a directory with the name specified during the installation, and places Stat Server in it. The installation routine then prompts you to install each of the Stat Server Java Extensions (MCR, OCC, and VCB) if the Extension installation packages were also deployed. Follow the steps described for each Extension starting with Step 2 on page 86.

On Windows

1. Open the directory to which you copied the Stat Server installation package.
2. Locate and double-click `setup.exe` to start installation.
3. If the installation routine detects previously installed Stat Server applications on your machine, you are prompted to do one of the following:
   - Install a new instance.
   - Perform maintenance on the existing application. To do this, select your Stat Server application.
4. Specify the destination directory into which you want Stat Server installed.
5. Click `Install` and `Finish` to complete the installation.

The installation routine installs your Stat Server application automatically as a Windows service.

If you run the Stat Server installation package from the *Real-Time Metrics Engine* CD, Stat Server automatically installs the MCR, OCC, and VCB Stat Server Java Extensions as well.

Installing Stat Server Following Manual Configuration

This section describes how to install Stat Server on UNIX and Windows platforms if you manually configured a Stat Server `Application` object within Configuration Manager.

On UNIX

1. On the Real-Time Metrics Engine 8.0 product CD in the appropriate `statserver/`*`operating_system`*`/` directory, locate the `install.sh` shell script.
2. Run this script from the command line by typing `install.sh`.

3. When prompted, specify the host name of the computer on which you want to install Stat Server.
4. When prompted, specify:
   a. The host name of the computer on which Configuration Server is running.
   b. The port client that applications use to connect to Configuration Server.
   c. The user name used to log in to the Configuration Layer.
   d. The password used to log in to the Configuration Layer.
5. The installation displays the list of Application objects of `StatServer` type configured for this host. Type the number of the Stat Server `Application` you want installed.
6. Specify the destination directory into which you want Stat Server installed.
7. If prompted for which version of the product to install, (32- or 64-bit), select the version appropriate for your environment.

As soon as the installation process completes, a message announces that installation was successful. The process creates a directory with the name specified during the installation, and places Stat Server in it. The installation routine then prompts you to install each of the Stat Server Java Extensions (MCR, OCC, and VCB) if the Extension installation packages were also deployed. Follow the steps described for each Extension, starting with Step 2 on page 86.

On Windows

1. From the Real-Time Metrics 8.0 CD, go to the `\statserver\windows` subdirectory.
2. Locate and double-click `setup.exe` to start installation.
3. If the installation routine detects previously installed Stat Server applications on your machine, you are prompted to either install a new instance or perform maintenance on the existing application. Select the former.
4. Specify the parameters for connecting to the Configuration Server where your Stat Server `Application` object has been configured.
5. Select your Stat Server application.
6. Specify the destination directory into which you want Stat Server installed.
7. Click `Install` and `Finish` to complete the installation.

The installation routine installs your Stat Server application automatically as a Windows service.

If you run the Stat Server installation package from the *Real-Time Metrics Engine* CD, Stat Server automatically installs the MCR, OCC, and VCB Stat Server Java Extensions as well.

Installing Java Extensions

Before installing a Stat Server Java Extension, you must have configured a Stat Server `Application` object and installed the Stat Server application on your machine. On the *Real-Time Metrics Engine* CD, Genesys provides the installation packages for MCR, OCC, and VCB Java Extensions, which are delivered in five `.jar` files:

- `eServiceContactStat.jar`
- `eServiceInteractionStat.jar`
- `eServiceSystemStat.jar`
- `OCCStatExtension.jar`
- `VCBStatExtension.jar`

You deploy these files in three separate installations.

Installing the eServices Extensions

You can install the `eService` Java Extensions, which are used for eServices (formerly known as Multimedia), on Windows and/or UNIX platforms.

On Windows

1. In the `\ext\mcr\` subdirectory of your deployed Stat Server installation package, locate and double-click `setup.exe`.
2. If the installation routine detects one or more previously installed extension on your machine, you are prompted to either install a new instance or perform maintenance on the existing extension. Select the former.
3. When prompted, specify the root folder of the Stat Server installation (for example, `C:\GCTI\StatServer_1)`, and click `Next`.

The installation routine deploys the `eServiceContactStat.jar, eService InteractionStat.jar,` and `eServiceSystemStat.jar` files in the `\java\ext\` subdirectory of your installed application.

On UNIX

1. On the *Real-Time Metrics Engine* CD, navigate to the `/ext/mcr/` subdirectory.
2. Run the `install.sh` script from the command line by typing:

   ```
   sh install.sh
   ```

3. When prompted, specify the full destination path where you want the MCR extension deployed on your machine.

   If the installation routine detects one or more installed extensions in the specified path, it prompts you to overwrite them or exit.

The installation routine deploys the `eServiceContactStat.jar, eService InteractionStat.jar,` and `eServiceSystemStat.jar` files in the `/java/ext` subdirectory of the path that you specified.

Installing the Outbound Contact Extension

You can install the `OCCStatExtension` Java Extension, which is used for the Outbound Contact solution, on Windows and/or UNIX platforms.

On Windows

1. In the `\ext\occ\` subdirectory of your deployed Stat Server installation package, locate and double-click `setup.exe`.
2. When prompted to specify the destination folder, indicate the root folder of the Stat Server installation (for example, `C:\GCTI\StatServer_1)` and click `Next`.

> Note: Select this folder carefully. The default choice provided by the installation routine likely differs from your intended destination.

The installation routine deploys `OCCStatExtension.jar` in the `\java\ext` subdirectory of your installed application.

On UNIX

1. On the *Real-Time Metrics Engine* CD, navigate to the `/ext/occ/` subdirectory.
2. Run the `install.sh` script from the command line by typing:

   ```
   sh install.sh
   ```

3. When prompted, specify the full destination path where you want the OCC extension deployed on your machine.

   If the installation routine detects one or more installed extensions in the specified path, it prompts you to overwrite them or exit.

The installation routine deploys `OCCStatExtension.jar` in the `/java/ext` subdirectory of the path that you specified.

Installing the Voice Callback Extension

You can install the `VCBStatExtension` Java Extension, which is used for the Voice Callback option of Enterprise Routing, on Windows or UNIX platforms.

On Windows

1. In the `\ext\vcb\` subdirectory of your deployed Stat Server installation package, locate and double-click `setup.exe`.
2. When prompted to specify the destination folder, indicate the root folder of the Stat Server installation (for example, `C:\GCTI\StatServer_1)` and click `Next`.

> Note: Select this folder carefully. The default choice provided by the installation routine likely differs from your intended destination.

The installation routine deploys `VCBStatExtension.jar` in the `\java\ext\` subdirectory of your installed application.

On UNIX

1. On the *Real-Time Metrics Engine* CD, navigate to the `/ext/vcb/` subdirectory.
2. Run the `install.sh` script from the command line by typing:

   ```
   sh install.sh
   ```

3. When prompted, specify the full destination path where you want the VCB extension deployed on your machine.

   If the installation routine detects one or more installed extensions in the specified path, it prompts you to overwrite them or exit.

The installation routine deploys `VCBStatExtension.jar` in the `/java/ext` subdirectory of the path that you specified.

Installing Stat Server Silently

InstallShield Silent is a third-party installation program that Genesys uses to facilitate the electronic software distribution for both server and GUI applications on Windows platforms. “Silent” installations eliminate the need for interactive dialog during the installation process. Instead, you create a single response file filled with the necessary parameters that InstallShield Silent references during subsequent silent installations.

After creating a response file and performing a silent installation, review the log file produced for a successful result code or any errors encountered.

Note: All Local Control Agent (LCA)–related functionality is disabled during a silent installation. This means that you will not be able to start and stop applications, detect their failures, or communicate their roles in the redundancy context.

Create the Response File

To select setup options and automatically record the InstallShield Silent response file, perform Steps 1–4 below.

Note: Use this procedure instead of double-clicking `setup.exe` from Windows Explorer.

---

1. Open a console window.
2. Enter the path to the directory in which you deployed the Stat Server installation files.
3. Issue the following command using the `--host` and `-r` command-line parameters:

   ```
   setup.exe --host=hostcomputer -r
   ```

Issuing this command creates a response file named `setup.iss`, which is stored in the main Windows directory.

4. Complete the installation choosing the options you want to deploy.

Then use the configured `setup.iss` file any time you must install an application with the specified parameters.

The script in Figure 3 shows a sample `setup.iss` file for Stat Server. Blank lines have been inserted between sections to facilitate reading.

Play Back the Response File

To install Stat Server silently on a local workstation:

1. Open a console window.
2. Enter the path to the directory where you have deployed Stat Server installation files.
3. Launch InstallShield Silent using the following command:
   `setup.exe --app=`*`applname`* `--host=`*`localhost`* `-s /f1"ResponsePath" /f2"LogPath"`

   where:
   - `--app=`*`applname`* is the name of the application you are installing. This name should exactly match the application name in the Configuration Database. Two dashes must precede the `app` parameter.
   - `--host=`*`localhost`* is the host name of the computer on which you want the application installed. This name should exactly match the host name in the Configuration Database. Two dashes must precede the `host` parameter.
   - `-s` is the silent flag.
   - `/f1"ResponsePath"` is an optional command-line parameter that specifies the full path to the response file. If you do not specify this parameter, InstallShield uses the default file name of `setup.iss` and looks for the file in the directory containing `Setup.exe`.
   - `/f2"LogPath"` is an optional command-line parameter that specifies the full path to the log file. If you do not specify this parameter, InstallShield assigns the default file name of `setup.log` and stores the file in the same directory used by `/f1`.

Note: Do not use spaces between the `/f1` or `/f2` parameter and its value in double quotation marks.

You could issue any of the following commands to play back the response file:

- `setup.exe -s`

  Both `/f1"*\`*`Setup.exe directory`*`\setup.iss"` and `/f2"*\`*`Setup.exe directory`*`\setup.log"` are implied on a Windows NT system.

- `setup.exe -s /f1"c:\winnt\MonitorAgents.iss"`

  `/f2"*\`*`Setup.exe directory`*`\setup.log"` is implied on a Windows NT system.
- `setup.exe -s /f2"c:\winnt\SuzysLog.log"`

  `/f1"*\`*`Setup.exe directory`*`\setup.iss"` is implied on a Windows NT system.

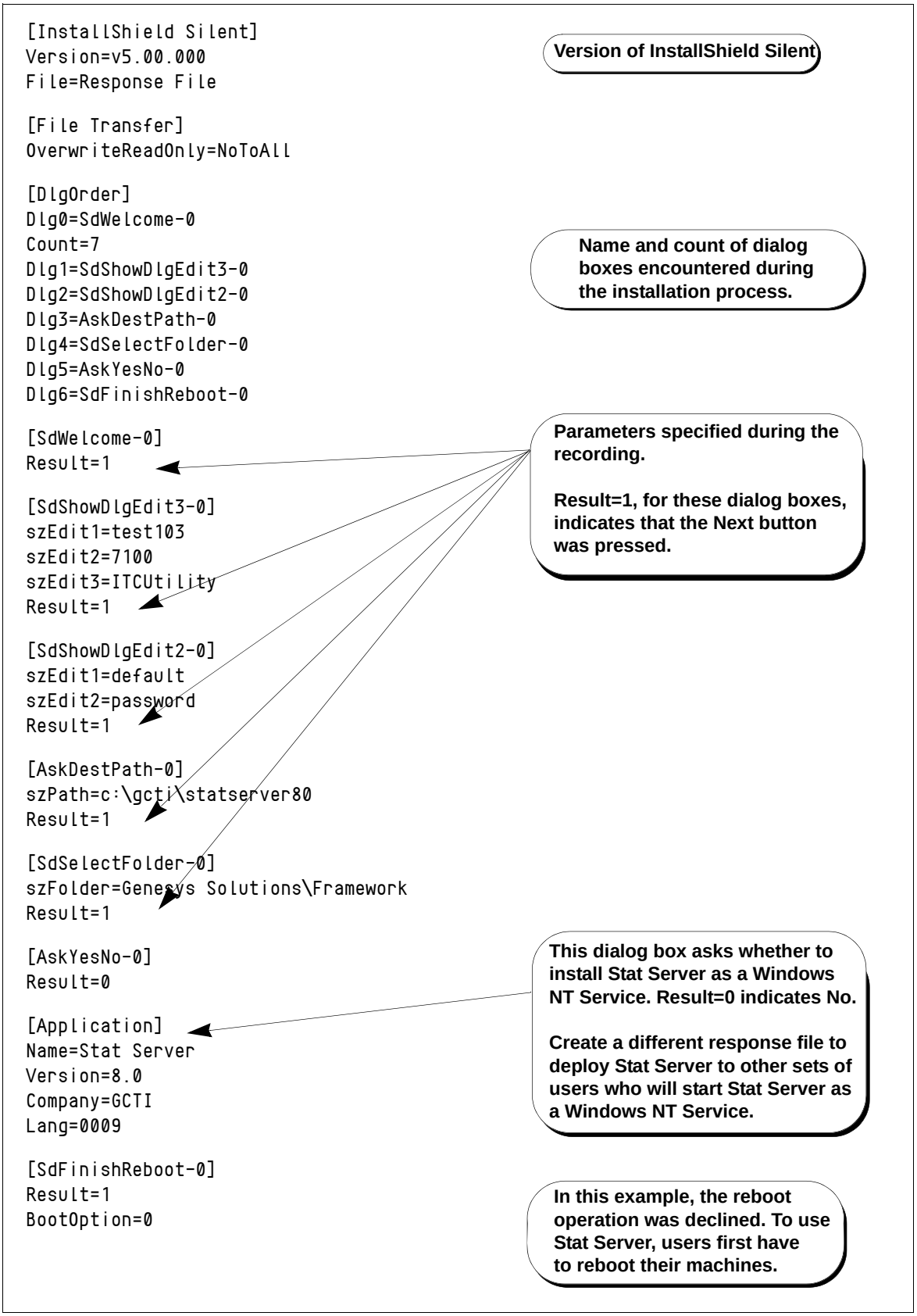


Figure 3: Sample Stat Server setup.iss File

Remote Silent Installation

Using standard operating system tools, your network administrator can design the scripts necessary to deploy the response file to any number of machines where Stat Server can be deployed remotely.

Analyze the Log File

When the response file has completed its playback, InstallShield Silent prints installation results to the file specified by `/f2`, as shown in the previous section. This log file contains three sections:

- The `[InstallShield Silent]` section identifies the version of InstallShield Silent used in the silent install and the file type as a log file.
- The `[Application]` section identifies the Stat Server application and version, the Genesys company abbreviation, and the language used in the recording. (`LANG=0009` indicates English.)
- The `[ResponseResult]` section contains the result code indicating whether the silent install succeeded. See Table 14 for the meanings of the `ResultCode` key given in this section.

Table 14: Log File ResultCode Key

Value	Meaning
0	Success
–1	General error
–2	Invalid mode
–3	Required data not found in the setup.iss file
–4	Not enough memory
–5	File does not exist
–6	Cannot write to the response file
–7	Unable to write to the uninstallation log file
–8	Invalid path to the InstallShield Silent response file
–9	Not a valid list type (string or number)
–10	Data type is invalid
–11	Unknown error during setup
–12	Dialog boxes are out of order
–51	Cannot create the specified folder
–52	Cannot access the specified file or folder
–53	Invalid option selected

The text in Figure 4 shows the contents of a log file for a Stat Server application that was successfully installed using InstallShield Silent. Again, blank lines have been inserted to facilitate reading.

```
[InstallShield Silent]
Version=v5.00.000
File=Log File

[Application]
Name=Stat Server
Version=8.0
Company=GCTI
Lang=0009

[ResponseResult]
ResultCode=0
```

Figure 4: Log File Showing Successful Deployment

Uninstalling the Stat Server Application

To uninstall a Stat Server application, you must first stop it. Refer to "Stopping Stat Server" on page 98 for this information. Uninstalling the Stat Server application differs from uninstalling its `Application` object in Configuration Server, which is described in "Stat Server Wizard—Uninstall Mode" on page 28.

Use either of the following two methods for uninstalling a Stat Server application from your machine:

- From the Control Panel
- During installation maintenance

From the Control Panel

1. Open `Add/Remove Programs`.
2. Locate and select the desired `Genesys Stat Server 8.0` application.
3. Click `Remove`.

During Installation Maintenance

1. From the path where the Stat Server Wizard deployed your installation package, or from the *Real-Time Metrics Engine* CD, double-click `setup.exe`.
2. Navigate to the `Maintenance Setup Type` screen and select the `Maintenance of the existing installation` option.
3. Select your Stat Server application from the list box and click `Next`.

4. At the Welcome screen, click Remove, and then Next.
5. At the Confirm Uninstall page, click Yes.
6. At the Maintenance Complete screen, click Finish.
7. At the After Installation screen, click Next. (This screen appears only on a Windows 2003 platform).

The Uninstall Wizard removes program registry entries, all shared and standard program files originally deployed by the installation routine, and then the program folder (or directory) itself, if it is empty.

So that you can reuse them later if desired, the Uninstall Wizard does not delete:

- The Stat Server database.
- The Stat Server Application object from the Configuration Server Database.
- Any log, snapshot, batch, or personal files in the Stat Server working directory.
- The java folder, unless you specifically uninstall Stat Server Java Extensions.

Refer to "Stat Server Wizard—Uninstall Mode" on page 28 to remove the Application object from Configuration Server.

Chapter

9 Starting and Stopping Stat Server

This chapter contains procedures for starting and stopping Stat Server on the supported platforms. Start procedures assume that you have properly configured and installed Stat Server. If not, refer to Chapters 4 and 8.

This chapter contains these sections:

You can start and stop your entire solution, including Stat Server, from the Solution Control Interface (SCI), which is the recommended approach. This chapter describes this method as well.

What Must Be Running Prior to Start

Genesys recommends that you start Stat Server with the following applications already running:

- Configuration Server
- RDBMS (if you use Stat Server with database functionality)
- DB Server (if you use Stat Server with database functionality)
- Java Runtime Environment (JRE) 1.4 or later (if you have configured Stat Server Java Extensions)

> **Note:** Genesys does not support JRE 1.6.

Starting Stat Server

You can start Stat Server in any of the following ways:

- From the Genesys SCI
- Manually on UNIX
- Manually on Windows
- On Windows as a Windows Service

Note: Prior to opening statistics at startup, Stat Server now checks that the binary format of the backup file is compatible with the running instance of Stat Server.

Starting Stat Server Using SCI

1. From the Applications view in SCI, select your Stat Server Application object on the list pane.
2. Click the Start button on the toolbar, or select Start from either the Action menu or the shortcut menu. (Right-clicking your Application object displays the shortcut menu.)
3. Click Yes in the confirmation box that appears. Your Stat Server application starts.

For information about how to use SCI, refer to *Framework 8.0 Solution Control Interface Help*.

Starting Stat Server on UNIX

1. Go to the directory where you have installed Stat Server.
2. At the command line, type:

   ```
   ./run.sh
   ```

   Or, type the name of the Stat Server executable followed by the appropriate command-line parameters using the following syntax:

   ```
   ./statserv -host hostname -port portno -app application
   ```

   where:

   - *hostname* refers to the name of the host on which Configuration Server is running.
   - *portno* refers to the communication port that client applications must use to connect to Configuration Server.
   - *application* refers to the name of the Stat Server Application object as defined to the Configuration Server.

Note: If the host or application name contains spaces or hyphens (-), enclose it in double quotation marks.

For example, to start Stat Server with parameters specifying the host as `cs-host`, port as `2020`, and name as `Stat Server 03`, type:

```
./statserv -host "cs-host" -port 2020 -app "Stat Server 03"
```

Starting Stat Server on Windows

Start Stat Server from the `Start` menu or open a console window, go to the directory where Stat Server is installed, and type the following command:

```
./statserv.exe -host hostname -port portno -app application
```

where:

- *`hostname`* refers to the name of the host on which Configuration Server is running.
- *`portno`* refers to the communication port that client applications must use to connect to Configuration Server.
- *`application`* refers to the name of the Stat Server `Application` object as defined to the Configuration Server.

Note: If the host or application name contains spaces or hyphens (-), enclose it in double quotation marks.

For example, to start Stat Server with parameters specifying the host as `cs-host`, port as `2020`, and name as `Stat Server 03`, from the Stat Server working directory, type:

```
statserv.exe -host "cs-host" -port 2020 -app "Stat Server 03"
```

If you used the Stat Server Wizard to configure Stat Server, the configuration process created a batch file named `StatServer_run.bat` and placed it in the Stat Server working directory. This file contains the complete command, with configured parameters, you need to start Stat Server. Simply double-click this file to start the application.

Starting Stat Server as a Windows Service

1. Open the Windows `Control Panel` and double-click the `Services` icon. The `Services` dialog box opens.
2. Select your Stat Server service from the list and click `Start`. (If you did not install Stat Server as a Windows Service, your application does not appear for selection in the `Services` list box.)

Note: Since you can install the Local Control Agent (LCA) as a Windows Service with the user interface disabled, all servers started through SCI, in this case, are started without a console, unless you specifically select the Allow Service to Interact with Desktop check box for both LCA and Stat Server.

Stopping Stat Server

You can stop Stat Server in any of the following ways:

- From the Genesys Solution Control Interface (SCI). (This is the recommended method.)
- Manually on UNIX.
- Manually on Windows.
- Via the Windows Control Panel.

Note: Be sure that the autorestart property is cleared for the Stat Server Application in the Configuration Manager to prevent Stat Server from self-starting.

Stopping Stat Server Using SCI

If you are using LCA and a Solution Control Server (SCS), you can stop Stat Server from SCI:

1. From the Applications view in SCI, select your Stat Server Application object on the list pane.
2. Click Stop on the toolbar, or select Stop either from the Action or the shortcut menu. (Right-clicking your Application object opens the shortcut menu.)
3. Click Yes in the box that appears to confirm your command.

SCI stops your Stat Server application.

Stopping Stat Server on UNIX

Stop Stat Server on UNIX using any one of the following methods:

- On the command line, type kill -SIGTERM *processid* where *processid* is Stat Server's UNIX process ID.
- Press ^C from the active Stat Server window.
- If you are using LCA and SCS, you can stop Stat Server from running on UNIX using SCI (described in the preceding subsection).

Stopping Stat Server on Windows

If Stat Server is running as an application—not as a Windows Service—switch to its console window and press `Ctrl+Break` to stop it. If you are running Stat Server as a Windows NT Service, you should stop it only from the Services Control Manager.

To stop Stat Server running as a Windows NT Service:

1. Open the `Control Panel` and double-click the `Services` icon. The `Services` dialog box opens.
2. Select your Stat Server service from the list and click `Stop`.

Chapter

10 Application Files

The Stat Server installation routine creates a root application folder with two subfolders:

- `java`
- `sql_scripts`

Tables 15, 16, and 17 in this chapter describe the files comprising each folder.

Warning! Do not attempt to run the SQL scripts manually because of the potential for data loss. They are intended only for Stat Server's internal use and advanced database administrators.

Table 15: Contents of the Root Folder

File Name	Description
common.lms	File storing log messages common to all Genesys components.
ip_description.xml	File storing installation package content.
read_me.html	File containing general information about the installation package.
startServer.bat	Batch file containing the Stat Server executable and command-line parameters used to start Stat Server.
statserv.exe	Application executable.
statserv.pdb	File for advanced troubleshooting of Stat Server on Windows operating systems.
StatServer.lms	File storing Stat Server–specific log messages.

Table 15: Contents of the Root Folder (Continued)

File Name	Description
java subfolder	Subfolder. See Table 17 for folder contents.
sql_scripts subfolder	Subfolder containing five subfolders, holding SQL scripts for each of the following RDBMS types: • Sybase • Informix • DB2 • Oracle • Microsoft SQL See Table 16 for the contents of each subfolder.

Table 16: Contents of the sql_scripts/[dbtype] Subfolder

File Name	Description
login_[dbtype].sql	SQL script that creates the `LOGIN` table (and indexes and procedures, as necessary) for the RDBMS type.
qinfo_[dbtype].sql	SQL script that creates the `QINFO` table (and indexes and procedures, as necessary) for the RDBMS type.
status_[dbtype].sql	SQL script that creates the `STATUS` table (and indexes and procedures, as necessary) for the RDBMS type.
status_ixnid_[dbtype].sql	A variation of the `status_[dbtype]` script that creates the `STATUS` table with one additional field, `IxnID`.
status_ltime_[dbtype].sql	A variation of the `status_[dbtype]` script that creates the `STATUS` table with two additional fields, `StartLocalTime` and `EndLocalTime`, to store the start and end times in the local time zone.
status_ltime_ixnid_[dbtype].sql	A variation of the `status_[dbtype]` script that creates the `STATUS` table with three additional fields, `IxnID`, `StartLocalTime`, and `EndLocalTime`, to store the start and end times in the local time zone.
voice_reasons_[dbtype].sql	SQL script that creates the `VOICE_REASONS` table (and indexes and procedures, as necessary) for the RDBMS type.

Table 17: Contents of the java Subfolder

File Name	Description
ssjcldr.class	Java class loader; a member of the Stat Server Java host environment
statserver.jar	Library that is part of the Stat Server Java SDK, which, in turn, is part of the Stat Server Java host environment
statserver_impl.jar	A member of the Stat Server Java host environment
kvlists.jar	Library that is part of the Stat Server Java SDK, which, in turn, is part of the Stat Server Java host environment. Stat Server uses this file in conjunction with Stat Server Java extensions.
kv65_adapter.jar	Library that is part of the Stat Server Java SDK, which, in turn, is part of the Stat Server Java host environment
ext folder	Directory to store the Genesys solution-specific extensions, such as: • `eServiceContactStat.jar` • `eServiceInteractionStat.jar` • `eServiceSystemStat.jar` • `OCCStatExtension.jar` • `VCBStatExtension.jar`
lib folder	Directory to store the Genesys' solution-specific libraries, such as: • `dsw_api_java.jar` • `dsw_extension_core.jar` • `dsw_transformers.jar`

Appendix

A Physical Data Models for Stat Server Tables

This appendix describes the database tables to which Stat Server stores data if the `status-table`, `qinfo-table`, `login-table`, and/or `voice-reasons-table` configuration options are enabled. (These configuration options are described in Table 2 beginning on page 44.) The information in this appendix is divided among the following topics:

Introduction

Stat Server stores status data about agents in the `STATUS` table and data about queues in the `QINFO` table. Stat Server also maintains information about agent login and logout events in its `LOGIN` table. These tables are independent and do not reference each other. Genesys Info Mart and custom reporting use these tables.

The `VOICE_REASONS` table stores hardware and software reasons for agents to change or continue the `Ready` and `NotReady` states and the `AfterCallWork` work mode, when handling voice interactions. Genesys Info Mart uses this table and makes this data available for custom reporting.

DBID refers to the database identifier that the Configuration Layer assigns to a telephony object when an enterprise is configured. For example, after you set up an agent either manually in the `Persons` folder of Configuration Manager or

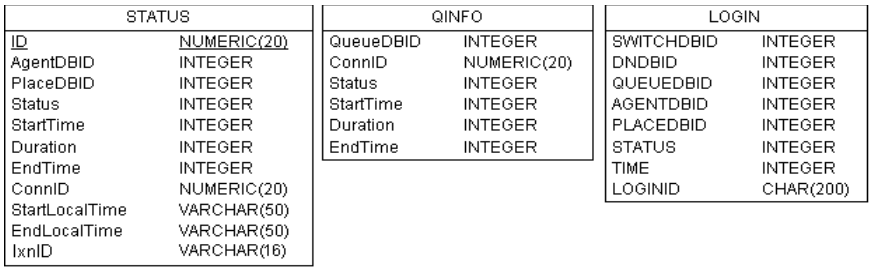
using the Framework Configuration Wizard, the Configuration Layer assigns a unique DBID to that agent.

Note: Stat Server, while functioning in backup mode, does not write data to its database, even if configured to do so. This enables the primary or backup Stat Server, while functioning as the primary application, to store data to the same database.

Table Schema by RDBMS

Figures 5 through 9 depict Stat Server table schema for the supported RDBMSs.

Note: Data from the `VOICE_REASONS` table is not available for custom reporting directly from the Stat Server database. Therefore, the structure of the `VOICE_REASONS` table is not provided in this guide.

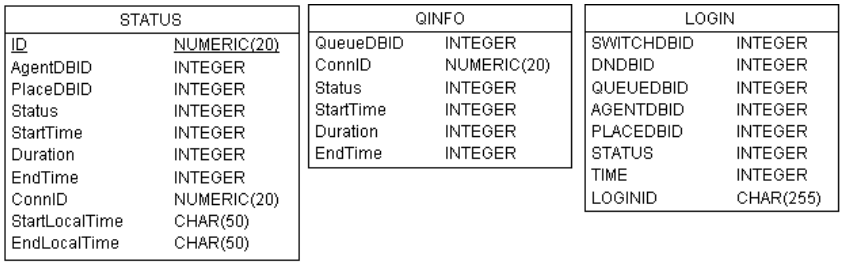


STATUS	
ID	NUMERIC(20)
AgentDBID	INTEGER
PlaceDBID	INTEGER
Status	INTEGER
StartTime	INTEGER
Duration	INTEGER
EndTime	INTEGER
ConnID	NUMERIC(20)
StartLocalTime	VARCHAR(50)
EndLocalTime	VARCHAR(50)
IxnID	VARCHAR(16)

QINFO	
QueueDBID	INTEGER
ConnID	NUMERIC(20)
Status	INTEGER
StartTime	INTEGER
Duration	INTEGER
EndTime	INTEGER

LOGIN	
SWITCHDBID	INTEGER
DNDBID	INTEGER
QUEUEDBID	INTEGER
AGENTDBID	INTEGER
PLACEDBID	INTEGER
STATUS	INTEGER
TIME	INTEGER
LOGINID	CHAR(200)

Figure 5: Table Schema for a DB2 Stat Server Database



STATUS	
ID	NUMERIC(20)
AgentDBID	INTEGER
PlaceDBID	INTEGER
Status	INTEGER
StartTime	INTEGER
Duration	INTEGER
EndTime	INTEGER
ConnID	NUMERIC(20)
StartLocalTime	CHAR(50)
EndLocalTime	CHAR(50)

QINFO	
QueueDBID	INTEGER
ConnID	NUMERIC(20)
Status	INTEGER
StartTime	INTEGER
Duration	INTEGER
EndTime	INTEGER

LOGIN	
SWITCHDBID	INTEGER
DNDBID	INTEGER
QUEUEDBID	INTEGER
AGENTDBID	INTEGER
PLACEDBID	INTEGER
STATUS	INTEGER
TIME	INTEGER
LOGINID	CHAR(255)

Figure 6: Table Schema for an Informix Stat Server Database

Note: For the Informix RDBMS, the `STATUS` table does not include the `IxnID` field.

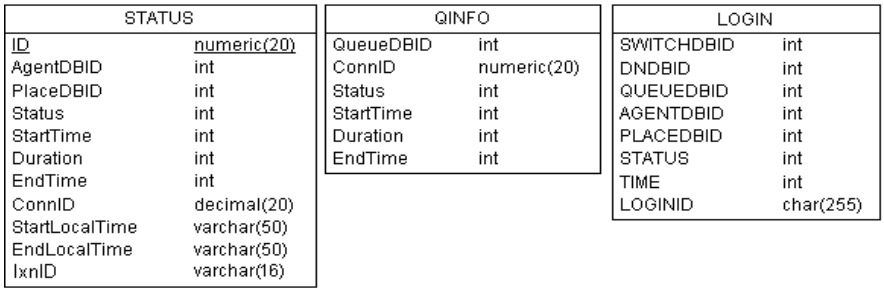


Figure 7: Table Schema for a Microsoft SQL Stat Server Database

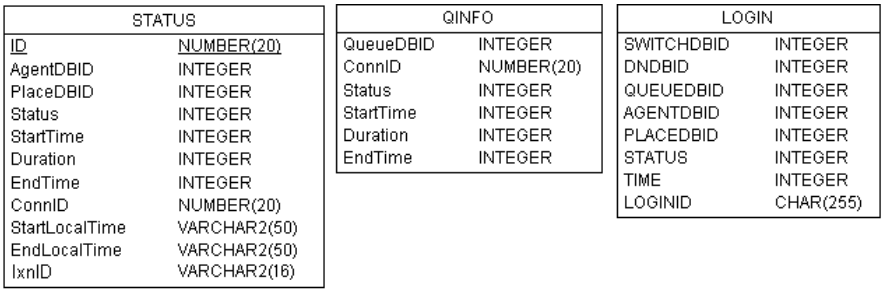


Figure 8: Table Schema for an Oracle Stat Server Database

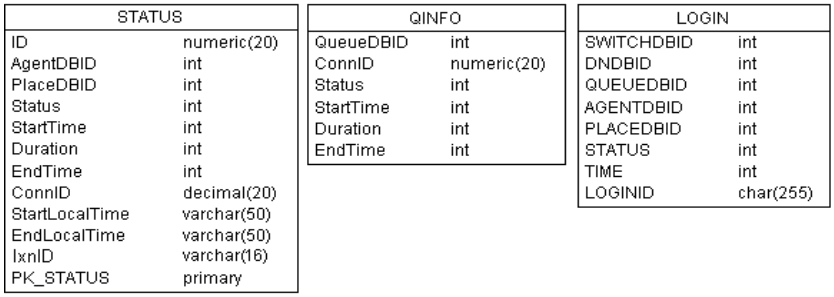


Figure 9: Table Schema for a Sybase Stat Server Database

Table and Column Descriptions

The Stat Server database contains four tables:

- `LOGIN`, described on page 108
- `QINFO`, described on page 109
- `STATUS`, described on page 110
- `VOICE_REASONS`, described on page 112

Note: The `VOICE_REASONS` table is not included in Stat Server deployment for Sybase and Informix RDBMSs.

The LOGIN Table

The LOGIN table contains the history of login and logout activity for resources on both voice and multimedia channels. Stat Server writes to this table if the login-table configuration option is set to yes.

Stat Server detects login activity, for T-Server and SIP Server clients, upon receipt of an EventAgentLogin TEvent; Stat Server detects logout upon receipt of EventAgentLogout.

For medias reported through Interaction Server, the pair of EventAgentLogin and EventMediaAdded events are used in Stat Server logic to determine agent readiness to process interactions on a particular media channel. The EventMediaRemoved and EventAgentLogout pair are the triggering logout events.

Table 18 describes the LOGIN table's fields, which are presented in order of appearance.

Table 18: Field Descriptions for the LOGIN Table

Field Name	Description
SWITCHDBID	The DBID of the switch at whose DN the agent has logged in or out.
DNDBID	The DBID of the DN at which the agent has logged in or out. This value is 0 (zero) if the agent has logged in to or logged off a media channel.
QUEUEDBID	The DBID of the ACD queue where the agent has logged in or out.
AGENTDBID	The DBID of the agent who has logged in or out.
PLACEDBID	The DBID of the place where the agent has logged in or out.
STATUS	1 if the agent has logged in. 0 if the agent has logged out.
TIME	Time, in seconds since 1 January 1970 UTC (Universal Time Coordinated), when the related login or logout event occurred.
LOGINID	The login ID of the resource for this record. The initial size of this field, as defined in the login.sql script for your RDBMS, is 255 characters, but you can adjust it as appropriate for your environment. Where the agent has logged in to or logged off a media channel, this field stores the media type. Stat Server gathers this information from the MediaType attribute of the triggering TEvent.

The QINFO Table

The QINFO table contains the history of voice interaction activities from the perspective of one or more mediation DNs that are registered to the Stat Server application. Stat Server writes to this table if the qinfo-table configuration option is set to yes. Table 19 describes this table's fields, which are presented in their order of appearance.

Table 19: Field Descriptions for the QINFO Table

Field Name	Description
QueueDBID	The queue's DBID.
ConnID	An identifier that T-Server assigns to the connected call. The value in this field is 0 (zero) if the status is not related to the call. In multi-site scenarios, if the first transfer connection ID differs from the current connection ID associated with the call, the value stored in this field is the first transfer connection ID. Prior to Stat Server release 7.0.3, this field stored the current connection ID.
Status	The status of the transition of a call through a queue whose DBID is displayed in the QueueDBID field (of this table). The possible values of 1–9 indicate the following statuses and durations: **Call Status** / **Code** / **Duration** Diverted from queue / 1 / Time in queue Abandoned within queue / 2 / Time in queue Diverted from queue (answered while ringing) / 3 / Time in queue plus time spent ringing Diverted from queue (abandoned while ringing) / 4 / Time in queue plus time spent ringing Party changed from queue (for consultation calls only) / 5 / Time in queue plus time spent ringing until party changed Diverted from queue (forwarded) / 6 / Time in queue Call cleared[a] (for virtual queues only) / 7 / Time in queue Call cleared after being stuck on a distribution DN / 8 / Time in queue Call cleared after being stuck while ringing at an agent's DN / 9 / Time at DN

Table 19: Field Descriptions for the QINFO Table (Continued)

Field Name	Description
StartTime	A sequence number representing the date and time when the status displayed in the `Status` field (of this table) began. The sequence begins with January 1, 1970, 12:01 AM UTC and increments every second. For example, `878159351` represents October 29, 1997, 13:09:11. Each new second represents an increment of `1` in the sequence.
Duration	The duration, in seconds, of the status displayed in the `Status` field.
EndTime	A sequence number representing the date and time when the status displayed in the `Status` field (of this table) ended. The sequence begins with January 1, 1970, 12:01 AM, UTC, and increments each second. For example, `878159351` represents October 29, 1997, 13:09:11. Each new second represents an increment of `1` in the sequence.

a. Indicates that a call was cleared from a virtual queue (diverted to an agent's DN from another virtual queue). This status is based on the `CallCleared` retrospective, instantaneous action. (Refer to the *Framework 8.0 Stat Server User's Guide* for a description of this action.)

The STATUS Table

The `STATUS` table contains the history of status changes for agent and place resources. This table also stores the current status for such resources that persist more than 600 seconds. Stat Server determines place status by the highest ranking action (as defined by Stat Server status priority tables) that occurs at the DNs it includes.

Starting with the 7.6.1 release, Stat Server supports status reporting for multimedia DNs—DNs capable of handling multiple simultaneous interactions. By setting the `multimedia-activity-in-status-table` configuration option to `yes`, Stat Server selectively accounts for non-voice-related actions on multimedia DNs in the status records that are written to this table.

Starting with the 8.0 release, Stat Server records the interaction IDs of multimedia interactions when the `ixn-id-in-status-table` configuration option is set to `yes`.

Note: The `StartLocalTime`, `EndLocalTime`, and `IxnID` fields appear only if the appropriate script was run to create the `STATUS` table. Refer to Table 16 on page 102 for descriptions of the scripts.

Stat Server writes to this table only if the `status-table` configuration option is set to `yes`. Table 20 describes this table's fields, which are presented in their order of appearance.

Table 20: Field Descriptions for the STATUS Table

Field Name	Description
ID	A unique key field used for internal purposes. Upon reaching 4,294,967,296 (that is, 2^{32}), Stat Server restarts the counter reusing all values starting from `1`, provided that no records are associated with the IDs to be reused. **Warning!** To store new records after the number of records in the `STATUS` table reaches 4,294,967,296, clear the `STATUS` table. To keep previous records, back up this table's data into a backup database prior to clearing the table.
AgentDBID	The database ID (DBID) of an agent or `0` (zero).
PlaceDBID	The DBID of the place associated with the agent whose ID is displayed in the `AgentDBID` field of this table or `0` (zero).

Status	The status of the agent whose DBID appears in the `AgentDBID` field or the status of the place whose DBID appears in the `PlaceDBID` field. If the values in the `AgentDBID` and `PlaceDBID` fields are both not `0`, then the agent and his or her place share the same status.
	The following lists `STATUS` field values and their significance:
	`4 WaitForNextCall (Ready)`
	`5 OffHook`
	`6 CallDialing`
	`7 CallRinging`
	`8 NotReadyForNextCall`
	`9 AfterCallWork`
	`13 CallOnHold`
	`16 ASM_Engaged`
	`17 ASM_Outbound`
	`18 CallUnknown`
	`19 CallConsult`
	`20 CallInternal`
	`21 CallOutbound`
	`22 CallInbound`
	`23 LoggedOut`

StartTime	A sequence number representing the date and time when the status displayed in the `Status` field (of this table) began. The sequence begins with January 1, 1970, 12:01 AM UTC and increments each second. For example, `878159351` represents October 29, 1997, 13:09:11. Each new second is represented by an increment of `1` in the sequence.
Duration	The duration, in seconds, of the status displayed in the `Status` field in this table.

Table 20: Field Descriptions for the STATUS Table (Continued)

Field Name	Description
EndTime	A sequence number representing the date and time when the status displayed in the `Status` field (of this table) ended. The sequence begins with January 1, 1970, 12:01 AM, UTC, and increments each second. For example, `878159351` represents October 29, 1997, 13:09:11. Each new second is represented by an increment of `1` in the sequence. Beginning with the 7.1 release, this field holds a `0` (zero) value if the status does not complete.
ConnID	An identification number that T-Server assigns to the connected call. The value in this field is `0` (zero) if the status is not related to a voice interaction. In multi-site scenarios, if the first transfer connection ID differs from the current connection ID associated with the call, the value stored in this field is the first transfer connection ID. Prior to Stat Server release 7.0.3, this field stored the current connection ID.
StartLocalTime	A string containing a user-defined format for the local time of status start. The format of the start local time is controlled by the `time-format` option. This field is populated if the `local-time-in-status-table` configuration option has been enabled.
EndLocalTime	A string that contains a user-defined format for the local time of status end. The format of the end local time is controlled by the `time-format` option. This field is populated if the `local-time-in-status-table` configuration option has been enabled.
IxnID	A string that contains the number that Interaction Server assigns to an interaction. The value of this field is null if the `ixn-id-in-status-table` configuration option is set to `off` or if the associated status for this record originated from a source other than Interaction Server. In conjunction with a `yes` setting for the `ixn-id-in-status-table` configuration option, it is also recommended, although not required, that you set `multimedia-activity-in-status-table` to `true`. **Note:** Stat Server does not populate this field for Informix RDBMSs.

The VOICE_REASONS Table

Stat Server writes to the `VOICE_REASONS` table if the `voice-reasons-table` configuration option is set to `yes` in the Stat Server application. This table contains the history of hardware and software reasons for each agent to change or continue the `Ready` and `NotReady` states and the `AfterCallWork` work mode when handling voice interactions. (Hardware reasons are reported by the switch whereas software reason are established at a software level by a request from a software application, such as an agent desktop.)

Stat Server retrieves `Reasons` information from data that is attached to the `EventAgentReady` and `EventAgentNotReady` TEvents for a DN assigned to a place that has a logged-in agent. Stat Server inserts reason records into the table retroactively—a record is added only after the `Reasons` value or work mode has changed or the DN state associated with the reason has ended.

The data from the Stat Server's `VOICE_REASONS` table is not available for custom reporting off the Stat Server database directly; therefore, no description of the `VOICE_REASONS` table structure is provided in this guide.

`Reasons` data is available to users of Genesys Info Mart release 7.2 and higher. Refer to the *Genesys Info Mart Operations Guide* for information about `Reasons` data in the Info Mart database.

Appendix

B Manually Purging Data from the Stat Server Database

Stat Server provides no utility to periodically purge unwanted data from the Stat Server database and Genesys provides no defined procedure for implementing the purge. What data to purge and the purge operation itself are left to your discretion.

The steps, however, are relatively simple:

1. Back up your Stat Server data.
2. Determine your purge criteria—for example, the date beyond which to purge data.
3. For time-related purge criteria, determine the UTC-equivalent integer for the targeted date beyond which you want to purge data.
4. Write and execute an SQL script to purge data based on your criteria.

Determining the Purge Criteria

This appendix provides one approach, based on time, for trimming down the data stored in the Stat Server database. You may want to purge data based on other criteria, such as deactivated resources or status. In addition, you may wish to apply different purge rules to each of the `STATUS`, `QINFO`, and `LOGIN` tables. Tailor the suggestions provided in this appendix to meet your business need.

Time-Related Fields in the Stat Server Database

Data in the Stat Server database is time-stamped in accordance with the time that Stat Server detected events from other servers. (The `UseSourceTimeStamps` feature does not pertain to data stored in the Stat Server database.) The `STATUS`

table holds the following time-related fields to measure when the status of a particular agent or place changes:

- StartTime (and StartLocalTime)
- EndTime (and EndLocalTime)

The QINFO table holds:

- StartTime (and StartLocalTime)
- EndTime (and EndLocalTime)

Finally, the LOGIN tables holds the Time time-related field.

Except for the LocalTime fields in the STATUS table, all time fields are based on Coordinated Universal Time (UTC), which measures the seconds from January 1, 1970, 12:01 AM. To purge data prior to a particular date, you must have the equivalent UTC integer value of your targeted date.

Note: Some EndTime fields may hold 0 values for incompleted statuses. Basing a purge operation solely on this field is not advisable.

Determining the UTC Equivalent for a Selected Date

To determine the number of seconds between your targeted date and January 1, 1970, calculate the number of days between these two dates, and multiply the result by 86,400—the number of seconds in one day. There are numerous websites, such as http://www.timeanddate.com, that can help you to calculate the difference between two dates, or you can query your own RDBMS, using its date-diff functions.

Designing a Purge Script

QINFO, LOGIN, and STATUS are independent tables in the Stat Server database; there are no fields joining these tables; no parent-child inter-relationships exists between them. Therefore, when deleting records, you need not be concerned about maintaining data integrity *in between* these tables, such as the integrity that is preserved by cascade-update and -delete operations for some databases. The absence of data in one Stat Server table has no impact on the content or significance of data in another Stat Server table.

One consideration to weigh in your purge script's design, however, is that of performance. If the volume of unwanted rows is large, executing one delete statement to purge this data will certainly impact RDBMS performance. Therefore, you should break up the operation so that the RDBMS purges data into whatever you determine to be manageable chunks.

The following generalized SQL statement deletes data:

```
DELETE StatServerTable WHERE criteria ;
```

To delete rows from the LOGIN table for resources that logged in prior to July 30, 2001, issue the following query against the database:

```
DELETE LOGIN WHERE Time < 996451200 ;
```

[996,451,200 = 11,533 days (between 1/1/70 and 7/30/01) * 86,400 sec/day]

This assumes that the volume of data in your database prior to July 30, 2001 is of a manageable enough size to be purged by one DELETE statement without adversely impacting performance.

Related Documentation Resources

The following resources provide additional information that is relevant to this software. Consult these additional resources as necessary.

Management Framework

- The *Framework 8.0 Deployment Guide,* which will help you configure and install other Framework components.
- The *Framework 8.0 Stat Server User's Guide,* for information about the makeup of a statistic and the manner in which Stat Server provides data to its clients.
- *Genesys Administrator Help,* for information about configuring Genesys applications using Genesys Administrator.

Genesys

- The *Genesys 8.0 Resource Capacity Planning Guide,* which explains how the Genesys model has been expanded to serve agents conducting contact center interactions across several media types.
- The *Genesys 8.0 Security Deployment Guide,* which will help you install the Genesys Security Pack and manage security certificates for the hosts in your contact center.
- The *Reporting Technical Reference Guide for the Genesys X Release,* which describes the stat type definitions provided by Genesys solutions.
- *Genesys Technical Publications Glossary,* which ships on the Genesys Documentation Library DVD and which provides a comprehensive list of the Genesys and computer-telephony integration (CTI) terminology and acronyms used in this document.

- *Genesys Migration Guide*, which ships on the Genesys Documentation Library DVD, and which provides documented migration strategies for Genesys product releases. Contact Genesys Technical Support for more information.
- Release Notes and Product Advisories for this product, which are available on the Genesys Technical Support website at http://genesyslab.com/support.

Information about supported hardware and third-party software is available on the Genesys Technical Support website in the following documents:

- *Genesys Supported Operating Environment Reference Manual*
- *Genesys Supported Media Interfaces Reference Manual*

Consult these additional resources as necessary:

- *Genesys 7 Interoperability Guide*, which provides information on the compatibility of Genesys products with various Configuration Layer Environments; Interoperability of Reporting Templates and Solutions; and Gplus Adapters Interoperability.
- *Genesys Licensing Guide*, which introduces you to the concepts, terminology, and procedures relevant to the Genesys licensing system.

For additional system-wide planning tools and information, see the release-specific listings of System Level Documents on the Genesys Technical Support website, accessible from the system level documents by release tab in the Knowledge Base Browse Documents Section.

Genesys product documentation is available on the:

- Genesys Technical Support website at http://genesyslab.com/support.
- Genesys Documentation Library DVD, which you can order by e-mail from Genesys Order Management at orderman@genesyslab.com.

Document Conventions

This document uses certain stylistic and typographical conventions—introduced here—that serve as shorthands for particular kinds of information.

Document Version Number

A version number appears at the bottom of the inside front cover of this document. Version numbers change as new information is added to this document. Here is a sample version number:

80fr_dep_statserver_11-2009_v8.0.001.00

You will need this number when you are talking with Genesys Technical Support about this product.

Screen Captures Used in This Document

Screen captures from the product graphical user interface (GUI), as used in this document, may sometimes contain minor spelling, capitalization, or grammatical errors. The text accompanying and explaining the screen captures corrects such errors *except* when such a correction would prevent you from installing, configuring, or successfully using the product. For example, if the name of an option contains a usage error, the name would be presented exactly as it appears in the product GUI; the error would not be corrected in any accompanying text.

Type Styles

Table 21 describes and illustrates the type conventions that are used in this document.

Table 21: Type Styles

Type Style	Used For	Examples
Italic	• Document titles • Emphasis • Definitions of (or first references to) unfamiliar terms • Mathematical variables Also used to indicate placeholder text within code samples or commands, in the special case where angle brackets are a required part of the syntax (see the note about angle brackets on page 122).	Please consult the *Genesys Migration Guide* for more information. Do *not* use this value for this option. A *customary and usual* practice is one that is widely accepted and used within a particular industry or profession. The formula, $x + 1 = 7$ where *x* stands for . . .
Monospace font (Looks like `teletype` or `typewriter text`)	All programming identifiers and GUI elements. This convention includes: • The *names* of directories, files, folders, configuration objects, paths, scripts, dialog boxes, options, fields, text and list boxes, operational modes, all buttons (including radio buttons), check boxes, commands, tabs, CTI events, and error messages. • The values of options. • Logical arguments and command syntax. • Code samples. Also used for any text that users must manually enter during a configuration or installation procedure, or on a command line.	Select the `Show variables on screen` check box. In the `Operand` text box, enter your formula. Click `OK` to exit the `Properties` dialog box. T-Server distributes the error messages in `EventError` events. If you select `true` for the `inbound-bsns-calls` option, all established inbound calls on a local agent are considered business calls. Enter `exit` on the command line.
Square brackets ([ ])	A particular parameter or value that is optional within a logical argument, a command, or some programming syntax. That is, the presence of the parameter or value is not required to resolve the argument, command, or block of code. The user decides whether to include this optional information.	`smcp_server -host [/flags]`
Angle brackets (< >)	A placeholder for a value that the user must specify. This might be a DN or a port number specific to your enterprise. **Note:** In some cases, angle brackets are required characters in code syntax (for example, in XML schemas). In these cases, italic text is used for placeholder values.	`smcp_server -host <confighost>`

Index

Symbols

A

B

C

D

E

F

G

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X