Reporting 8.0

CCPulse+

Administrator's Guide

The information contained herein is proprietary and confidential and cannot be disclosed or duplicated without the prior written consent of Genesys Telecommunications Laboratories, Inc.

Copyright © 1999–2014 Genesys Telecommunications Laboratories, Inc. All rights reserved.

About Genesys

Genesys is the world's leading provider of customer service and contact center software—with more than 4,000 customers in 80 countries. Drawing on its more than 20 years of customer service innovation and experience, Genesys is uniquely positioned to help companies bring their people, insights and customer channels together to effectively drive today's customer conversation. Genesys software directs more than 100 million interactions every day, maximizing the value of customer engagement and differentiating the experience by driving personalization and multi-channel customer service—and extending customer service across the enterprise to optimize processes and the performance of customer-facing employees. Go to www.genesys.com for more information.

Each product has its own documentation for online viewing at the Genesys Customer Care 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 and the Genesys logo are registered trademarks of Genesys Telecommunications Laboratories, Inc. All other company names and logos may be trademarks or registered trademarks of their respective holders.

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.

Customer Care from Genesys

If you have purchased support directly from Genesys, please contact Genesys Customer Care. Before contacting Customer Care, please refer to the ***Genesys Care Program Guide*** for complete contact information and procedures.

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.genesys.com

Document Version: 80rt_ad_ccpulse+_02-2014_v8.0.202.00

Table of Contents

Preface **7**

About CCPulse+ 7
Intended Audience 8
Making Comments on This Document 8
Contacting Genesys Customer Care 8
Document Change History 9
New in Document Version 8.0.202.00 9
New in Document Version 8.0.201.00 9
New in Document Version 8.0.101.00 9
New in Document Version 8.0.001.00 11

Chapter 1 **CCPulse+ Overview 13**

Overview 13
New in This Release 14
New in Release 8.0.2 14
New in Release 8.0.1 14
New in Release 8.0.0 16

Chapter 2 **Fine-Tuning CCPulse+ Configuration 17**

common and security Sections 18
Transport Layer Security Support 19
Client-Side Port Definition 19
CustomStatistic Section 21
HistoricalView Section 22
ObjectFormat Section 23
Substitution Rules 26
Storage Section 29
UserInterface Section 33
View Section 36
Workspace Section 38
Changing the Display of Dates and Time 43
CCPulse+ Registry Settings 44

Chapter 3 **Historical Reporting Considerations for CCPulse+ 49**

Data Mart Connections 49
Composite Metrics 50
Column Names 50
The Last N Intervals Parameter 51
Historical Associations for Statistics 51
Propagation of Template Changes 51

Chapter 4 **Server Connections to CCPulse+ 53**

Configuration Server Connection Failure 54
Stat Server Connection Failure 56
DB Server Connection Failure 58

Chapter 5 **Using Custom Statistics 59**

Identifying Custom Statistics 59
Hidden Statistics 60
Using JScript Language 60
Referencing Basic Statistics 60
Using Predefined Objects in Formulas 62
Formatting the Results 68
Displaying Capacity Information 68

Chapter 6 **Thresholds and Actions 71**

Overview 71
Predefined Objects 72
Predefined Thresholds 77
How to Use Thresholds 77
Explanations of the Predefined Thresholds 79
Predefined Actions 84
How Actions Function 84
Explanations of the Predefined Actions 85

Chapter 7 **Measuring CCPulse+/Stat Server Traffic 91**

Determining Factors 91
The Number of Requests 91
The Number of Responses 92
The Size of One Request/Response 92

Chapter 8 **Troubleshooting** **93**

Agents Having Trouble Logging In 93
Wizards Are Disabled 94
Import/Export Utility Is Disabled 94
No DB Server Connection Error Message 95
Agents Missing from Call Center Objects Pane 95
General SQL Server Error Message 96
Agents Cannot Monitor Agent Activity 96
Sudden Stat Server Disconnection 96
Stat Server Missing from New Workspace 97
No Historical Associations Error Messages 97
Cannot Recover Workspace 97
Can't View Multi-Tenant Data 98

Supplements **Related Documentation Resources** **99**

Document Conventions **101**

Index **103**

Preface

Welcome to the *Reporting 8.0 CCPulse+ Administrator's Guide*. This document presents information that CCPulse+ administrators need in order to fine-tune and troubleshoot CCPulse+ installation. This guide is valid only for the 8.0.*x* release of this product.

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

This preface contains the following sections:

- About CCPulse+, page 7
- Intended Audience, page 8
- Making Comments on This Document, page 8
- Contacting Genesys Customer Care, page 8
- Document Change History, page 9

In brief, you will find the following information in this guide:

- Special considerations concerning CCPulse+ historical reporting
- Instructions for how to customize the user interface
- Instructions for how to create and format custom statistics
- Explanations of the threshold and action scripts
- A list of the historical statistics that are associated with the CCPulse+ real-time statistics
- Troubleshooting suggestions

About CCPulse+

CCPulse+ enables both real-time and historical monitoring of contact center activity. CCPulse+ monitors the status and statistics related to contact center objects (such as agents, queues, Routing Points, and interaction queues) and displays them on the user's desktop.

Intended Audience

This guide is intended primarily for CCPulse+ administrators. It assumes that you have a basic understanding of:

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

You should also be familiar with Genesys Framework architecture and functions, particularly Stat Server statistics and configuration.

CCPulse+ administrators have access to CCPulse+ functions that are not available to other users. CCPulse+ administrators can:

- Operate the Threshold and Action wizards.
- Operate the Template Wizard to create, modify, or delete CCPulse+ templates.
- Use the Import/Export Utility to copy CCPulse+ templates, thresholds, and actions from one storage to another.

A strong knowledge of VBScript might be required for some functionality, such as creating custom formulas.

Making Comments on This Document

If you especially like or dislike anything about this document, please feel free to e-mail your comments to Techpubs.webadmin@genesys.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 information in this document only and to the way in which the information is presented. Speak to Genesys Customer Care 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 Customer Care

If you have purchased support directly from Genesys, please contact Genesys Customer Care.

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

Document Change History

This section lists topics that are new in the current release of *this document*, or that have changed significantly since the preceding release.

New in Document Version 8.0.202.00

The document has been further updated to support CCPulse+ release 8.0.2. The following topics have been added or have changed significantly since the previous 8.0 release of this document:

- A chapter containing outdated instructions for silent installation ("Installing CCPulse+ Silently") has been removed. The outdated method used a `setup.iss` response file. The currently supported method, which uses the `genesys_silent.ini` file in the CCPulse+ installation package (IP), is standard Genesys functionality.

New in Document Version 8.0.201.00

The document has been updated to support CCPulse+ release 8.0.2. The following topics have been added or have changed significantly since the previous 8.0 release of this document:

- "Example: Formatting an Agent" on page 26—Updated example to correctly use `EmployeeID`.
- "CCPulse+ Registry Settings" on page 44—Added GridSortingInterval registry setting.
- "Server Connections to CCPulse+" on page 53—Updated the reconnection details.
- "How to Use Thresholds" on page 77—Updated the actions CCPulse+ performs when a statistic value changes.

New in Document Version 8.0.101.00

The document has been updated to support CCPulse+ release 8.0.1. The following topics have been added or have changed significantly since the previous 8.0 release of this document:

- "New in Release 8.0.1" on page 14—A new subsection in the "New in This Release" section that describes the feature enhancements that are introduced in release 8.0.100.
- "common and security Sections" on page 18—New sections that hold configuration options that are not unique to CCPulse+, but do enable new CCPulse+ functionality—specifically:
  - `enable-ipv6`
  - `inactivity-timeout`

- "Transport Layer Security Support" on page 19—A new subsection that describes Genesys Transport Layer Security (TLS), a new feature that allows for secure data exchange among components.
- "Client-Side Port Definition" on page 19—A new subsection that describes a new feature that enables you to control how CCPulse+ connects to various servers and how to configure client-side port definition using the `Network.ini` file.
- "HistoricalView Section" on page 22—A new section that holds configuration options that affect the presentation of data in CCPulse+ historical views—specifically:
  - `FirstDayOfWeek`
  - `MinDayNumberInFirstWeek`
  - `NoAggregationInterval`
  - `TimeZone`
  - `WholeWeeks`
- "ObjectFormat Section" on page 23—New functionality that allows you to define how CCPulse+ objects are displayed in the workspace, in all views, and in the Extended Current Status pane. You have the ability to format each CCPulse+ object by using a format string.
- "DisableAgentCurrentState" on page 39—Updated with the ability to prevent the creation and display of other agent statistics.
- "DisableThresholdsDlg" on page 40—Added option to the `Workspace` section that was added in the Reporting 8.0 release, but was previously not included in *this document*.
- "CCPulse+ Registry Settings" on page 44—Updated to indicate that the newly added "AgentStats" configuration setting on page 45 is the *only* setting that can be edited by using the Microsoft Windows Registry Editor. Three configuration settings have moved to the `Settings` menu option in CCPulse+ and can no longer be edited by using the Microsoft Windows Registry Editor. These configuration settings are the following:
  - `GraphTitleFontSize`
  - `GraphBkColor`
  - `GraphAxisFontSize`
- "The Last N Intervals Parameter" on page 51—Updated to correctly describe the behavior of `Last N Intervals` in CCPulse+, previously it was described incorrectly in *this document.*
- "Identifying Custom Statistics" on page 59—Updated to included new hidden statistic icons that have been added.
- "Hidden Statistics" on page 60—A new subsection that describes new functionality that allows you to hide both basic and custom statistics by using the Templates Wizard.
- Figure 9 on page 61—Updated to include new hidden statistic icons.

New in Document Version 8.0.001.00

The document has been updated to support CCPulse+ release 8.0. The following topics have been added or have changed significantly since the previous 7.6 release of this document:

- "New in Release 8.0.0" on page 16—A new subsection in the "New in This Release" section that describes the feature enhancements that are introduced in release 8.0.0.
- "Storage Section" on page 29—New configuration options have been added in the `Storage` section to store objects in binary or XML format—specifically:
  - `DefaultStorageType`
  - `BinActionsStorageFullPath`
  - `BinTemplatesStorageFullPath`
  - `BinThresholdsStorageFullPath`
  - `BinWorkspacesStorageFullPath`
  - `XMLActionsStorageFullPath`
  - `XMLTemplatesStorageFullPath`
  - `XMLThresholdsStorageFullPath`
  - `XMLWorkspacesStorageFullPath`
- "UserInterface Section" on page 33—A new `EnableXMLExport` configuration option was added to the `UserInterface` section.
- "Configuration Server Connection Failure" on page 54—An improved reconnection procedure has been added, when the CCPulse+ connection to Configuration Server or Stat Server fails.
- Support for CCPulse+ reporting of Genesys Info Mart queries has been discontinued in the 8.*x* release. The previously documented `QueryStorageFull Path` configuration option has been removed from Chapter 2.

Chapter

1 CCPulse+ Overview

This chapter introduces you to CCPulse+, which is a graphical user interface (GUI) component of Genesys Solution Reporting.

This chapter includes the following sections:

- Overview, page 13
- New in This Release, page 14

Overview

CCPulse+ enables both real-time and historical monitoring of contact center activity. CCPulse+ monitors the status and statistics that are related to contact center objects (such as agents, queues, Routing Points, and interaction queues) and displays them on the user's desktop. Using CCPulse+ you can:

- Customize real-time, historical, and query-based views of contact center objects.
- Monitor the current state and activity of objects throughout the contact center, to help make decisions about staffing, scheduling, and call-routing strategies.
- Create threshold/action associations that generate some kind of notification when a specified activity reaches the threshold level you set.
- Print views, which can then function as basic reports of contact center activity.

CCPulse+ interfaces with the Services and Configuration Layers of the Genesys Framework to monitor agent performance of one or more tenants from one or more sites.

If you are using Historical Reporting, CCPulse+ also connects to the Reporting Data Mart Services.

Figure 1 illustrates the architecture delivering these capabilities. For detailed information on CCPulse+ interface, refer to the *Reporting 8.0 CCPulse+ Help*. For information about overall Solution Reporting architecture, refer to the *Reporting 8.0 Deployment Guide*.

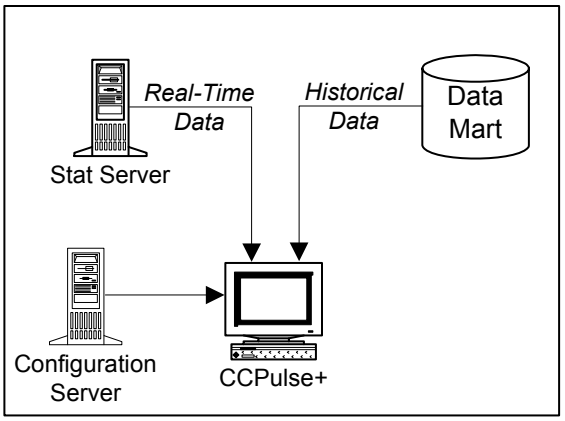

Figure 1: CCPulse+ Architecture

New in This Release

This section describes new or changed functionality that was introduced in CCPulse+ 8.0.x releases:

- New in Release 8.0.2, page 14
- New in Release 8.0.1, page 14
- New in Release 8.0.0, page 16

New in Release 8.0.2

Starting with CCPulse+ release 8.0.2, CCPulse+ provides the following new or changed functionality:

- Updated CCPulse+ to release 8.0.2.

New in Release 8.0.1

Starting with CCPulse+ release 8.0.1, CCPulse+ provides the following new or changed functionality:

- The `HistoricalView` section on page 22 affects the presentation of data in CCPulse+ historical views. This section contains the following new configuration options:

> **Note:** The `HistoricalView` section and its configuration options are previously undocumented in the 8.0 release.

  - FirstDayOfWeek, on page 22
  - WholeWeeks, on page 23
  - MinDayNumberInFirstWeek, on page 23
  - TimeZone, on page 23
  - NoAggregationInterval, on page 23
- The common and security sections on page 18 hold configuration options that are not unique to CCPulse+, but do enable CCPulse+ functionality. This section contains the following new configuration options:
  - enable-ipv6, on page 18 which enables IPv6 for all IP connections for CCPulse+.
  - inactivity-timeout, on page 18 which specifies the amount of time in which a user can be inactive before the inactivity timeout occurs and the user is locked out of the CCPulse+ session. The user must then log back in to continue with the session.
- CCPulse+ now supports Genesys Transport Layer Security (TLS) on page 19, which is secure data exchange among components. TLS is an industry-standard protocol for secure communications over the Internet.
- CCPulse+ now supports client-side port definition on page 19, which enables you to control how CCPulse+ connects to various servers and how to configure client-side port definition with the Network.ini file.
- The ObjectFormat section on page 23 describes a new formatting section and the options that use a format string to define how CCPulse+ objects are displayed in the workspace, in all views, and in the Extended Current Status pane.
- The DisableAgentCurrentState configuration option on page 39 has been updated to allow CCPulse+ administrators to prevent CCPulse+ users from creating and displaying other agent statistics. The AgentStats registry setting on page 45 also enables this functionality, but is configured on an individual basis.
- The DisableThresholdsDlg configuration option on page 40 was added to the Workspace section. This option was added in the Reporting 8.0 release, but previously was not included in *this document*.
- The GraphTitleFontSize, GraphBkColor, and GraphAxisFontSize CCPulse+ configuration settings on page 44, have been removed from the registry and moved to the Settings menu option under the Tools menu of the CCPulse+ interface. Previously, these configuration settings could be edited by using *only* the Microsoft Windows Registry Editor. Now they can be edited by using *only* the Settings menu option under the Tools menu of CCPulse+. For more information about how to configure these settings by using menu options, refer to the *Reporting 8.0 CCPulse+ Help.*

The "Hidden statistics" subsection on page 60 discusses the ability to hide both basic and custom statistics. The new hidden basic and custom statistics icons are shown on page 59. For more information about how to hide statistics in CCPulse+, refer to the *Reporting 8.0 CCPulse+ Help*.

New in Release 8.0.0

This subsection describes new or changed functionality that was introduced in the initial 8.0 release of CCPulse+:

- New configuration options in the `Storage` section to store objects in binary or XML format, described beginning on page 30:
  - `DefaultStorageType`
  - `BinActionsStorageFullPath`
  - `BinTemplatesStorageFullPath`
  - `BinThresholdsStorageFullPath`
  - `BinWorkspacesStorageFullPath`
  - `XMLActionsStorageFullPath`
  - `XMLTemplatesStorageFullPath`
  - `XMLThresholdsStorageFullPath`
  - `XMLWorkspacesStorageFullPath`
- New configuration options in the `UserInterface` section:
  - `EnableXMLExport`, on page 33—A previously undocumented option from the 7.0.1 release
- An improved reconnection procedure, when the CCPulse+ connection to Configuration Server or Stat Server fails, beginning on page 54.

Support for CCPulse+ reporting of Genesys Info Mart queries has been discontinued in the 8.*x* release. The previously documented `QueryStorageFullPath` configuration option has been removed from Chapter 2.

Other features that are introduced in the 8.0 release, such as dynamic column sizing and sorting, the display of capacity information, horizontal and vertical tabbing, and the security banner that is displayed when the application is invoked are described in either *Reporting 8.0 CCPulse+ Help* or the *Reporting 8.0 Deployment Guide.*

Genesys

Chapter

2 Fine-Tuning CCPulse+ Configuration

After the initial configuration of the CCPulse+ `Application` object and installation, you can use Configuration Manager to update the CCPulse+ application. This chapter describes how to fine-tune the CCPulse+ configuration. It contains the following sections:

- common and security Sections, page 18
- CustomStatistic Section, page 21
- HistoricalView Section, page 22
- ObjectFormat Section, page 23
- Storage Section, page 29
- UserInterface Section, page 33
- View Section, page 36
- Workspace Section, page 38
- Changing the Display of Dates and Time, page 43
- CCPulse+ Registry Settings, page 44

The CCPulse+ application itself does not consider the case of configuration information it reads from Configuration Server. However, the RDBMS of your Configuration Server database might be case sensitive. As you fine-tune your CCPulse+ `Application` object, keep this in mind. Also, for those configuration options for which you specify `true/false` values, any of the following additional values is also valid:

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

common and security Sections

The `common` and `security` sections hold configuration options that are not unique to CCPulse+, but that do enable CCPulse+ functionality, do require your configuration, and are important for you to know. (See Table 1.) Two security features that are provided in the Genesys software and supported by CCPulse+—"Transport Layer Security Support" on page 19 and "Client-Side Port Definition" on page 19—are also described.

Table 1: Configuration Options for the Common and Security Sections

Section	Option	Description
common	enable-ipv6	Use this option to enable IPv6 for all IP connections for CCPulse+. This option is configured in the `common` section of the CCPulse+ `Application` object by using either Genesys Administrator or Configuration Manager. Setting this option to `1`, specifies that CCPulse+ supports IPv6. It is set to `0` by default, to ensure backward compatibility. For more information about how to configure IPv6, refer to the *Framework 8.1 Deployment Guide*. Default Value: `0` Valid Values: `0`, `1` Change Takes Effect: Upon restart
security	inactivity-timeout	Use this option to specify the amount of time (in minutes) in which a user can be inactive—that is, not interact with the system in any way—without any impact on the user's CCPulse+ session. After the timeout expires, the user is locked out of the CCPulse+ session. The user must log back in to continue with the session. This option is configured in the `security` section of the CCPulse+ `Application` object by using either Genesys Administrator or Configuration Manager. The default value of `0` means that the feature is disabled. For more information about how to configure inactivity timeout, refer to the *Genesys 8.0 Security Deployment Guide*. For more information about how inactivity timeout functions in CCPulse+, refer to the *Reporting 8.0 CCPulse+ Help.* Default Value: `0` Valid Value: Any nonnegative integer Change Takes Effect: Upon restart

Transport Layer Security Support

CCPulse+ supports secure data exchange among other Genesys components through the Transport Layer Security (TLS) protocol, which is an industry standard for secure communications over the Internet.

All Genesys components are configured in Configuration Manager. To enable secure data exchange among the components, you must configure additional parameters in the `Host` objects, as well as in the `Application` objects that represent these components.

For detailed instructions about how to add the TLS feature to existing connections of your Genesys configuration, including CCPulse+, see the *Genesys 8.0 Security Deployment Guide.*

Client-Side Port Definition

The client-side port definition feature enables you to control how CCPulse+ connects to various servers. This includes the ability to control the initial connection to a Configuration Server. Specifying client-side port is required when you want CCPulse+ to establish connections with servers behind a firewall that is configured to limit incoming connections to a well-defined set of client ports due to enhanced security considerations.

CCPulse+ configures client-side port definition using the `Network.ini` file. This file resides in the CCPulse+ installation directory. The contents of this file are loaded during CCPulse+ startup, and are not reloaded. If the `Network.ini` file is modified, you must restart CCPulse+ to allow the changes to take effect. If the file does not exist, then client-side port definition will not be available.

When CCPulse+ is about to connect to a server, CCPulse+ looks up the client-side parameters in the `Network.ini` file. If the appropriate information is found, CCPulse+ uses this in an attempt to connect. Otherwise, CCPulse+ attempts the connection as usual with the parameter selected automatically by Microsoft Windows.

Network.ini File

This section explains the details about the format of the `Network.ini` file. The file format is similar to the standard Microsoft Windows `INI` file, except that it does not have sections, instead it has `key-value` pairs.

Each `key-value` pair in the `Network.ini` file should be formatted as follows:

```
target-host:target-port=[local-host]:local-port
```

- `target-host`—the name of the target host to which CCPulse+ connects. Specify this name the same way that it is specified in the Host object in Configuration Manager and in the `Host` parameter in the login dialog. This parameter is case-insensitive. For example, `myhost.mydomain.com`, `MYhOst.MyDoMaiN.com`, and `MYHOST.MYDOMAIN.COM` all refer to the same host.

  Note: You must specify the `target-host` name the same way that it is specified in Configuration Server.

  For example, if the full host name is:

  - `myserever.mydomain.com`

  But in Configuration Server it uses the short name:

  - `myserver`

  In the `Network.ini` file you also must also use the short name:

  - `myserver`

  In the above example, if the `Network.ini` file contains the full host name (`myserever.mydomain.com`), then client-side port definition will not work.

- `target-port`—the port of the target server to which CCPulse+ connects. This must be a valid TCP port number. Zero and negative values are not allowed.
- `local-host`—the local host address that will be used to establish the connection with a specified target host and port.

  This is optional and only needs to be specified if the client machine has multiple network addresses. Skipping this step results in Microsoft Windows automatically selecting this parameter. In most cases (including VPN connections) this is acceptable.
- `local-port`—the local port used to establish the connection with a specified target host and port.

  This must be a valid TCP port number. Negative values are not allowed. A value of zero is allowed, but results in Microsoft Windows automatically selecting the port.

Note: CCPulse+ maintains two connections with Configuration Server, which means you must configure two client-side ports for this server. This is done by creating two separate entries in the `Network.ini` file that contain the same remote host and port, but a different local port, for example:

```
myserver.mydomain.com:7000=myworkstation.mydomain.com:59000
myserver.mydomain.com:7000=myworkstation.mydomain.com:59001
```

CustomStatistic Section

The `CustomStatistic` section holds two configuration options that affect the display of custom statistics within the CCPulse+ interface. (See Table 2.)

Table 2: Configuration Option for the CustomStatistic Section

Option	Description

ExtendedCurrentStatus	Use this option to configure CCPulse+ to display reasons in the current state statistics that are provided through agent views. If this option is set to `true`, CCPulse+ displays the reasons for the current state statistic as one or more strings that are separated by semicolons. If it is set to `false`, the reasons for the current state statistics are not displayed.
	To implement this feature fully, you must also define the underlying stat type for agent current state statistics, as follows:
	`Category=CurrentTargetState`
	`MainMask=*`
	`Objects=Agent`
	`Subject=AgentStatus`
	This `DNAction`-based stat type enables Stat Server to provide data along with duration and actual status data.
	In Figure 2, the `AgentStateReason` view displays a reason of `1` for the agent `Employee_ID_601` who is in the state `NotReadyForNextCall`. In this example, a threshold and action have also been applied to this statistic. For information on setting thresholds and actions, refer to the discussion starting on page 71. Table 18 on page 72 shows how to use reasons in thresholds.
	Default Value: `false`
	Valid Values: `true, false`
	Change Takes Effect: Upon restart

MediaTypes	This option populates the drop-down list of media types that are available in the `Media Filter` section of the `Properties` dialog box for the `CurrentTargetState` statistic category. Default Value: `voice` Valid Value: <comma-separated list of media types> (Example: `"voice, email, voip"`) Change Takes Effect: Upon restart

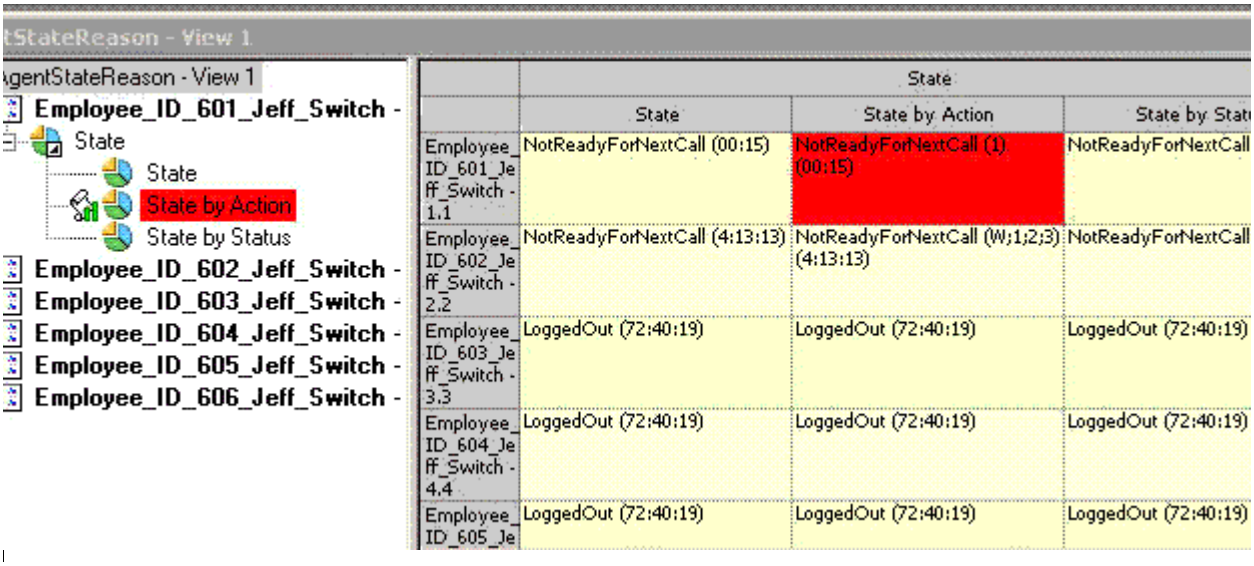

Figure 2: CCPulse+ Displays Reason when Option Is Set to True

HistoricalView Section

The `HistoricalView` section holds configuration options that affect the presentation of data in CCPulse+ historical views. The options in this section were designed after the DataMart ETL Service configuration options; thus, many cases, you can check your Genesys ETL Service configuration to determine the appropriate value for the corresponding CCPulse+ option. (See Table 3.)

Table 3: Configuration Options for the HistoricalView Section

Option	Description
FirstDayOfWeek	Use this option to specify the first day of the week for reports that have weekly intervals. Different counties begin the week on different days (for example, the first day of the week in the USA is Sunday, while the first day of the week in the Russia is Monday). If CCPulse+ is configured for historical reporting with weekly aggregation, please check your Genesys ETL Service Configuration to determine the appropriate value for this option. **Note:** Standard ISO-8601 assumes that the first day of the week is Monday. Default Value: `Sunday` Valid Values: `Sunday-Saturday` Change Takes Effect: Upon restart

Table 3: Configuration Options for the HistoricalView Section (Continued)

Option	Description
MinDayNumberInFirstWeek	Use this option to specify how many days in the same week of the first week of the year should be included at a minimum. By default one day is included, which conforms to the United States and Russian week numbering rules. Please check your Genesys ETL Service Configuration to determine the appropriate value for this option. To emulate ISO-8601, configure this option to `4`. Default Value: `1` Valid Values: `1-7` Change Takes Effect: Upon restart **Note:** This option is effective only when `WholeWeeks` is configured to `Yes`.
NoAggregationInterval	This option specifies the time-interval resolution, in minutes, for nonaggregated data in your DataMart database. Configure this option *only* if you use a nonstandard data collection interval other than the default of 15 minutes. Default Value: `15` Valid Values: `1-1440` Change Takes Effect: Upon restart
TimeZone	This option enables daylight savings time for the specified time zone. Default Value: `default` Valid Values: `GMT+HH, GMT+HHDST, GMT+HH:MM, GMT-HH:MM, GMT+HH:MMDST, GMT-HH:MMDST`. The `default` value represents the default system settings from your current location, where `HH` is `Hours` (`0-11`), `MM` is `Minutes` (`0-59`), and `DST` will add `Daylight Savings Time`. Change Takes Effect: Upon restart
WholeWeeks	Use this option to specify whether CCPulse+ recognizes that DataMart is using whole weeks. Please check your Genesys ETL Service Configuration to determine the appropriate value for this option. Default Value: `no` Valid Values: `yes, no` Change Takes Effect: Upon restart

ObjectFormat Section

CCPulse+ release 8.0.1 introduces functionality that allows you to define how CCPulse+ objects are displayed in the workspace, in all views, and in the Extended Current Status pane. The `ObjectFormat` section holds configuration

options that affect the display format of each CCPulse+ object type. The name of each option in the `ObjectFormat` section represents the available CCPulse+ object types and the corresponding description defines what can be formatted by using the format string. (See Table 4.)

Table 4: Configuration Options for the ObjectFormat Section

Option	Description
ACDPosition	This option defines the format of the ACD Position object.
Agent	This option defines the format of the Agent object.
AgentGroup	This option defines the format of the Agent Group object.
CallingList	This option defines the format of the Calling List object.
Campaign	This option defines the format of the Campaign object.
CampaignAgentGroup	This option defines the format of the Campaign Agent Group object.
CampaignCallingList	This option defines the format of the Campaign Calling List object.
CampaignPlaceGroup	This option defines the format of the Campaign Place Group object.
Chat	This option defines the format of the Chat object.
CoBrowse	This option defines the format of the CoBrowsing object.
Dialer	This option defines the format of the S-Dialer object.
DN	This option defines the format of the DN object.
EMail	This option defines the format of the E-mail object.
Extension	This option defines the format of the Extension object.
ExternalRoutingPoint	This option defines the format of the External Routing Point object.
ExternalRoutingPointInTheGroup	This option defines the format of the External Routing Point object shown inside a group.
Fax	This option defines the format of the Fax object.
InteractionQueue	This option defines the format of the Interaction Queue object.
Place	This option defines the format of the Place object.
PlaceGroup	This option defines the format of the Place Group object.
Queue	This option defines the format of the Queue object.
QueueGroup	This option defines the format of the Queue Group object.

Table 4: Configuration Options for the ObjectFormat Section (Continued)

Option	Description
QueueInTheGroup	This option defines the format of the Queue object shown inside a group.
Reason	This option defines the format of the Reason object.
RoutingPoint	This option defines the format of the Routing Point object.
RoutingPointGroup	This option defines the format of the Routing Point Group object.
RoutingPointInTheGroup	This option defines the format of the Routing Point object shown inside a group.
RoutingQueue	This option defines the format of the Routing Queue object.
RoutingQueueInTheGroup	This option defines the format of the Routing Queue object shown inside a group.
Script	This option defines the format of the Script object.
ServiceNumber	This option defines the format of the Service Number object.
ServiceNumberGroup	This option defines the format of the Service Number Group object.
ServiceNumberInTheGroup	This option defines the format of the Service Number object shown inside a group.
SimpleRouting	This option defines the format of the Simple Routing object.
Skill	This option defines the format of the Skill object.
Switch	This option defines the format of the Switch object.
Tenant	This option defines the format of the Tenant object.
Video	This option defines the format of the Video object.
VirtualQueue	This option defines the format of the Virtual Queue object.
VirtualQueueInTheGroup	This option defines the format of the Virtual Queue object shown inside a group.
VirtualRoutingPoint	This option defines the format of the Virtual Routing Point object.
VirtualRoutingPointInTheGroup	This option defines the format of the Virtual Routing Point object shown inside a group.
VoiceTreatmentPort	This option defines the format of the Voice Treatment Port object.
VoIP	This option defines the format of the VoIP object.

You have the ability to format each CCPulse+ object by using a format string that consists of two parts:

a. The field types, and their widths and formats on the screen.

Notes:
- This part of the format string is compatible with the C-language `printf` function format argument. For more information, refer to http://www.cplusplus.com/reference/clibrary/cstdio/printf/.
- The ";" (semicolon) symbol is allowed to occur *only* at the *END* of the first part of the format string, and not at any other location in this part of the string. The semicolon symbol is used to indicate the end of the first part of the format string.

b. The properties of the corresponding CCPulse+ object

Example: Formatting an Agent

For `EmployeeID=Employee`, `FirstName=Johnathon`, and `LastName=Smith`:

- `%.5s, %.5s %.5s`

  produces the following result:

  `Emplo, John Smith`

- `%.6s, %.7s %.5s`

  produces the following result:

  `Employ, Johnath Smith`

If the `ObjectFormat` section is defined, but not all object's format strings are configured, CCPulse+ uses substitution rules. (See Table 5.) If the `ObjectFormat` section is not defined, CCPulse+ defaults to legacy formatting rules.

Substitution Rules

When a format string for an object type is not defined, CCPulse+ can substitute the format string for this object with a more general object type. For example, `Extension` can be substituted with the format string for `DN`. Otherwise, it defaults to legacy formatting rules. Table 5 lists the CCPulse+ format string substitutions.

Table 5: CCPulse+ Format String Substitutions

Object Type	Substitution
ACD Position	DN
Agent	No substitution. It displays as you format it by using the format string or defaults to legacy formatting rules.

Table 5: CCPulse+ Format String Substitutions (Continued)

Object Type	Substitution
Agent Group	No substitution. It displays as you format it by using the format string or defaults to legacy formatting rules.
Calling List	No substitution. It displays as you format it by using the format string or defaults to legacy formatting rules
Campaign	No substitution. It displays as you format it by using the format string or defaults to legacy formatting rules
Campaign Agent Group	Agent Group
Campaign Calling List	Calling List
Campaign Place Group	Place Group
Chat	DN
CoBrowsing	DN
DN	No substitution. It displays as you format it by using the format string or defaults to legacy formatting rules
E-Mail	DN
Extension	DN
External Routing Point	DN
External Routing Point in the Group	External Routing Point
Fax	DN
Interaction Queue	Script
Place	No substitution. It displays as you format it by using the format string or defaults to legacy formatting rules
Place Group	No substitution. It displays as you format it by using the format string or defaults to legacy formatting rules.
Queue	No substitution. It displays as you format it by using the format string or defaults to legacy formatting rules
Queue Group	No substitution. It displays as you format it by using the format string or defaults to legacy formatting rules
Queue in the Group	Queue

Table 5: CCPulse+ Format String Substitutions (Continued)

Object Type	Substitution
Reason	No substitution. It displays as you format it by using the format string or defaults to legacy formatting rules
Routing Point	No substitution. It displays as you format it by using the format string or defaults to legacy formatting rules
Routing Point Group	No substitution. It displays as you format it by using the format string or defaults to legacy formatting rules
Routing Point in the Group	Routing Point
Routing Queue	DN
Routing Queue in the Group	Routing Queue
S-Dialer	DN
Script	No substitution. It displays as you format it by using the format string or defaults to legacy formatting rules
Service Number	No substitution. It displays as you format it by using the format string or defaults to legacy formatting rules
Service Number Group	No substitution. It displays as you format it by using the format string or defaults to legacy formatting rules
Service Number in the Group	Service Number
Simple Routing	Script
Skill	No substitution. It displays as you format it by using the format string or defaults to legacy formatting rules
Switch	No substitution. It displays as you format it by using the format string or defaults to legacy formatting rules
Tenant	No substitution. It displays as you format it by using the format string or defaults to legacy formatting rules.
Video	DN
Virtual Queue	No substitution. It displays as you format it by using the format string or defaults to legacy formatting rules
Virtual Queue in the Group	Virtual Queue

Table 5: CCPulse+ Format String Substitutions (Continued)

Object Type	Substitution
Virtual Routing Point	No substitution. It displays as you format it by using the format string or defaults to legacy formatting rules
Virtual Routing Point in the Group	Virtual Routing Point
Voice Treatment Port	DN
VoIP	DN

Storage Section

A *storage* is a file in which information of a specific type resides. CCPulse+ stores the following in separate storage files:

- Actions
- Templates
- Thresholds
- Workspaces

The Real-Time Reporting Wizard prompts you for values for these storages. As an alternate method, you can use Configuration Manager to configure storage locations in the `Storage` section of your CCPulse+ `Application` object. This is accomplished at the `Options` tab.

Beginning with the 8.0 release, you can specify the format of CCPulse+ storages as one of the following:

- `CCPulse+ 7.x Storage`, in `Legacy` (OLE storage format) which is provided for backward compatibility
- `CCPulse+ Binary Data Files`, in binary format, and `CCPulse+ XML Data Files`, in XML format

To enter a storage location within Configuration Manager:

1. Double-click the desired item in the `Name` column to open the `Edit Option` window.
2. In the `Value` text box, enter the full path and file name of the location in which the storage file should reside.
3. Click `OK`. The `Storage` list now displays the updated storage location.

Notes: The templates and works options must define valid and existing storages in order to start the CCPulse+ application.

If a storage location is to be shared by multiple users, ensure that all have access to it.

Table 6: Configuration Options for the Storage Section

Type	Option	Description
General Options	DefaultStorageType	Specifies which storage format that a CCPulse+ application uses. If this option is not defined or the value for it is specified incorrectly, CCPulse+ uses the default value, `Legacy` (OLE storage format). Valid values: `Legacy, Binary, XML` Default value: `Legacy`
Options for CCPulse 7.*x* Storages	ActionsStorageFullPath	Defines the full path and name of the file in which CCPulse+ stores pre-8.0 actions. This option is retained for backward compatibility.
	TemplatesStorageFullPath	Defines the full path and the name of the file in which CCPulse+ stores pre-8.0 templates. This option is retained for backward compatibility.
	ThresholdsStorageFullPath	Defines the full path and the name of the file in which CCPulse+ stores pre-8.0 thresholds. This option is retained for backward compatibility.
	WorkspacesStorageFullPath	Defines the full path and the name of the file in which CCPulse+ stores pre-8.0 user workspaces. This option is retained for backward compatibility.

Table 6: Configuration Options for the Storage Section (Continued)

Type	Option	Description
Options for CCPulse+ Binary Data Files	BinActionsStorageFullPath	Defines the full path to the valid directory name in which CCPulse+ stores, in binary format, actions that result when an administrator clicks `Finish` in the Actions Wizard. This path must preexist and point to a valid path; CCPulse+ does not create this path if it does not exist.
	BinTemplatesStorage FullPath	Defines the full path to the valid directory name in which CCPulse+ stores, in binary format, templates that result when an administrator creates a new template and clicks `Finish` in the Template Wizard. This path must preexist and point to a valid path; CCPulse+ does not create this path if it does not exist.
	BinThresholdsStorage FullPath	Defines the full path to the valid directory name in which CCPulse+ stores, in binary format, thresholds that result when an administrator clicks `Finish` in the Threshold Wizard. This path must preexist and point to a valid path; CCPulse+ does not create this path if it does not exist.
	BinWorkspacesStorage FullPath	Defines the full path to the valid directory name in which CCPulse+ stores, in binary format, user workspaces. This path must preexist and point to a valid path; CCPulse+ does not create this path if it does not exist.

Table 6: Configuration Options for the Storage Section (Continued)

Type	Option	Description
Options for CCPulse+ XML Data Files	XMLActionsStorage FullPath	Defines the full path to the valid directory name in which CCPulse+ stores, in XML format, actions that result when an administrator clicks `Finish` in the Actions Wizard. This path must preexist and point to a valid path; CCPulse+ does not create this path if it does not exist.
	XMLTemplatesStorage FullPath	Defines the full path to the valid directory name in which CCPulse+ stores, in XML format, templates that result when an administrator creates a new template and clicks `Finish` in the Template Wizard. This path must preexist and point to a valid path; CCPulse+ does not create this path if it does not exist.
	XMLThresholdsStorage FullPath	Defines the full path to the valid directory name in which CCPulse+ stores, in XML format, thresholds that result when an administrator clicks `Finish` in the Threshold Wizard. This path must preexist and point to a valid path; CCPulse+ does not create this path if it does not exist.
	XMLWorkspacesStorage FullPath	Defines the full path to the valid directory name in which CCPulse+ stores, in XML format, user workspaces. This path must preexist and point to ta valid path; CCPulse+ does not create this path if it does not exist.

UserInterface Section

The `UserInterface` section holds configuration options with which you can specify how CCPulse+ is to display data in its interface. (See Table 7.)

Table 7: Configuration Options for the UserInterface Section

Option	Description
DisableThresholdsDlg	Use this option to enable contact-center operators who are configured as CCPulse+ Users, to set thresholds, or to prevent them from doing this. This option controls visibility of the `Set Threshold` menu item on the shortcut menu that is invoked when the user right-clicks a statistic or statistics group. Setting this configuration option has no meaning for users who are configured as CCPulse+ Administrators. Full threshold and action functionality is always available to CCPulse+ Administrators. Default Value: `false` Valid Values: `true, false` Change Takes Effect: Upon restart **Note:** This option is not present in the CCPulse+ template.
EnableDNAliases	Use this option to configure CCPulse+ to display either DN numbers or DN aliases in the CCPulse+ workspace. To display DN aliases, specify `true` for this option's value. To display DN numbers, specify `false`. CCPulse+ synchronizes DN aliases with any changes you might make to a DN object's properties. Default Value: `true` Valid Values: `true, false` Change Takes Effect: Upon restart
EnableXMLExport	Use this option to allow users to make use of the `Export to XML` menu item on the CCPulse+ `File` menu. The `Export to XML` menu item enables users to export their workspaces in XML format to a location that they designate. If this option is set to the default value of `false`, this menu item will not appear on the `File` menu. The workspace that you export is for informational purposes only. CCPulse+ does not import information about workspaces from this file nor does the export affect the location of the workspaces storage. Default Value: `false` Valid Values: `true, false` Change Takes Effect: Upon restart

Table 7: Configuration Options for the UserInterface Section (Continued)

Option	Description
ShowAgentLoginID	Use this option to configure CCPulse+ to display the agent's login ID in the `Extended Current Status` window. Specify `true` to display login IDs, or `false` to hide them. This window might display `n/a` (for "not applicable") in place of a login ID when the ID is unavailable, or the agent is logged out. This option does not apply to Agent Group objects or other object types. Default Value: `true` Valid Values: `true, false` Change Takes Effect: Upon restart
ShowAgentSkills	Use this option to configure CCPulse+ to display the skills associated with selected agents in the `Extended Current Status` window. Specify `true` to display agent skills or `false` to hide them. If no skills are defined in Configuration Manager or if skills have not been assigned to the selected agent(s) and `ShowAgentSkills` is set to `true`, CCPulse+ displays `n/a` (for not applicable) in the `Extended Current Status` window. The 7.0.1 release of CCPulse+ introduced a `Skill` icon ( ), which precedes the listing of skills in the `Extended Current Status` window. Default Value: `false` Valid Values: `true, false` Change Takes Effect: Upon restart
ShowDurationBefore StatusName	Use this option to affect the format in which CCPulse+ displays information about current status in the `Views` pane. If this option is set to `true`, CCPulse+ displays first the duration of a particular state, followed by the current state, by using the format: `(hh:mm:ss) currstatus`. If it is set to `false`, CCPulse+ first displays the name of the current status and then its duration: `currstatus (hh:mm:ss)`. By default, CCPulse+ displays status duration after the current status, even if the option is absent from the configuration of your CCPulse+ `Application` object. Such placement facilitates sorting by current status should you want to use this functionality. (Incidentally, CCPulse+ does not allow you to sort by duration.) You might, however, prefer that duration precede current status. Figure 3 provides an illustration of this preference. Default Value: `false` Valid Values: `true, false` Change Takes Effect: Upon restart

Table 7: Configuration Options for the UserInterface Section (Continued)

Option	Description
ShowStatusReasons	Use this option to configure CCPulse+ to display reason-code attached data. This data provides a hardware- or software-related reason for an object's current status and is associated with the agents that you select in the `Extended Current Status` window. CCPulse+ obtains this information from Stat Server, which, in turn, gets this information from one of two attributes that are supplied by each T-Server `TEvent`: • `Extensions` attribute—Its `ReasonCode` key (for hardware-related reasons). • `Reasons` attribute (for data that is supplied by software such as the Genesys SoftPhone or ActiveX). Specify `true` to display the value of this reason code or `false` to hide it. Default Value: `false` Valid Values: `true, false` Change Takes Effect: Upon restart Reason codes apply only to the following noncall-related statuses: • `LoggedIn` • `LoggedOut` • `AfterCallWork` • `Ready` • `NotReady` As reason codes are associated with a particular DN, CCPulse+ cannot display reason codes if the agent is not logged in to a DN. In addition, CCPulse+ cannot display reason codes if they are attached to a `TEvent` in other than the `Reasons` or `Extension` attributes. If T-Server attaches no data to the TEvent, CCPulse+ displays `n/a` (for not applicable) in the `Extended Current Status` window. The 7.0.1 release of CCPulse+ introduced a `Reasons` icon ( ), which precedes the listing of reasons in the `Extended Current Status` window. Refer to the *Genesys 7 Events and Models Reference Manual* for information about `TEvent` structure. **Note:** In some cases, CCPulse+ might show user data and business attributes; (for example, `MediaType)`, in addition to actual reasons. Use a string for the reasons values; (for example, `Lunch` or `Meeting),` to separate actual reasons from possible non- reason data. `Extended Current State` window and `Current state` in the View can show reasons if enabled by the appropriate option.

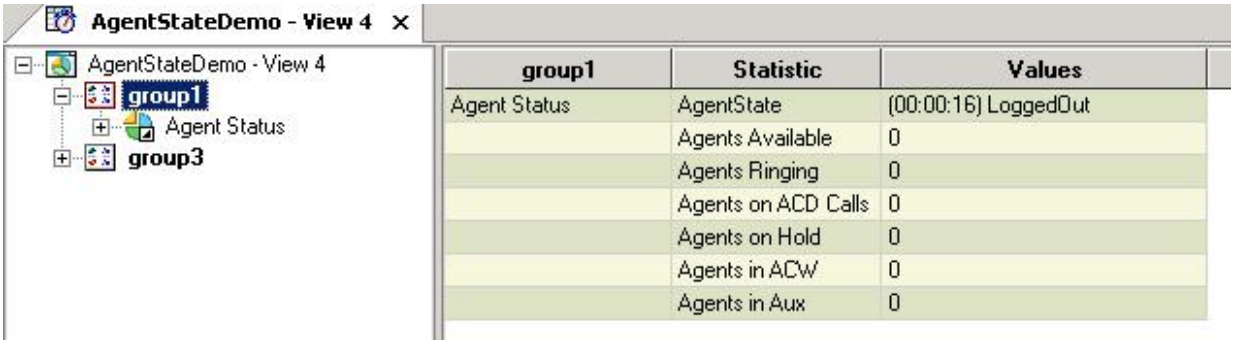

Figure 3: CCPulse+ Displays Duration First when ShowDurationBeforeStatusName Is Set to True

View Section

The `View` section holds configuration options that affect the presentation of data in CCPulse+ views. (See Table 8.)

Table 8: Configuration Options for the View Section

Option	Description
CurrentStateRefresh Interval	This option specifies the time interval at which `CurrentState` statistics will be refreshed in the view. If the view does not contain any `CurrentState` statistics, CCPulse+ ignores this option. CCPulse+ adjusts any value specified over the maximum value to the maximum value. **Note:** If you enter a `0` (zero), a negative number, or any alphanumeric string, you will disable this setting. The current state duration will be not be refreshed until it receives an update from the Stat Server. Default Value: `5` (seconds) Valid Values: `0` to `86399` Change Takes Effect: Upon restart
DelayBeforeRemoval	This option specifies the number of seconds that logged-out agents remain visible, but in a dimmed (greyed out) state, in a real-time V/AG dynamic membership view, before CCPulse+ removes them from the view entirely. A value of `0` instructs CCPulse+ to immediately remove agents from the view; however, in reality, some small amount time could pass before the view is updated. CCPulse+ resets any value specified outside the range of valid values to `0`. Default Value: `15` (seconds) Valid Values: `0` to `3600` Change Takes Effect: Upon restart

Table 8: Configuration Options for the View Section (Continued)

Option	Description
DisableCreateRemove Views	Use this option to enable CCPulse+ Administrators to restrict the creation and removal of views by contact-center operators, or to prevent them from doing this. The setting of this configuration option has no meaning for users who are configured as CCPulse+ Administrators. Administrators can always create and/or remove views, regardless of this option's setting. However, setting this value to `true` prevents agents who are configured as CCPulse+ Users from performing these actions. Default Value: `false` Valid Values: `true, false` Change Takes Effect: Upon restart **Note:** This option is not present in the CCPulse+ template.
DisableHideShowActions OnGraph	Use this option to enable or disable hide/show actions on the `Graph` view. Setting this configuration option will greatly reduce the time it takes for CCPulse+ to open a workspace with hide/show actions. Default Value: `false` Valid Values: `true, false` Change Takes Effect: Upon restart **Note:** This option is not present in the CCPulse+ template.
DisableRowColVisibility Change	Use this option to enable contact-center operators who are configured as CCPulse+ Users, to alter the visibility of rows and/or columns in the `Table` view, or to prevent them from doing this. In addition, if you set this configuration option to `true`, the shortcut menu items to hide and show rows and columns are disabled for CCPulse+ users. The setting of this configuration option has no meaning for users who are configured as CCPulse+ Administrators. Administrators can always hide and show rows and columns in the `Table` view and the corresponding shortcut menu items remain enabled, regardless of this option's setting. Default Value: `false` Valid Values: `true, false` Change Takes Effect: Upon restart **Note:** This option is not present in the CCPulse+ template.

Table 8: Configuration Options for the View Section (Continued)

Option	Description
MaxNumberOfAgents	This option specifies the maximum number of agents that CCPulse+ displays in a real-time V/AG dynamic membership view regardless of the number of available agents or their current status. The default value that is provided in the CCPulse+ 7.5+ application template allows up to 75 agents in the display; this is also the value that CCPulse+ uses if this option is not set. CCPulse+ sets any value that is outside the range of valid values to `1`. Default Value: `75` (agents) Valid Values: `1` to `450` Change Takes Effect: Upon restart

Workspace Section

The `Workspace` section holds configuration options that affect your CCPulse+ workspace. (See Table 9.) Note that because the options defined in this section are defined within the CCPulse+ `Application` object in Configuration Server, their settings affect all CCPulse+ instances that are configured to use the object. You cannot change these settings for every instance of CCPulse+ individually. Instead, you have to define a new `Application` object for each combination of configuration settings that you wish to set and then configure each instance to use the appropriate `Application` object.

Table 9: Configuration Options for the Workspace Section

Option	Description
CurrentStateEnable	Use this option to enable or disable the display of current status for all configuration objects that are selected for monitoring on the Call Center Objects pane. Consider using this option if you want to increase runtime performance of CCPulse+ and decrease startup time—especially if you regularly open workspaces that contain a large number of objects. Default Value: `true` Valid Values: `true, false` Change Takes Effect: Upon restart

Table 9: Configuration Options for the Workspace Section (Continued)

Option	Description
DisableAgentCurrentState	Use this option to enable or disable the visibility of current-state information on individual agents in all CCPulse+ panes for contact-center operators who are configured as CCPulse+ Users. The visibility of current-state information for other object types remains unaffected by the setting of this option; instead, visibility is controlled by the `CurrentStateEnable` configuration option. In situations in which your configuration includes one-person agent groups, consider adding a pseudo `Person` object to these groups, to preserve an environment in which operators should not be able to obtain current-state information on individual agents. Setting this option also affects all contact-center operators configured as CCPulse+ Users who are logged into the same CCPulse+ application, by preventing the creation and displaying of other agent statistics. To restrict users on an individual basis, set the `AgentStats` registry setting on page 45. Setting this option has no meaning for users who are configured as CCPulse+ Administrators. Administrators can always create and/or display agent statistics and view current-state information on individual agents regardless of this option's setting. However, setting this value to `false` prevents agents who are configured as CCPulse+ Users from performing the following actions: • Displaying agent status icons for agents in tree view. • Creating historical and real-time views for individual agents. • Creating historical and real-time views for members, and dynamic membership views for agent groups. • Setting thresholds for agents. • Displaying an extended status for individual agents. • Displaying an extended status for agent groups. Default Value: `false` Valid Values: `true, false` Change Takes Effect: Upon restart

Table 9: Configuration Options for the Workspace Section (Continued)

Option	Description
DisableThresholdsDlg	Use this option to enable or disable contact-center operators who are configured as CCPulse+ Users, to set and/or remove thresholds. The setting of this configuration option has no meaning for users who are configured as CCPulse+ Administrators. Administrators can always set and/or remove thresholds, regardless of this option's setting. However, setting this value to `true` prevents agents who are configured as CCPulse+ Users from performing these actions. Default Value: `false` Valid Values: `true, false` Change Takes Effect: Upon restart
EnableAgentCapacity	This option enables or disables the display of agent-capacity information per media in the `Object` tree of the CCPulse+ interface. Default Value: `false` Valid Values: `true, false` Change Takes Effect: Upon restart
EnableAgentCapacityStatus	This option, in conjunction with the `EnableAgentCapacity` configuration option, enables or disables the display of agent and agent group capacity information in the `Object` tree. If this option is set to `true` and the `EnableAgentCapacity` option is set to `true`, CCPulse+ displays agents' capacity in the `Object` tree. If this option is set to `false`, or the `EnableAgentCapacity` option is set to `false`, CCPulse+ does not show capacity. Default Value: `false` Valid Values: `true, false` Change Takes Effect: Upon restart

Table 9: Configuration Options for the Workspace Section (Continued)

Option	Description
EnableAgentGroupCapacity	This option, in conjunction with the `EnableAgentCapacity` and `EnableAgentCapacityStatus` configuration options, enables or disables the calculation of capacity per media for agent groups in the `Object` tree of the CCPulse+ interface. If this option is set to `true` and both the `EnableAgentCapacity` and `EnableAgentCapacityStatus` option are also set to `true`, CCPulse+ displays agent groups' capacity in the `Object` tree. If this option is set to `false`, or either of the `EnableAgentCapacity` or `EnableAgent CapacityStatus` option are set to `false`, CCPulse+ does not show capacity. Agent group capacity is defined as the sum of the corresponding media capacity values for the agents in the group: `Group max capacity/current interactions/routable interactions = Sum (Agent max capacity/current interactions/routable interactions)` **Note:** Displaying agent group capacity will have a linear impact on CCPulse+ performance (depending on the number of calls that the group members receive). Default Value: `false` Valid Values: `true, false` Change Takes Effect: Upon restart
IgnoreNewObjectNotifications	Use this option to specify whether CCPulse+ should accept or ignore notifications from Configuration Server about newly created reportable objects. If this option is set to `true`, CCPulse+ ignores notifications of new objects—any new reportable objects will not appear in the workspace. If this option is set to `false`, new objects will appear instantly in the `Call Center Objects` pane. When the workspace is saved, these objects become a permanent part of the workspace, like the other configuration objects that were explicitly selected for monitoring when the workspace was initially created. Default Value: `false` Valid Values: `true, false` Change Takes Effect: Upon restart

Table 9: Configuration Options for the Workspace Section (Continued)

Option	Description
SafeScriptingMode	Use this option to specify whether CCPulse+ will spawn a separate thread to execute VBScripts. If set to `true`, threshold and action VBScripts will be executed using a separate thread. This is a safe but slower method of operation. If set to `false`, all scripts will be executed using the main process, and the `Timeout` option, shown on the `Threshold Creation` and `Action Creation` pages of the Threshold and Action wizards (respectively), will become disabled. Figure 4 shows the bottom half of the `Threshold Creation` page with the timeout area disabled. Default Value: `true` Valid Values: `true, false` Change Takes Effect: Upon restart
UseStatServerVirtualGroups	Use this configuration option to control from where CCPulse+ gathers information about virtual agent group membership—from either Stat Server or Configuration Server. Stat Server enables you to specify a wider range of criteria in order to qualify virtual agent group membership. These criteria include a consideration of the ACD queue or switch into which an agent is logged in, skill expressions that are configured for the agent, and real-time data such as status. Configuration Server, on the other hand, uses skill expressions alone to determine group composition. Refer to "Virtual Agent Groups" in the *Framework Stat Server User's Guide* for additional information. If this option is set to `true`, CCPulse+ obtains information about group membership from Stat Server. If it is set to `false`, CCPulse+ obtains information from Configuration Server. Regardless of the setting of this option, please note that the new feature—creating dynamic agent- and virtual agent-group views—uses Stat Server as the source for obtaining group membership. Refer to the "Creating Dynamic Virtual/Agent Group Views" topic in *Reporting CCPulse+ Help* for more information about dynamic views. **Note:** This feature is available only in the CCPulse+ real-time environment. CCPulse+ historical reporting does not capture dynamic changes to the content of virtual agents groups. Default Value: `false` Valid Values: `true, false` Change Takes Effect: Upon restart

Table 9: Configuration Options for the Workspace Section (Continued)

Option	Description
WorkspaceAutoLoad	This option determines whether CCPulse+ loads the last workspace used when CCPulse+ starts. If this option is set to `true`, CCPulse+ loads the workspace that you were using the last time that you closed CCPulse+, to enable you to bypass the explicit selection steps that are otherwise required. If it is set to `false`, CCPulse+ does not automatically load any workspace. CCPulse+ stores the name of the most recently used workspace in the Microsoft Windows registry. Default Value: `true` Valid Values: `true, false` Change Takes Effect: Upon restart

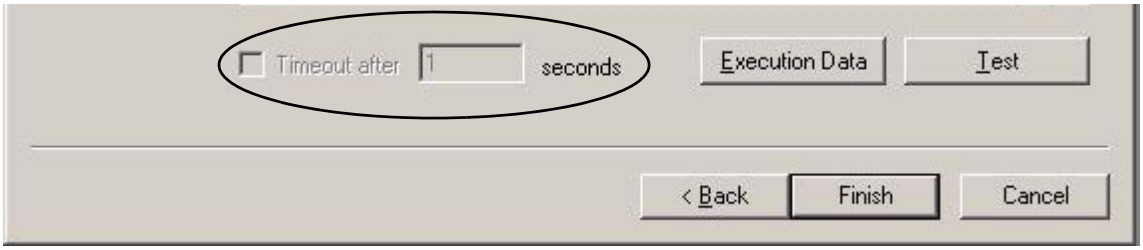

Figure 4: No Timeout Specification with SafeScriptingMode=False

Changing the Display of Dates and Time

CCPulse+ release 7.0.1 introduced functionality that allows you to define how time and dates appear in hourly and daily historical reports. Specify the desired format on the `Date` and `Time` tabs of your Control Panel's `Regional Settings`, and CCPulse+ will automatically change its column-title display of your hourly and daily reports to the format that is specified. CCPulse+ also uses these regional settings for date and time formats on the Daily and Hourly Historical report titles.

Figure 5 shows instances of two different hour-format settings. The first image uses a regional hour format of `M/d/yy` representing July 27, 2001 as `7/27/01`. The second image uses the `dd/MM/yyyy` format and displays the same date as `27/07/2001`.

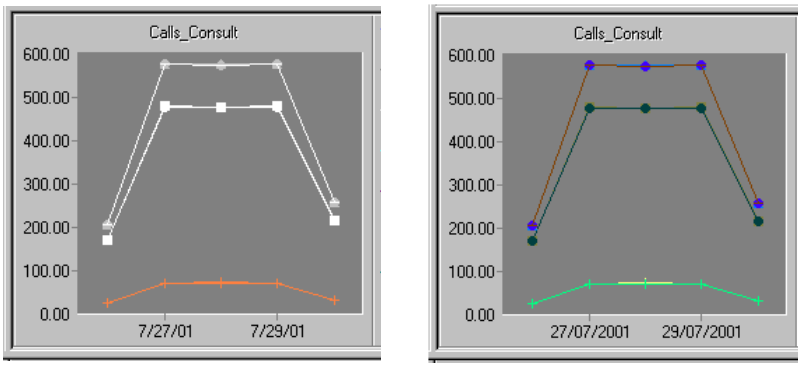

Figure 5: Different Hour-Format System Settings Reflected in Historical Daily Views

Figure 6 shows instances of two different time-format settings. The first image uses a regional time format of `h:mm:ss tt`, representing 4 o'clock in the afternoon as `4:00:00 PM`. The second image uses the `HH:mm:ss` format and displays the same time as `16:00:00`.

Although CCPulse+ accepts dynamic system changes of regional time and dates, if CCPulse+ is currently displaying a view when the change is made, you must deactivate and reactivate the view in order to observe its effect.

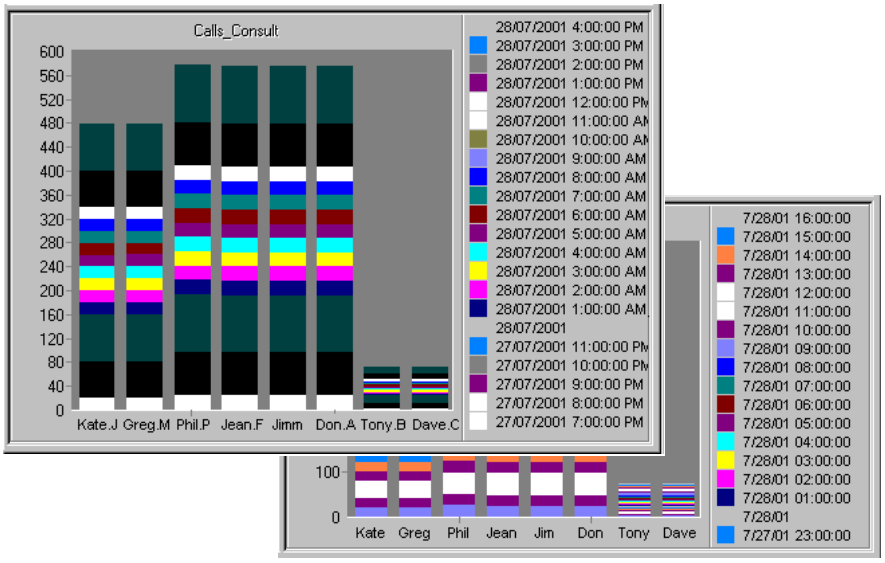

Figure 6: Different Time-Format System Settings Reflected in Historical Hourly Views

CCPulse+ Registry Settings

You can edit the `AgentStats` CCPulse+ configuration setting *only* by using the Microsoft Windows Registry Editor. Even though you can edit the `FormatAgentInfo` configuration setting using the Microsoft Windows Registry

Editor, Genesys recommends that you use the `CfgApplication` configuration option in the `ObjectFormat` section to set this value.

Where applicable, Genesys recommends that you use the `Settings` menu option in CCPulse+ to set these values. However, if you prefer to configure these settings directly using the Registry, refer to the path information that is provided for each setting in the following table. (See Table 10.)

You can no longer edit the following three CCPulse+ configuration settings by using the Microsoft Windows Registry Editor. They have been removed from the registry and moved to the `Settings` menu option under the `Tools` menu of the CCPulse+ interface:

- `GraphTitleFontSize`
- `GraphBkColor`
- `GraphAxisFontSize`

Note: Unlike settings that are configured in the CCPulse+ `Application` object, which apply to all CCPulse+ instances that use that application, registry key changes affect only the local workstation.

Access the root for all these settings through the following path in the Registry:

`HKEY_CURRENT_USER\Software\GCTI\CallCenter\`

Configure your user interface by entering the desired value next to the appropriate key. (See Table 10.)

Table 10: CCPulse+ Registry Settings

Registry Setting	Description
AgentStats	Use this registry setting to enable CCPulse+ Administrators to prevent the creation and displaying of other agent statistics by contact-center operators who are configured as CCPulse+ Users on an individual basis. Setting the `AgentStats` registry setting has no meaning for users who are configured as CCPulse+ Administrators. Administrators can always create and/or display agent statistics, regardless of this setting. However, setting this value to `RESTRICTED_STATS` prevents agents who are configured as CCPulse+ Users from performing the following actions: • Displaying agent status icons for agents in tree view. • Creating historical and real-time views for individual agents. • Creating historical and real-time views for members, and dynamic membership views for agent groups. • Setting thresholds for agents. • Displaying an extended status for individual agents. • Displaying an extended status for agent groups.

Table 10: CCPulse+ Registry Settings (Continued)

Registry Setting	Description
AgentStats (continued)	Setting the `DisableAgentCurrentState` configuration option under the `Workspace` section, on page 39 to `true` will also enable this functionality, but will affect all CCPulse+ users logged into the same CCPulse+ application. Setting the `AgentStats` registry setting to `RESTRICTED_STATS` on individual computers, enables this functionality on an individual user basis. Default Value: Depends on the `DisableAgentStatus` configuration option. If `DisableAgentStatus` =`true`, the default value of `AgentStats` is `RESTRICTED_STATS`. If `DisableAgentStatus` = `false`, the default value of `AgentStats` is `FULL_STATS`. Valid Values: `RESTRICTED_STATS, FULL_STATS` Path:`HKEY_LOCAL_MACHINE\SOFTWARE\GCTI\CallCenter\VIEW_STATS value AgentStats`
DeltaAvgTime	Specifies the default value, in seconds, of the `Insensitivity` parameter for statistics in the `Average Time` category. Default Value: `2` Valid Values: `2–600` Path: `HKEY_CURRENT_USER\Software\GCTI\CallCenter\CUSTOM DEFINITION`
DeltaCntCalls	Specifies the default value, in seconds, of the `Insensitivity` parameter for statistics in the `Current Number` category. Default Value: `1` Appropriate Values: `1–100` Path: `HKEY_CURRENT_USER\Software\GCTI\CallCenter\CUSTOM DEFINITION`
DeltaTotalTime	Specifies the default value, in seconds, of the `Insensitivity` parameter for statistics in the `Total Time` category. Default Value: `10` Valid Values: `10–600` Path: `HKEY_CURRENT_USER\Software\GCTI\CallCenter\CUSTOM DEFINITION`

Table 10: CCPulse+ Registry Settings (Continued)

Registry Setting	Description

FormatAgentInfo	Specifies how CCPulse+ displays agent information on the `Call Center Objects` pane of the CCPulse+ main window.
	Default Value: Varies, depending on the language that is detected by your operating system
	Valid Values: Strings that contain any combination of the following parameters:
	`%n`=Employee ID
	`%F`=First name
	`%f`=Abbreviated first name
	`%L`=Last name
	`%l`=Abbreviated last name
	For example, if you have an agent who is named Vince Duncan and whose employee number is `9001`, the following string:
	`FormatAgentInfo = %n-%f.%L`
	yields the following display:
	`9001 - V. Duncan`
	Note: Separate the parameters by using punctuation with which you want the elements to display.
	Path: `HKEY_CURRENT_USER\Software\GCTI\CallCenter\CUSTOM DEFINITION`

GridSortingInterval	Specifies how often, in seconds, CCPulse+ re-sorts the grid if it is set to sort in either ascending or descending order by column. Default Value: 60 Valid Values: 5-600 Path: `HKEY_CURRENT_USER\Software\GCTI\CallCenter\PulseGridProperties\GridSortingInterval`
HistoricalUpdate Frequency	Specifies the default value, in seconds, of the `Notification Frequency` property for statistics in the `ElapsedTimePercentage`, `RelativeTimePercentage`, `RelativeNumberPercentage`, and `TotalNumberInTimeRangePercentage` categories. Default Value: `60` Valid Values: `60`–`600` Path: `HKEY_CURRENT_USER\Software\GCTI\CallCenter\CUSTOM DEFINITION`
IntervalForCurrent Update	Specifies the default value, in seconds, of the `Notification Frequency` property for `Current State` and `Current Number` statistics. Default Value: `2` Valid Values: `2`–`600` Path: `HKEY_CURRENT_USER\Software\GCTI\CallCenter\CUSTOM DEFINITION`

Table 10: CCPulse+ Registry Settings (Continued)

Registry Setting	Description
NotifyEvery	Specifies the default value, in seconds, of the `Notification Frequency` property for statistics. Default Value: `60` Valid Values: `60–1800` Path: `HKEY_CURRENT_USER\Software\GCTI\CallCenter\CUSTOM DEFINITION`

Genesys

Chapter

3 Historical Reporting Considerations for CCPulse+

This chapter discusses CCPulse+ historical reporting. It covers certain points to consider before you use CCPulse+ historical reporting functionality. Using this functionality is completely optional. If you intend to only use CCPulse+'s real-time functionality, you do not have to connect CCPulse+ to a Data Mart and you can skip this chapter. If you are interested in employing CCPulse+'s historical reporting functionality, read the following sections:

- Data Mart Connections, page 49
- Composite Metrics, page 50
- Column Names, page 50
- The Last N Intervals Parameter, page 51
- Historical Associations for Statistics, page 51
- Propagation of Template Changes, page 51

Data Mart Connections

You must configure a Data Mart connection on the `Connections` tab of your CCPulse+ `Application` object. If CCPulse+ cannot connect to the Data Mart, it cannot perform any historical reporting. If you use the CCPulse+ Wizard to configure CCPulse+, you can specify a Data Mart application on the ETL Runtime window. Otherwise, you must update your CCPulse+ `Application` object manually.

CCPulse+ does not maintain a constant connection to Data Mart. Instead, it connects for each request made and then disconnects until the next request. This functionality saves Data Mart resources.

Note: For CCPulse+, you can use a Data Mart that has only one active Configuration Server defined to it. (*Active* indicates that Configuration Server has an ODS [Operational Data Storage] assigned to it within ETL Assistant.) If your Data Mart collects information from multiple Configuration Servers, you cannot use it with CCPulse+.

Composite Metrics

Prior to release 7.0, you could not use CCPulse+ to supply composite metrics in reports—except for those metrics already provided by the ready-to-use reports. To correct this shortcoming, the 7.0 release introduced a custom-formula capability, allowing you to define averages, percentages, and more. Refer to the "Template Wizard" topic in *Reporting 8.0 CCPulse+ Help* for information on how to use this feature.

Column Names

Data Mart stores information in folder templates that include only a single object (Agent, Place, Queue, and so on). However, in CCPulse+, you can create custom templates that include multiple compatible objects. To link object data stored in Data Mart, CCPulse+ identifies metrics by the Data Mart column name.

For this to function properly, the Data Mart folder template for each object to be incorporated into the CCPulse+ template must include the *identical* column name used in the CCPulse+ template. Otherwise, data from the Data Mart template is omitted from the CCPulse+ template.

Warning! If more than one custom template uses the same column name to specify different metrics, CCPulse+ might combine these metrics incorrectly.

For more information about the Data Mart folder templates and column names provided in reports from Genesys solutions, refer to the "Data Mart Folder Templates" and "Data Mart Composite Metrics" sections in Chapter 6 of the *Reporting Technical Reference Guide*.

The Last N Intervals Parameter

CCPulse+ lets you select `Last N Intervals` as a time parameter for historical reporting. CCPulse+ searches for data in the `N *` (aggregation) intervals for the current time and then displays a maximum of `N Intervals`, depending on data availability.

For example, if you request data from the last three weeks, CCPulse+ searches for data in the previous three-week interval, and then returns a maximum of three records. However, if the data collection for that object or metric ceased several months ago, CCPulse+ returns a null response. This optimization methodology prevents resource-intensive searches for data that exists at some unspecified point in the past. Refer to the "CCPulse+ Performance" section of the *Reporting 7.5 Reference Manual* to learn about other parameters that affect both real-time and historical performance.

To retrieve such data, make a request for data from a specified set of dates.

Historical Associations for Statistics

You cannot set historical associations for custom report layouts until you complete the following:

1. Create a layout template by using Data Modeling Assistant (DMA)
2. Assign a specific column name to it.
3. Create and activate a report layout based on the new layout template.

After the statistic has been propagated to Data Mart—which usually takes a few hours—you can make the historical association in CCPulse+.

Note: CCPulse+ reads all available Data Mart metrics at start-up only. In order for CCPulse+ to accept newly created or changed metrics, you must restart it.

Propagation of Template Changes

Earlier releases of CCPulse+ did not propagate to existing historical views any changes that you made to the underlying CCPulse+ historical template. Starting with the 7.0.2 release, however, CCPulse+ does propagate such changes upon reload of your workspace. This behavior now parallels how CCPulse+ propagates changes made to real-time templates.

Chapter

4 Server Connections to CCPulse+

Defined within the CCPulse+ `Application` is connection information about the following servers:

- Stat Server, for monitoring real-time data
- DB Server, for gathering historical data that is stored in Data Mart (optional connection)

CCPulse+ maintains connections with Configuration Server to retrieve information about CCPulse+ application configuration and call center objects.

This chapter describes how CCPulse+ behaves when any of these server connections, or the connections to their designated backups, fails. This chapter also describes what actions you should take to regain full functionality under these circumstances.

Note: For information about the Genesys security feature, client-side port definition, which controls and secures the connections between CCPulse+ and these servers, see "Client-Side Port Definition" on page 19.

The scenarios in this chapter begin with a CCPulse+ session that has all of its connections established. For any number of reasons, the connections might subsequently fail—and in any order—to affect normal CCPulse+ operation.

This chapter contains the following sections:

- Configuration Server Connection Failure, page 54
- Stat Server Connection Failure, page 56
- DB Server Connection Failure, page 58

Configuration Server Connection Failure

CCPulse+ maintains two persistent connections to Configuration Server (or Configuration Server Proxy, if so configured) upon login. If one of these connections is dropped, the CCPulse+ status bar instantly changes from `CfgServer ONLine` to `CfgServer OFFLine`, but the CCPulse+ application continues to function. A broken Configuration Server connection means that CCPulse+ cannot receive updates about changes to the configuration of contact-center objects such as:

- Object additions or deletions (for example, the addition/deletion of DNs, agents, and queues) that CCPulse+ would otherwise monitor.
- Group composition changes (for example, additions/deletions to agent groups and place groups).

Furthermore, you cannot open an existing workspace or create a new one if CCPulse+ has no connection to Configuration Server.

With regard to changes that you make to the CCPulse+ `Application` object itself, CCPulse+ reads this information only upon startup; therefore, a Configuration Server disconnection after an initial successful connection affects CCPulse+'s ability to read its own configuration only upon the *next* startup.

In a failed Configuration Server connection, CCPulse+ silently executes its reconnection procedure.

1. CCPulse+ attempts to reconnect automatically to the current Configuration Server (primary) using the number of `Reconnect Attempts` configured for this application in Configuration Manager. The following message will appear in the status bar: `Reconnecting to CfgServer (attempt X of Y).`

   Figure 7 shows the `Server Info` tab of a configuration Server application in Configuration Manager, where the number of `Reconnect Attempts` is configured.

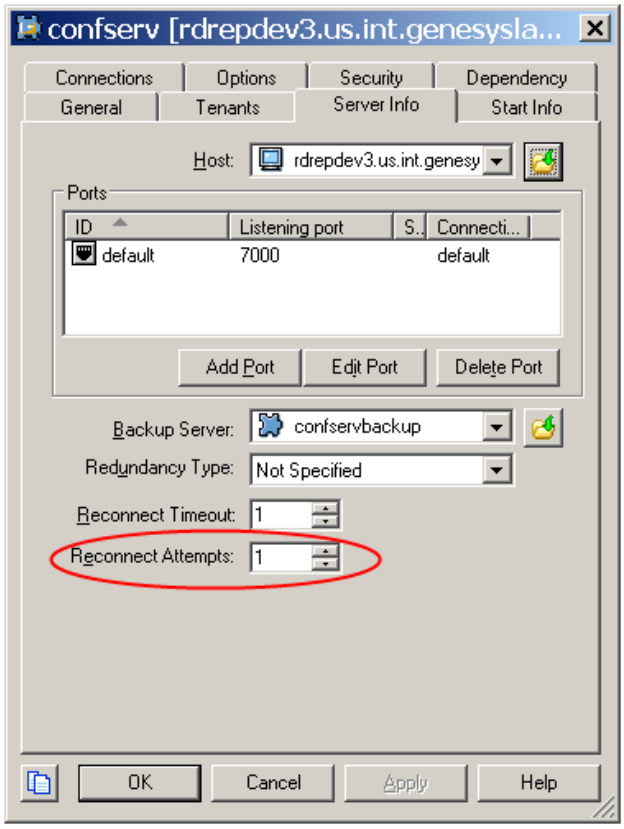

Figure 7: Number of Reconnection Attempts

2. When number of attempts is reached without success, CCPulse+ attempts to switchover from primary to backup and back until the connection succeeds or you exit CCPulse+.
3. When the switchover happens, the status bar displays the attempt and the number of configured attempts for the server to which it switched and the number of re-connect attempts is reset.

   Upon a successful connection to either the primary or backup Configuration Server, the CCPulse+ status bar reads as follows:

   `Primary/Backup CfgServer ONLine`
4. If there is still an error connecting to both the primary and backup server, CCPulse+ displays an error message with the information about which server's last connection attempt failed—for example, the backup server.
5. When Configuration Server (or its backup, if specified) does get back online, you must close and then reopen your workspace in order for CCPulse+ to acknowledge Configuration Server status and for CCPulse+ to receive Configuration Server updates. CCPulse+ checks the status of its connection to Configuration Server every time that a workspace is opened.

CCPulse+'s connection to Configuration Server can fail for a number of reasons, including the following:

- Someone closes Configuration Server or kills its process.
- Network problems break CCPulse+'s connection to Configuration Server.

You can continue to use a previously opened and activated CCPulse+ workspace without a connection to Configuration Server; however, you cannot open or create a new workspace without this connection.

Stat Server Connection Failure

CCPulse+ maintains a continuous connection to Stat Server when there is an active workspace open. When this connection fails, the CCPulse+ status bar instantly changes from `Primary/Backup StatServer ONLine` to `StatServer OFFLine`, but the CCPulse+ application continues to function. A broken Stat Server connection means that:

- CCPulse+ cannot send requests for real-time statistics to Stat Server.
- CCPulse+ cannot receive real-time statistical updates from Stat Server.
- CCPulse+ cannot read changes to virtual group composition (if the `UseStatServerVirtualGroups` configuration option is set to `true`).
- You cannot apply thresholds or actions to real-time statistics (but you can still apply these to historical statistics).
- You cannot open the `Extended Current Status` window for any object.
- You cannot retrieve capacity information for any object.

In a failed Stat Server connection, CCPulse+ next silently executes its reconnection procedure.

1. CCPulse+ attempts to reconnect automatically to the current Stat Server (primary) using the number of Reconnect Attempts configured for this application in Configuration Manager. The following message will appear in the status bar: `Reconnecting to StatServer (attempt X of Y)`.

   Figure 8 shows the `Server Info` tab of a Stat Server application in Configuration Manager, where this parameter is configured.

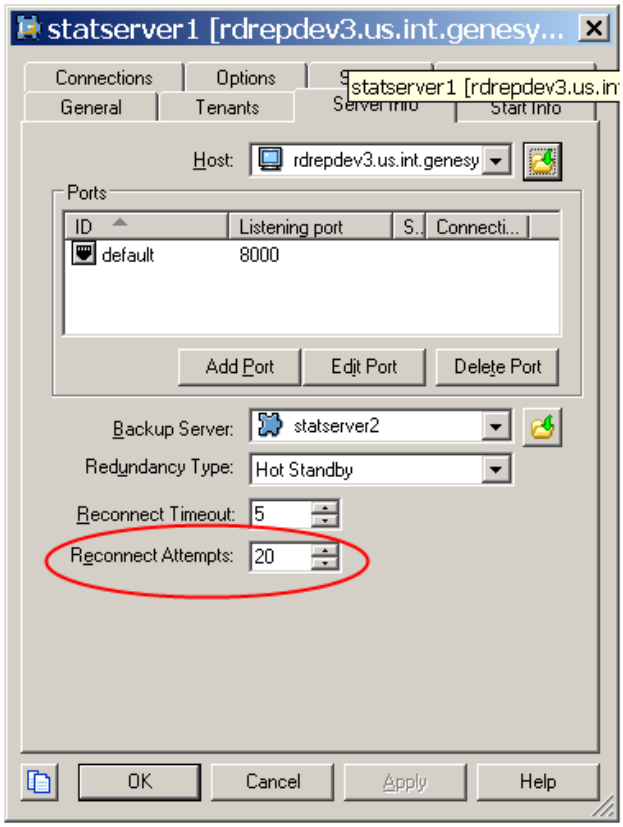

Figure 8: Number of Reconnection Attempts

2. When number of attempts is reached without success, CCPulse+ attempts to switchover from primary to backup and back until the connection succeeds or you exit CCPulse+.
3. When the switchover happens, the status bar displays the attempt and the number of configured attempts for the server to which it switched and the number of re-connect attempts is reset.

   Upon a successful connection to either the primary or backup Stat Server, the CCPulse+ status bar reads as follows:

   `Primary/Backup StatServer ONLine`
4. If there is still an error connecting to both the primary and backup server, CCPulse+ displays an error message with the information about which server's last connection attempt failed—for example, the backup server.
5. When Stat Server (or its backup) does get back online, you must close and then reopen your workspace in order for CCPulse+ to acknowledge the connection and resume normal operation. CCPulse+ checks the status of its connection to Stat Server only when you open a workspace or upon execution of its reconnection procedure.

CCPulse+'s connection to Stat Server can fail for a number of reasons including the following:

- StatServer becomes overloaded.
- Someone closes Stat Server.
- Someone kills the Stat Server process.
- Network problems break CCPulse+'s connection to Stat Server.

You can continue to use a previously opened and activated CCPulse+ workspace without a connection to its Stat Server; although this mode of operation is not particularly useful. However, you cannot open or create a new workspace without this connection.

DB Server Connection Failure

CCPulse+ does not maintain a continuous connection to the DB Server for Data Mart (through its database access point). Instead, CCPulse+ connects to this server (or its backup) on a request-by-request basis, such as when you refresh or create an historical view. When establishing an initial connection to a pair of primary and backup DB Servers, CCPulse tries to switch over to the backup server, if configured, without prompting you.

If initial connection fails after trying backup server CCPulse+ displays an error message with the information about which server's last connection attempt failed—for example, the backup server.

If this backup is not running or if one is not configured, CCPulse+ does not receive the historical information that is requested, and your historical view, for example, displays no data. Under these circumstances, you can continue to use CCPulse+ for reporting real-time data that is gathered directly from Stat Server, provided that this server is running.

As soon as DB Server is back online, you do not need to close or reopen the workspace to regain historical reporting. Merely re-request the historical information that you sought by reopening the related historical view.

CCPulse+'s automatic connection to a designated backup DB Server is a feature that was introduced in the 7.0.2 release.

Chapter

5 Using Custom Statistics

When you are creating or modifying templates, you can build custom statistics from the template's basic statistics by constructing your own formulas through the use of JScript. A basic statistic has assigned to it a Stat Server stat type definition. This chapter includes the following topics:

- Identifying Custom Statistics, page 59
- Using JScript Language, page 60
- Referencing Basic Statistics, page 60
- Using Predefined Objects in Formulas, page 62
- Formatting the Results, page 68

Identifying Custom Statistics

In the CCPulse+ interface, you can distinguish basic statistics (which are predefined in CCPulse+ templates) from custom statistics, as well as if your statistics have been hidden, by the appearance of the icon that precedes the statistic name in the `Predefined Statistics` window of the Templates Wizard. Both basic and custom statistics can be hidden during template creation by using the Templates Wizard:

denotes a basic statistic.

denotes a custom statistic.

denotes a hidden basic statistic.

denotes a hidden custom statistic.

Figure 9, for example, shows six basic statistics and five custom statistics, some of which are hidden, from a `Callback Queue` template in a cutaway of the `Predefined Statistics` window.

Hidden Statistics

CCPulse+ release 8.0.1 introduces functionality that allows you to hide both basic and custom statistics by using the Templates Wizard. Hidden statistics are identified by the appearance of the hidden icon that precedes the statistic name (see page 59). These hidden icons, along with basic and custom statistic icons, are displayed in the `Predefined Statistics` window of the Templates Wizard (see Figure 9). There are three options from which to choose when you are hiding statistics:

- Hide in All Views—Hides the custom statistic in any type of view (tree, table, or graph) for CCPulse+ users, and locks the statistic for CCPulse+ administrators. The main purpose of these statistics is to receive data that can be viewed and used in formulas.
- Hide in Table View—Hides the custom statistic in table view. The corresponding statistic is hidden, but it can be shown at any time in table view.
- Hide in Graph View—Hides the custom statistic in graph view, but it will still be visible in other views.

Administrators can use the “Hide in Graph View” option to increase the performance of CCPulse+ views by hiding insignificant statistics in graph views. For detailed information about how to hide custom or basic statistics, refer to the *Reporting 8.0 CCPulse+ Help*.

Using JScript Language

CCPulse+ supports the JScript language for the custom statistics you define. As a result, you can use the arithmetic operators and delimiters available in this language, including: `+ / * – . ? : == ( ) ; != +=`

Refer to your JScript documentation for a description of these operators and syntax guidelines.

Referencing Basic Statistics

CCPulse+ uses the following syntax for referencing basic statistics in formulas:

```
ccpulse.group("StatisticGroupName").statistic("StatisticName");
```

The trailing semicolon is optional. CCPulse+ includes the name of the statistic group to distinguish the statistic, in case another statistic of the same name exists in another statistical group. If either `StatisticGroupName` or

StatisticName contains no spaces, CCPulse+ drops the corresponding group and statistic delimiters, the parentheses, and the double quotes:

ccpulse.*StatisticGroupName*.*StatisticName*

Example 1 demonstrates the definition of the All Entered custom statistic, which is provided in the Callback Queue template.

Example 1

The Callback Queue template (which is provided with the Voice Callback [VCB] option) makes use of several custom statistics. Figure 9 shows a cutaway of CCPulse+'s Predefined Statistics window and many of this template's statistics. The Requested Statistics list in this window contains the template's categorization of statistic groups and statistics, and it highlights one custom statistic in particular, All Entered, which belongs to the Total Entered statistic group.

Note: Figure 9 contains hidden statistics (CB Entered, Live Distributed, EWT, CB EWT, and AWT). For more information about hidden statistics, see page 60.

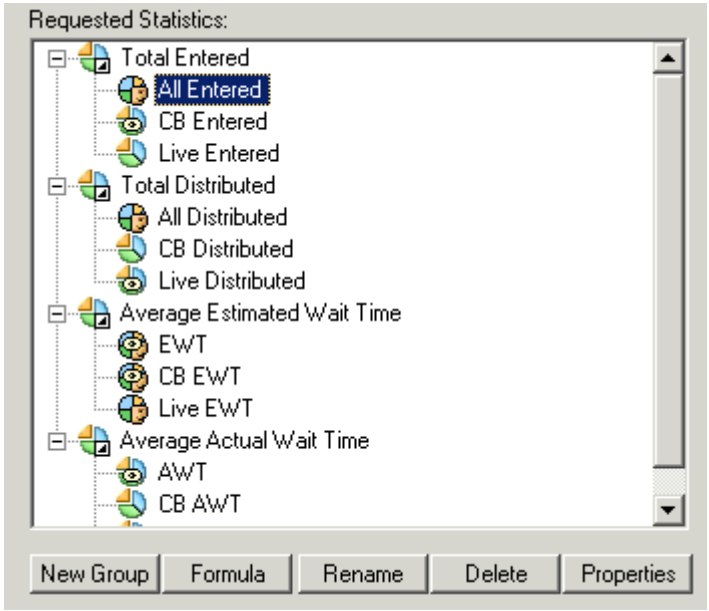

Figure 9: The Predefined Statistics Window

When you open the properties of the All Entered statistic, CCPulse+ opens the ActiveX Scripting Expression dialog box and displays the statistic's formula. Figure 10 on page 62 provides a cutaway of the ActiveX Scripting Expression dialog box, which shows that the definition of All Entered is based on two basic statistics, CB Entered and Live Entered, both of which belong to the Total Entered group.

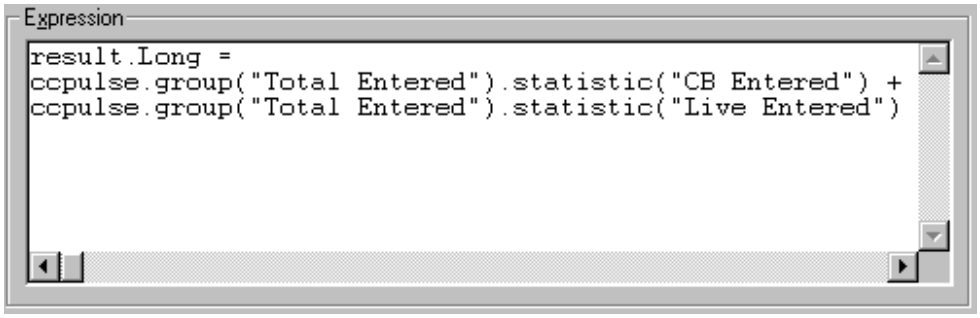

Figure 10: Definition of the All Entered Custom Statistic

Using Predefined Objects in Formulas

In the calculation of custom statistics, you can use qualitative information about a statistic other than its value by applying the `state`, `dn`, and/or `Action` predefined objects within the formula. Example 4 on page 66 provides an expansive example that uses the `state` predefined object in a formula.

Status-Related Information

To obtain information about the current status of a statistic in a formula, use the `state` predefined object and its properties (see Tables 11 and 12). To activate this functionality, be sure to set the `ExtendedCurrentStatus` configuration option to `True` and to define the statistic's underlying stat type appropriately. The description of the option on page 21 provides these details.

Table 11: Properties of the state Object for AgentState

Property	Type	Description
type	String	Type of `state` object. In the 7.5.x release, extended current status information is provided only for `Agent` and `Agent Group` objects. Genesys recommends that you validate an object's `type` property as `AgentState` before using other properties of this object. If the `state` object is not available, the value of the `type` property is `undefined`. In this case, all other properties of the `state` object will be unavailable, which may result in errors.
StartTime	OLE Date	Coordinated Universal Time (UTC) date and time when the current status was established. Suitable for use in the `Date` object that is available in the JScript.
Status	String	Current status name, as provided by Stat Server.
AgentID	String	`EmployeeID`, usually set within Configuration Server.
PlaceID	String	ID of the place where the agent is logged in.
LoginID	String	Login ID used by the agent.

Table 11: Properties of the state Object for AgentState (Continued)

Property	Type	Description
DNs	Collection	Contains information about the DNs used by or available to the agent. The list can be iterated by using `Enumerator` object (available in Jscript) to obtain specific DN data. Refer to the `GetDNState` function in Example 2 for an example on how to use this property. The list can also be accessed using the `Count` property and the `Value` function.
Capacities	Collection	Contains the collection of capacity items for an agent. Each capacity item contains the following properties: `MediaType`, `CurrentInteractions`, `RoutableInteractions`, and `MaxInteractions` (see Table 13). To return this collection of capacity items, set the `EnableAgentCapacity [Workspace]` configuration option to `true`.

Table 12: Properties of the state Object for AgentGroupState

Property	Type	Description
type	String	Type of `state` object. In the 7.5.x release, extended current status information is provided only for `Agent` and `Agent Group` objects. Genesys recommends that you validate an object's `type` property as `AgentGroupState` before using other properties of this object. If the `state` object is not available, the value of the `type` property is `undefined`. In this case, all other properties of the `state` object will be unavailable, which may result in errors.
StartTime	OLE Date	UTC date and time when the current status was established. Suitable for use in the `Date` object that is available in the JScript.
AgentGroupID	String	Name of the agent group.
Capacities	Collection	Contains the collection of capacity items for an agent group. Each capacity item contains the following properties: `MediaType`, `CurrentInteractions`, `RoutableInteractions`, and `MaxInteractions` (see Table 13). To return this collection of capacity items, set the `EnableAgentGroupCapacity [Workspace]` configuration option to `true`.

Table 13 provides a description of the properties of the Capacities items.

Table 13: Properties of the Capacity Items

Property	Type	Description
MediaType	String	The name of the media.
CurrentInteractions	Integer	The number of current interactions.
RoutableInteractions	Integer	The number of available routable interactions.
MaxInteractions	Integer	The maximum number of interactions.

The formula script can count the Capacities collection by using either the enumerator or the index (for example, by using `.value(index)` method). The number of capacity items are available through the `.Count` property. Certain capacity items for specific media are also available by using `.value("media_name")`; for example, `var voice_capacity = state.Capacities.Values("voice")`. Refer to Example 2 and Example 3.

Example 2

The following is an example of the formula script for `Agents`:

```
if(state.type == "AgentState")
{
      var i, item, s = "";
      for(i = 0; i < state.Capacities.Count; i++)
      {
          item = state.Capacities.Value(i);
          s += item.MediaType;
          s += "(" + item.CurrentInteractions + "," +
item.MaxInteractions + "," + item.RoutableInteractions + "),";
      }
      result.Text = s;
}
else
      result.Text = "n\\a";
```

Example 3

The following is an example of the formula script for `Agent Groups`:

```
if(state.type == "AgentGroupState")
{
      var i, item, s = "";
      for(i = 0; i < state.Capacities.Count; i++)
      {
          item = state.Capacities.Value(i);
```

```
            s += item.MediaType;
            s += "(" + item.CurrentInteractions + "," +
item.MaxInteractions + "," + item.RoutableInteractions + "), ";
      }
      result.Text = s;
}
else
      result.Text = "n\\a";
```

Note: In the CCPulse+ workspace, the `Graph` view displays only one number —current capacity per media.

DN-Related Information

To obtain DN-related information in a formula, use the `DN` predefined object with its properties listed in Table 14. This object can only be used in conjunction with the `state` predefined object.

Table 14: Properties of the DN Object

Property	Type	Description
type	String	This string contains the value `DNState`.
DNType	String	The directory number's type, such as `Position`, `Extension`, or `Chat`.
DN	String	The directory number.
StartTime	Long	Greenwich Mean Time (GMT) date and time when current status has been established. Suitable for use in the `Date` object available in the JScript.
Status	String	Current status name of the directory number as provided by Stat Server.
SwitchID	String	The switch's identification number, as registered in Configuration Server, to which the DN belongs.
Actions	Collection	Contains information about the list of actions currently occurring at a DN. DNs can hold several simultaneous actions. The list can be iterated by using `Enumerator` object (available in Jscript) to obtain specific agent data. Refer to the `GetAction` function in Example 4 for an example showing how to use this property.

Action-Related Information

To obtain action-related information use the `Action` predefined object and its properties described in the Table 15. This object can only be used in conjunction with DN predefined object. Please note that the names of masks can coincide with the names of statuses.

Table 15: Properties of the Action Object

Property	Type	Description
type	String	This string contains the value `Action`.
Action	String	Name of the action, such as those listed on page 76 without the `SDNA`, `SRPA`, or `SCMPA` prefixes.
StartTime	Long	GMT date and time when this action has been established. Suitable for use in the `Date` object available in the JScript.

Example 4

The following example illustrates how you can use the `state` predefined object within a customized statistic's formula to determine an agent's current state:

```
result.Text = GetAgentState();
function GetAgentState()
{
     if(state.type != "AgentState")
                return "n/a";
     var r = "(" + FormatDate(state.StartTime) + ") ";
     r += state.AgentID;
     r += " [Place: " + state.PlaceID;
     r += " Login: " + state.LoginID + "]";
     r += ", Status: " + state.Status;
     r += "; DNs: ";

     for(var e = new Enumerator(state.DNs); !e.atEnd();
e.moveNext())
     {   r += GetDNState(e.item());   }
     return r;
}

function FormatDate(dateVal)
{
     var dateObj = new Date(dateVal);
     return dateObj.getHours() + ":" + dateObj.getMinutes() + ":"
             + dateObj.getSeconds();
}

function GetDNState(dn)
{
     var r = "(" + FormatDate(dn.StartTime) + ") ";
     r += dn.DN;
     r += " [Switch: " + dn.SwitchID;
     r += ", Type: " + dn.DNType + "], Status: " + dn.Status;
```

```
    r += "; Actions: "

    for(var e = new Enumerator(dn.Actions); !e.atEnd();
e.moveNext())
    {   r += GetAction(e.item()) + " ";   }
    return r;
}

function GetAction(a)
{
    var r = "(" + FormatDate(a.StartTime) + ") ";
    r += a.Action;
    return r;
}
```

The information that CCPulse+ returns is the same information that appears in the `Extended Current Status` window when requested for `Agent` objects. Figure 11 illustrates sample data returned in the Example 4 script. Note that the time information returned from running the script reflects startup time (the time that the current status occurred)—not duration as shown in the `Extended Current Status` window.

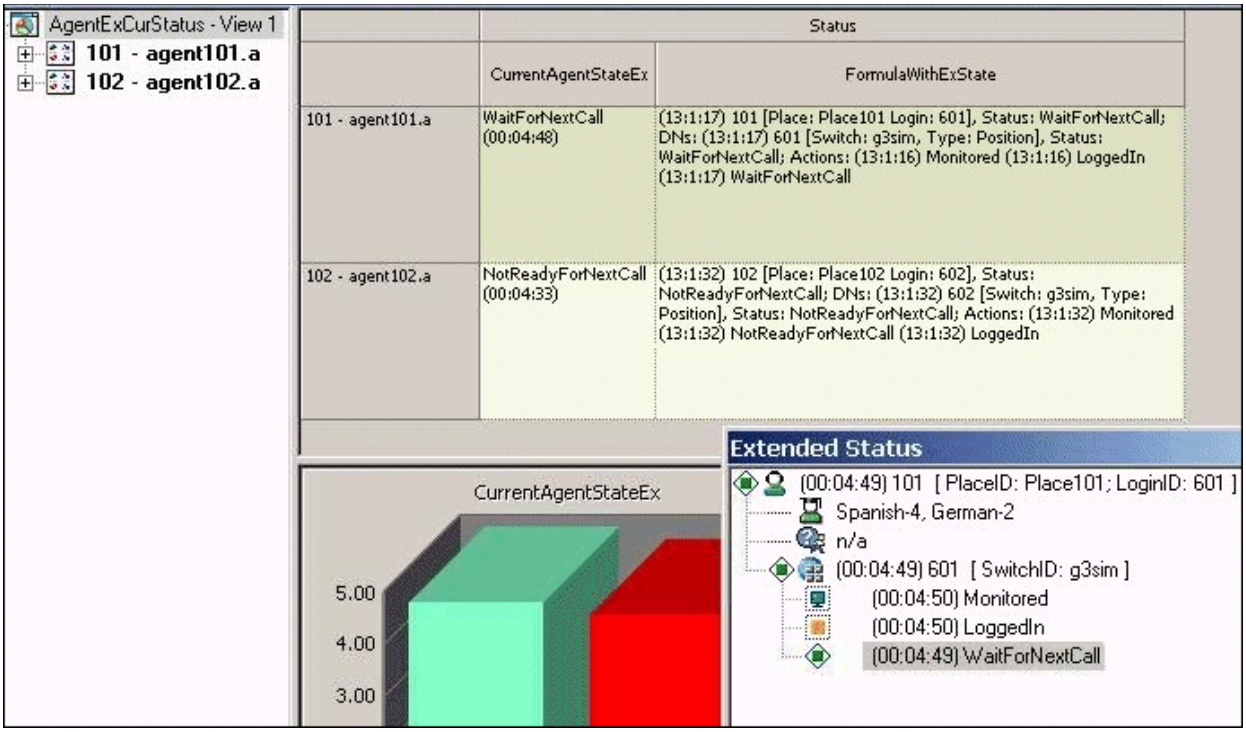

Figure 11: Example 4 Results.

Formatting the Results

CCPulse+ enables you to specify the meaning of the result CCPulse+ returns for custom statistics:

- `result.Duration = formula`
  Displays the returned result in `hh:mm:ss` format.
- `result.Long = formula`
  Displays the returned result in integer format.
- `result.Float = formula`
  Displays the returned result in floating-point format.
- `result.Text = formula`
  Displays the returned result in string format.

If you omit the `result` object from the formula, CCPulse+ might not return the result as you expect it. For example, if you try to calculate duration based on a set of time statistics that you create, and if you do not use `result.duration` in your formula, CCPulse+ might return a `Long` value and not apply the `hh:ss:mm` or other time format.

Figure 10 on page 62 illustrates the use of `result.Long` in the definition of the `All Entered` statistic.

Displaying Capacity Information

CCPulse+ provides two options for formatting the Capacities collection of statistics in the table and graph views.You can:

- Enter a user-defined format string (see Table 16).
- Select a predefined format (see Table 17).

To chose either option, from the `Statistic` dialog window in the CCPulse+ Template Wizard:

- Select the media for which CCPulse+ will display the capacity information.

  Note: The list of available media is populated by the values set for the `MediaTypes` configuration option. See Chapter 2 for more information on this option.

- Select the media format from the drop-down menu, or enter a user-defined format.

Figure 12 shows the predefined formats in a cutaway of the `Statistic` dialog box.

Note: Capacity information is available in the `CurrentAgentTargetState` or `CurrentAgentGroupTargetState` statistic (`CurrentTargetState` statistical category).

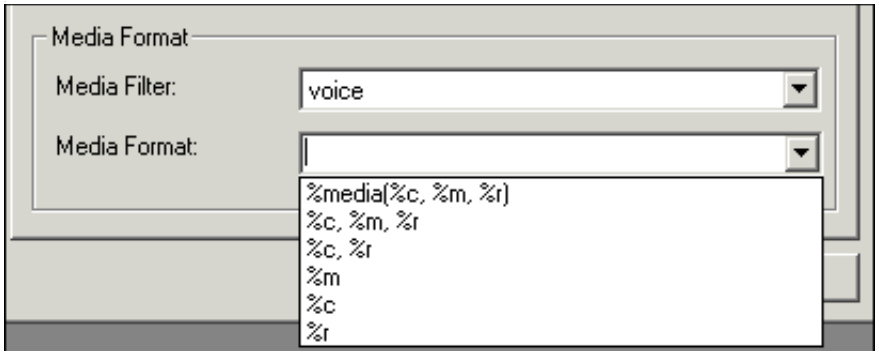

Figure 12: Cutaway of the Statistic Dialog Box

Table 16 lists the placeholder values for which you can substitute a user-defined sting. Note that you must preface the strings entered with a `%` sign.

Table 16: User-Defined Format Strings for Capacity Information

Placeholder	Type	Description
%media	String	Displays the name of the media in a single cell.
%m	Single integer	Displays the maximum interactions in a single cell.
%c	Single integer	Displays the current interactions in a single cell.
%r	Single integer	Displays routable interactions in a single cell.

Table 17 on page 70 lists the predefined formats you can select from a drop-down menu of format choices.

Table 17: Predefined Formats for Capacity Information

Format	Example	Description
%media(%c, %m, %r)	voice(0, 3, 3)	Displays media name, current interactions, maximum interactions, and routable interactions in a single cell. This is the default format.
%c, %m, %r	0, 3, 3	Displays current interactions, maximum interactions, and routable interactions in a single cell delimited by commas.
%c, %r	0, 3	Displays current and routable interactions in a single cell delimited by commas.
%c	0	Displays the current interactions.
%r	3	Displays the routable interactions.
%m	3	Displays the maximum interactions.

Chapter

6 Thresholds and Actions

By linking thresholds and actions, you can configure CCPulse+ so that it performs a specific action whenever the criteria that you set in a threshold definition are met. This chapter contains the following sections:

- Overview, page 71
- Predefined Objects, page 72
- Predefined Thresholds, page 77
- Predefined Actions, page 84

Overview

This chapter presents the predefined:

- Objects that you can use in the threshold and action VBScripts to create and edit thresholds and actions.
- Thresholds that are installed with CCPulse+.
- Actions that are installed with CCPulse+.

All existing thresholds and actions that are applied to the root level of a view are propagated automatically to any new objects that are added to the view. You do *not* have to apply thresholds manually to statistics of newly added objects.

You can add a customized threshold to a statistic of a specific contact-center object, even if there is a pre-existing threshold applied at the root level. In this case, the customized threshold will take precedence. Any changes to the root-level threshold will not affect a statistic with a customized threshold. A new icon will designate the associated customized threshold.

For instructions on using the Threshold Wizard and Action Wizard to create and edit thresholds and actions, and for information on associating thresholds and actions for specific contact-center objects, see the *Reporting 7.5 CCPulse+ Help*.

Predefined Objects

Use the predefined objects listed and described in Table 18 to modify the predefined CCPulse+ thresholds and actions or to create new ones using the CCPulse+ Threshold and/or Action wizards.

Table 18: Predefined Objects for Use in Threshold and Action Scripts

Object	Type	Description
Threshold Use this object to set and retrieve the data used in threshold calculation, which is available in both threshold and action scripts.		
Result	Boolean	Contains the result of the execution of a threshold script. If the result is not set in the script, it is assumed to be `False`. Example: `Threshold.Result = True`
StatValue	Variant	Contains the current value of the statistic to which the threshold script is applied. Example: `If (Threshold.StatValue = x) then ...`
StatAlias	String	Alias of the statistic, as defined in the template. **Note:** The alias is an empty string if a threshold is applied to a configuration object in the main tree.
ActionDelay	Integer	Used to delay action execution by *n* seconds. If the threshold script that set this value is executed again before its actions are completed, and the result is `False`, action execution is canceled. This allows for so-called duration thresholds, the actions of which are executed if a certain condition is `True` for the amount of time specified. Example scenario: The value of a statistic changes. CCPulse+ executes the `Threshold1` script, which contains an action delay of 10 seconds. The script returns a `True` value. You can expect CCPulse+ to execute the associated actions 10 seconds later. • If the value of the same statistic changes 5 seconds later, CCPulse+ re-executes the `Threshold1` script. If the script returns `False`, CCPulse+ cancels execution of the delayed actions. • If there is no statistic changes before the 10 seconds elapse, CCPulse+ executes the actions, thereby showing that a particular condition was `True` for the duration of 10 seconds. Example: `Threshold.ActionDelay = 10`

Table 18: Predefined Objects for Use in Threshold and Action Scripts (Continued)

Object	Type	Description
ActionDelay (continued)		**Note:** ActionDelay functionality is based on the duration of Current Status only—not Current Status and Reason together, unless status and reason are coupled together at the moment of an actual change in Current Status. To obtain the correct time duration when Current Status and Reason are coupled together, you can configure your statistic to use the `CurrentTime` statistical category with a specific status mask that is filtered by a specific reason.
StatReasons	String	Use this read-only property to evaluate reasons in threshold and action scripts. The following threshold script illustrates how you can use this property when it is applied to an agent `CurrentState` statistic: `If (Threshold.StatValue = StatAction.SDNANotReadyForNnextCall and Threshold.StatReasons = "Lunch") then Threshold.Result = True end if` Note that reasons and action codes can appear only for following agent states: • `LoggedIn` • `NotReadyForNextCall` • `AfterCallWork` • `WaitForNextCall` Reasons can originate from more than one source (T-Server, hardware, or other), and they can be provided to CCPulse+, via Stat Server, in string or integer format. CCPulse+ converts all reasons to string format, separating each with a semicolon (;). To effectively evaluate the concatenated string, the CCPulse+ Administrator might have to parse it using VBScript.
Configuration Object Properties These represent the primary properties of any configuration object (also available in action scripts).		
CFGTenantDbid	Long	tenant dbid Example: `Threshold.CFGTenantDbid`
CFGDbid	Long	object dbid Example: `Threshold.CFGDbid`
CFGObjectID	String	object ID Example: `Threshold.CFGObjectID`

Table 18: Predefined Objects for Use in Threshold and Action Scripts (Continued)

Object	Type	Description
CFG*Type*	Integer	Type of object (`short CfgObjectType` enumeration value, for instance, value `4` means `CFGPlace`). The enumeration values for the available object types are as follows: • `Switch–1` • `DN–2` • `Person–3` • `Place–4` • `AgentGroup–5` • `PlaceGroup–6` • `Tenant–7` • `Script–12` • `DNGroup–17` • `CallingList–26` • `Campaign–27` Example: `Threshold.CFGDNGroup` CCPulse+ support for the `CFGScript` predefined object was introduced in the 7.0.1 release.
CCPulseNotifier Use this object to send notifications to CCPulse+ (also available in action scripts).		
SetColor	Color: Integer	Notifies CCPulse+ of a color change. A member of the Color object can be passed as a parameter. Example: `CCPulseNotifier.SetColor(Color.Red).` Or a user-defined RGB value can be passed instead. Example: `CCPulseNotifier.SetColor(RGB(50,50,50)).` For further information about the RGB function, consult your Visual Basic Script (VBScript) documentation. **Note:** You must set color commands for color names using the English language.
ResetColor	N/A	Notifies CCPulse+ that the color of the object (set by `SetColor`) should be reset to its original color. Example: `CCPulseNotifier.ResetColor`
Hide	N/A	Use this operation to hide a configuration object in the `Call Center Objects` pane or `Tree` view, depending on the threshold application.
Show	N/A	Use this operation to show a configuration object on the `Call Center Objects` pane, the `Tree` view, and the `Graph` view. Used to display an object that was previously hidden using the `Hide` action associated with a different threshold.

Table 18: Predefined Objects for Use in Threshold and Action Scripts (Continued)

Object	Type	Description
Color (Enumeration Object) The properties of this object are constant RGB values that you can use with the `SetColor` and `ResetColor` methods (also available in action scripts). Some commonly used color codes are listed below. Examples: `CCPulseNotifier.SetColor(Color.Red)` or `CCPulseNotifier.SetColor(RGB(255,0,0))` The following commands (and their corresponding RGB values) are available with the `Color` object: Red RGB (255,0,0) Blue RGB (0,0,255) Green RGB (0,255,0) DarkRed RGB (128,0,0) DarkGreen RGB (0,128,0) Cyan RGB (0,255,255) DarkCyan RGB (0,128,128) Magenta RGB (255,0,255) DarkMagenta RGB (128,0,128) Yellow RGB (255,255,0) White RGB (255,255,255) Black RGB (0,0,0) Gray RGB (192,192,192) DarkGray RGB (128,128,128) **Note:** You must set color commands for color names using the English language.		

Table 18: Predefined Objects for Use in Threshold and Action Scripts (Continued)

Object	Type	Description
StatAction (Enumeration Object) The properties of this object are the names of the DNs, route points, and campaign actions as described in the statistics library (available only in threshold scripts). Example: `if ( Threshold.StatValue = StatAction.SDNALoggedIn ) then ...`		
Current Status Name	**Current State Value**	

SDNAWaitForNextCall	4	Each Current Status has an associated Current State value which is constant.
		The prefixes in the listing to the left mean the following:
		DNA—Directory Number Action
		RPA—Routing Point Action
		CMPA—Campaign Action
		The *S* prefix on each indicates that the item is a Stat Server entity (rather than an entity of Configuration Server or any other server). Elsewhere within the CCPulse+ interface, these actions appear without these prefixes.
		Example: `if (Threshold.StatValue=StatAction.SDNANotMonitored) then ...`
		`Threshold.Result=True`
		`end if`
		CCPulse+ executes the corresponding action script when the selected DN statistic achieves `NotMonitored` status.
		Note: *ASM* is a reference to the Active Switching Matrix call model used in conjunction with predictive dialing in the Outbound Contact solution. Refer to Genesys Outbound Contact documentation for further details.

SDNACallDialing	6	
SDNACallRinging	7	
SDNANotReadyForNextCall	8	
SDNACallUnknown	18	
SDNACallConsult	19	
SDNACallInternal	20	
SDNACallOutbound	21	
SDNACallInbound	22	
SDNALoggedOut	23	
SDNACallOnHold	13	
SDNANotMonitored	0	
SDNAMonitored	1	
SDNALoggedIn	2	
SDNAOnHook	3	
SDNAOffHook	5	
SDNAAfterCallWork	9	
SRPANotMonitored	0	
SRPAMonitored	1	
SRPANotReadyForNextCall	2	
SDNAASM_Engaged	16	
SDNAASM_Outbound	17	
SRPAWaitForNextCall	13	
SRPACallWait	9	
SCMPAStatusDeactivated	0	
SCMPAStatusActivated	1	
SCMPAStatusRunning	2	

Predefined Thresholds

This section describes the following threshold scripts which are provided with each Genesys solution:

- AgentLoggedIn
- AgentLoggedOut
- CampaignActivated
- CampaignDeactivated
- ExactValue
- ExactValueWDuration
- GreaterThanLimit
- GreaterThanLimitWDuration
- LessThanLimit
- LessThanLimitWDuration
- LessThanSpecifiedLimit
- NotEqual
- NotEqualWDuration
- NotReadyForNextCall
- OutOfRange
- OutOfRangeWDuration
- WithinRange
- WithinRangeOfValues
- WithinRangeWDuration

The purpose of providing these scripts is to demonstrate how you might apply thresholds within your own CCPulse+ environment.

The scripts and threshold definitions are identical for each Genesys solution and are stored in a file that is named `thresholds.stg`.

Note: Because these are sample scripts, it is likely that their absolute values have little significance within your environment. You are encouraged to modify them to more meaningful values.

How to Use Thresholds

The following discussion assumes that you are familiar with the VBScript programming language and with the CCPulse+ predefined objects that are described in "Predefined Objects" on page 72.

To be effective, you must use thresholds in conjunction with actions. When using thresholds and actions, keep in mind the following points:

- CCPulse+ executes associated actions only if the threshold returns a `True` result.

  This means that you cannot create a threshold to perform one action if its result is `True` and another action if its result is `False`.
- With CCPulse+, you can apply more than one threshold to a statistic.

  You can have one threshold call the first action, the second threshold call the second action, and so on.
- You can apply one or more actions to a particular threshold.
- You can apply a chain of thresholds to a particular statistic.

CCPulse+ performs the following actions when a statistic value changes:

a. CCPulse+ calculates all appropriate thresholds.

b. CCPulse+ checks if any thresholds for the previous statistic value result in a `false` value, in which case CCPulse+ cancels the respective actions.

c. CCPulse+ checks if any thresholds for the changed statistic value result in a `true` value, in which case CCPulse+ executes all the expected actions.

If there are overlapping conditions (that is, more than one condition is true), then they are all evaluated and their actions are executed. To ensure that only one action gets triggered, create thresholds without overlapping conditions.

- You can also use object linking and embedding (OLE) automation objects within threshold scripts, because VBScript allows this.

Note: It is *not* recommended to use OLE objects with either a GUI-like appearance or with dialog boxes that require explicit user confirmation.

And, starting with release 7.2, you can configure the `SafeScriptingMode` option to specify whether CCPulse+ spawns a separate thread to execute VBScripts. Refer to “SafeScriptingMode” on page 42 for more information about this option.

Real-World Example

Consider the following business scenario, which uses thresholds and actions to aid a CCPulse+ operator in managing a queue named `ClientQueue`.

Management has decided that 5 calls or fewer waiting in `ClientQueue` is the desired situation; 10 or more calls waiting represents an alarming condition; and anywhere in between is acceptable.

Three custom thresholds, `UnderCapacity`, `AtCapacity`, and `OverCapacity`, have been applied to the `CallsWaiting` statistic, which is a statistic that is monitored for `ClientQueue`. The custom thresholds are defined as follows:

UnderCapacity

```
if Threshold.StatValue < 5 then
   Threshold.Result = true
end if
```

AtCapacity

```
if Threshold.StatValue >= 5 and Threshold.StatValue < 10 then
   Threshold.Result = true
```

```
end if
```

OverCapacity

```
if Threshold.StatValue >= 10 then
   Threshold.Result = true
end if
```

A color action is assigned to each threshold to change the background color of a table cell holding the `CallsWaiting` statistic when CCPulse+ evaluates the threshold to be `True`. The `UnderCapacity` threshold triggers the `SetGreen` action. `AtCapacity` triggers the `SetYellow` action, and `OverCapacity` triggers the `SetRed` action.

Now CCPulse+ users have a visual cue to quickly assess `ClientQueue` performance.

Explanations of the Predefined Thresholds

For each predefined threshold, this section gives the VBScript and the object types to which you can apply it.

Note: Some of these scripts include numbers by default. These numbers are intended simply to indicate where to insert a value in the script. You must replace these numbers with ones suited to your environment.

AgentLoggedIn

This threshold script returns a `True` value when the agent is logged in—that is, when CCPulse+ receives the `Logged In` status from Stat Server.

```
if Threshold.StatValue = StatAction.SDNALoggedIn then
   Threshold.Result = true
end if
```

Used with the `Agent` object.

Note: Stat Server reports to its clients the highest ranking status of an object. After an agent logs in, his/her status almost immediately changes from `Logged In` to `Ready` or `Not Ready`, which holds a higher ranking. Hence, the threshold script above is rarely tripped even though an agent may be logged in for quite some time. For this reason, Genesys recommends that you revise your script to reflect the condition when agents have not logged out:

```
if Threshold.StatValue <> StatAction.SDNALoggedOut then
   Threshold.Result = true
end if
```

AgentLoggedOut

The threshold returns a `True` value when the agent is logged out.

```
if Threshold.StatValue = StatAction.SDNALoggedOut then
   Threshold.Result = true
end if
```

Used with the `Agent` object.

Using the `Hide` action with this threshold removes the logged out agent from the `Table` view as well as the `Tree` view.

Note: This functionality does not apply to the `Graph` view.

CampaignActivated

This threshold script returns a `True` value when the `Campaign` object has been activated.

```
if Threshold.StatValue = StatAction.SCMPAStatusActivated then
   Threshold.Result = true
end if
```

Used with the `Campaign` object.

CampaignDeactivated

This threshold script returns a `True` value when the `Campaign` object has been deactivated.

```
if Threshold.StatValue = StatAction.SCMPAStatusDeactivated then
   Threshold.Result = true
end if
```

Used with the `Campaign` object.

ExactValue

This threshold script returns a `True` value if the value of the statistic to which the threshold is applied equals the value you set. By default, the value is set to `3`.

```
if Threshold.StatValue = 3 then
   Threshold.Result = true
else
   Threshold.Result = false
end if
```

ExactValueWDuration

This threshold script returns a `True` value if the value of the statistic to which the threshold is applied equals the value you set. By default, this value is set to

3. If `True`, CCPulse+ executes the associated action a specified number of seconds later. By default, `ActionDelay` is set to `10`.

```
if Threshold.StatValue = 3 then
   Threshold.ActionDelay = 10
   Threshold.Result = true
else
   Threshold.Result = false
end if
```

GreaterThanLimit

This threshold script returns a `True` value if the value of the statistic to which the threshold is applied is greater than the value you set. By default, this value is set to `10`.

```
if ( Threshold.StatValue > 10) then
    Threshold.Result = True
else
    Threshold.Result = False
end if
```

GreaterThanLimitWDuration

This threshold script returns a `True` value if the value of the statistic to which the threshold is applied is greater than the value you set. By default, this value is set to `10`. If `True`, CCPulse+ executes the associated action a specified number of seconds later. By default, `ActionDelay` is set to `10`.

```
if Threshold.StatValue > 10 then
   Threshold.ActionDelay = 10
   Threshold.Result = True
else
   Threshold.Result = False
end if
```

LessThanSpecifiedLimit

This threshold script returns a `True` value if the value of the statistic to which the threshold is applied is less than the value you set. By default, this value is set to `777`.

```
if (Threshold.StatValue < 777) then
    Threshold.Result = True
else
    Threshold.Result = False
end if
```

Used with the `Calling List` object.

LessThanLimit

This threshold script returns a `True` value if the value of the statistic to which the threshold is applied is less than the value you set. By default, this value is set to `10`.

```
if ( Threshold.StatValue < 10 ) then
    Threshold.Result = True
else
    Threshold.Result = False
end if
```

LessThanLimitWDuration

This threshold script returns a `True` value if the value of the statistic to which the threshold is applied is less than the value you set. By default, this value is set to `10`. If `True`, CCPulse+ executes the associated action a specified number of seconds later. By default, `ActionDelay` is set to `10`.

```
if Threshold.StatValue < 10 then
   Threshold.ActionDelay = 10
   Threshold.Result = True
else
   Threshold.Result = False
end if
```

NotEqual

This threshold script returns a `True` value if the value of the statistic to which the threshold is applied equals any nonnull value other than the value you set. By default, this value is set to `3`.

```
if Threshold.StatValue <> 3 then
   Threshold.Result = true
else
   Threshold.Result = false
end if
```

NotEqualWDuration

This threshold script returns a `True` value if the value of the statistic to which the threshold is applied equals any nonnull value other than the value you set. By default, this value is set to `3`. If `True`, CCPulse+ executes the associated action a specified number of seconds later. By default, `ActionDelay` is set to `10`.

```
if Threshold.StatValue <> 3 then
   Threshold.ActionDelay = 10
   Threshold.Result = true
else
   Threshold.Result = false
```

```
end if
```

OutOfRange

This threshold script returns a `True` value if the value of the statistic to which the threshold is applied lies outside the range of values between the values you set. By default, these values are set to `5` and `10`.

```
if Threshold.StatValue < 5 Or Threshold.StatValue > 10 then
   Threshold.Result = true
else
   Threshold.Result = false
end if
```

OutOfRangeWDuration

This threshold script returns a `True` value if the value of the statistic to which the threshold is applied lies outside the range of values between the values you set. By default, these values are set to `5` and `10`. If `True`, CCPulse+ executes the associated action a specified number of seconds later. By default, `ActionDelay` is set to `10`.

```
if Threshold.StatValue < 5 Or Threshold.StatValue > 10 then
   Threshold.ActionDelay = 10
   Threshold.Result = true
else
   Threshold.Result = false
end if
```

WithinRange

This threshold script returns a `True` value if the value of the statistic to which the threshold is applied lies exclusively between the values you set. By default, these values are set to `5` and `10`.

```
if Threshold.StatValue > 5 and Threshold.StatValue < 10 then
   Threshold.Result = true
else
   Threshold.Result = false
end if
```

WithinRangeOfValues

This threshold script returns a `True` value if the value of the statistic to which the threshold is applied lies exclusively between the values you set. By default, these values are set to `50` and `1000`.

```
if Threshold.StatValue > 50 and Threshold.StatValue < 1000 then
   Threshold.Result = true
else
   Threshold.Result = false
```

```
end if
```

WithinRangeWDuration

This threshold script returns a `True` value if the value of the statistic to which the threshold is applied lies exclusively between the values you set. By default, these values are set to `5` and `10`. If `True`, CCPulse+ executes the associated action a specified number of seconds later. By default, `ActionDelay` is set to `10`.

```
if Threshold.StatValue > 5 and Threshold.StatValue < 10 then
   Threshold.ActionDelay = 10
   Threshold.Result = true
else
   Threshold.Result = false
end if
```

Predefined Actions

Actions enable CCPulse+ to alert you when a particular condition has been met with respect to a statistic's value. This section describes the following action scripts which are provided with each Genesys solution:

- `ActiveWebBrowser`
- `ExecuteProcess`
- `Hide`
- `PCSpeakerAudioAlert`
- `ResetColor`
- `SendE-MailWithCDO`
- `SetColor`
- `Show`
- `state`
- `WAVAudioAlert`

This section provides the information that you need to customize the action scripts to meet your contact-center requirements. The scripts are identical for each Genesys solution and are stored in a file that is named `actions.stg`.

Actions have little meaning outside the context of their associated thresholds. Therefore, review "Predefined Thresholds" on page 77 and see the "Threshold/Action Association" topic in the *Reporting 8.0 CCPulse+ Help*.

How Actions Function

The results of actions persist until CCPulse+ evaluates the result of the associated threshold to be `False`. For example, if you have used the `SetColor` action to apply a `Green` color when the value of `TotalCalls` is greater then `100`, the `Green` color persists until the value of `TotalCalls` falls below `100`.

In this example:

- `TotalCalls` is the targeted statistic.
- `GreaterThan100` is the applied threshold.
- `SetColorGreen` is the threshold's associated action.

The following discussion assumes you are familiar with the VBScript programming language and with the CCPulse+ predefined objects described in "Predefined Objects" on page 72.

Explanations of the Predefined Actions

In the descriptions that follow, the term *targeted statistic* refers to the statistic to which CCPulse+ applies the action's associated threshold.

ActivateWebBrowser

This action opens Microsoft Internet Explorer and navigates to the specified URL. Use the following script format:

```
Dim Browser
Set Browser =
CreateObject("InternetExplorer.Application.1")
Browser.Navigate("www.genesys.com")
Browser.Visible = True
```

ExecuteProcess

The sample that is provided uses a COM component that is defined in the `GsSampleTE Objects.DLL` file. The source code for this file is included in the CCPulse+ installation package. For this script to work, you must compile it by using Visual Basic.

Note: This sample is provided "as is" for your reference. Although every effort has been taken to ensure that this sample functions correctly, Genesys does not offer product support for it.

To create the Genesys File System object (`FSObject`) and execute a script, use the following script:

```
Set FSObject =
CreateObject("GsSampleTEObjects.GsActionFSObject")

FSObject.Execute("C:\Program Files\Plus!\Microsoft
Internet\iexplore.exe")
```

This example opens Microsoft Internet Explorer within Windows NT.

Hide

This action hides the object to which the action's associated threshold has been assigned on the `Call Center Objects` pane, in the `Tree` view, and in the `Grid` view (the row that contains the specified statistic). CCPulse+ hides the object when the conditions set for the associated threshold are `True`. The object remains hidden unless you configure a second threshold/action combination

for the targeted statistic using an alternative threshold condition and the `Show` action.

```
CCPulseNotifier.Hide
```

PCSpeakerAudioAlert

This action uses a COM component that is defined in the `GsSampleTEObjects.DLL` file. The source code for this file is included in the CCPulse+ installation package. For this script to work, you must compile it by using Visual Basic.

To create the Genesys Sound object, use the following script:

```
Set SoundObject =
   CreateObject("GsSampleTEObjects.GsActionSoundObject")
   SoundObject.Beep
```

ResetColor

This action sets to its default color the background color of the table cell that contains the targeted statistic.

```
CCPulseNotifier.ResetColor
```

SendE-MailWithCDO

Preinstall Microsoft Outlook 98 or Microsoft Exchange Server version 5.5 to execute the following script.

```
   Dim msgBody
   Dim thresholdInfo
   msgBody="A mail from the CCPulse+ threshold engine!"+vbNewLine
   thresholdInfo="StatValue:"+CStr(Threshold.StatValue)+vbNewLine+
   "StatAlias:"+CStr(Threshold.StatAlias)+vbNewLine+
   "CFGTenantDbid:"+CStr(Threshold.CFGTenantDbid)+vbNewLine+
   "CFGDbid:"+CStr(Threshold.CFGDbid) + vbNewLine +
   "CFGObjectID:"+CStr(Threshold.CFGObjectID)+vbNewLine+
   "CFGType:"+CStr(Threshold.CFGType)+vbNewLine
   SendMail "","",msgBody+thresholdInfo,"Alarm notification from
       CCPulse+"

   Sub SendMail(profile,recipient,message,subject)
       Dim objSession, objAddrEntry, oInbox, colMessages, oMessage,
       colRecipients, oRecipient

       Set objSession = CreateObject("MAPI.Session")
       objSession.Logon profile,"",True,False,0, False

       Set objAddrEntry = objSession.CurrentUser
       Set oInbox = objSession.Inbox
```

```
    Set colMessages = oInbox.Messages
    Set oMessage = colMessages.Add()
    Set colRecipients = oMessage.Recipients

    If recipient = "" Then
        colRecipients.Add  objAddrEntry
    Else
        colRecipients.Add  recipient
    End If
    colRecipients.Resolve

    oMessage.Subject = subject
    oMessage.Text = msg
    oMessage.Send

    objSession.Logoff
    Set objSession = nothing
End Sub
```

SetColor

This action sets to whatever color you choose—for example, dark red—the background color of the table cell that contains the targeted statistic.

```
CCPulseNotifier.SetColor( Color.DarkRed )
```

You can modify this action script to use any of the CCPulse+ predefined colors, or you can specify a custom color in RGB format—for example:

```
CCPulseNotifier.SetColor( RGB( 7, 77, 177 ) )
```

The available predefined colors are the following:

- `Blue`
- `Green`
- `Red`
- `White`
- `Black`
- `Gray`
- `DarkGray`
- `DarkRed,`
- `Yellow`
- `DarkYellow`
- `DarkGreen`
- `Cyan`
- `DarkCyan`
- `DarkBlue`
- `Magenta`
- `DarkMagenta`

Show

This action displays or makes reappear the targeted object in the `Call Center Objects` pane, in the `Tree` view, and in the `Graph` view (the row that contains the specified statistic). The object remains visible unless you configure a second threshold/action combination for the targeted statistic by using an alternative threshold condition and the `Hide` action.

```
CCPulseNotifier.Show
```

state

This action returns information about an agent's state at a particular DN including when the status began, the agent's ID, the login ID, and the status that occurs at the DN:

```
result.Text = GetAgentState();
function GetAgentState()
{
if(state.type != "AgentState")
return "n/a";
var r = "(" + FormatDate(state.StartTime) + ") ";
r += state.AgentID;
r += " [Place: " + state.PlaceID;
r += " Login: " + state.LoginID + "]";
r += ", Status: " + state.Status;
r += "; DNs: ";
for(var e = new Enumerator(state.DNs); !e.atEnd(); e.moveNext())
{ r += GetDNState(e.item()); }
return r;
}

function FormatDate(dateVal)
{
var dateObj = new Date(dateVal);
return dateObj.getHours() + ":" + dateObj.getMinutes() + ":"
+ dateObj.getSeconds();
}

function GetDNState(dn)
{
var r = "(" + FormatDate(dn.StartTime) + ") ";
r += dn.DN;
r += " [Switch: " + dn.SwitchID;
r += ", Type: " + dn.DNType + "], Status: " + dn.Status;
r += "; Actions: "
for(var e = new Enumerator(dn.Actions); !e.atEnd(); e.moveNext())
{ r += GetAction(e.item()) + " "; }
return r;
}

function GetAction(a)
{
var r = "(" + FormatDate(a.StartTime) + ") ";
r += a.Action;
return r;
}
```

WAVAudioAlert

You can implement COM objects by using Visual Basic or C++. The sample that is provided uses a COM component that is defined in the `GsSampleTEObjects.DLL` file. The source code for this file is included in the CCPulse+ installation package. For this script to work, you must compile it by using Visual Basic.

Note: This sample is provided "as is" for your reference. Although every effort has been taken to ensure that this sample functions correctly, Genesys does not offer product support for it.

To create the Genesys Sound object, use the following script:

```
Set SoundObject =
CreateObject("GsSampleTEObjects.GsActionSoundObject")
CreateObject("GsSampleTEObjects.GsActionFSObject")
```

Specify the full path to a `.WAV` file in order to play it—for example:

```
SoundObject.PlaySoundFile ("c:\winnt\media\tada.wav")
```

Genesys

Chapter

7 Measuring CCPulse+/Stat Server Traffic

If your environment places CCPulse+ and Stat Server on different local area networks (LANs), you might be interested in determining the impact of CCPulse+/Stat Server traffic on your wide area network (WAN). This chapter shows how you can approximate CCPulse+/Stat Server traffic flow and includes the following sections:

- Determining Factors, page 91
- The Number of Requests, page 91
- The Number of Responses, page 92
- The Size of One Request/Response, page 92

Determining Factors

Traffic between CCPulse+ and Stat Server is dependent on many factors, including:

- The speed of both computers.
- The number of requests and responses in the data stream.
- Network speed—for example, a network operating at 100 Mbits/second might yield poor performance.

The data stream is dependent on how fast your network operates. Obviously, a fast network prevents many traffic problems.

The Number of Requests

The traffic specifically from CCPulse+ to Stat Server depends on:

- The number of statistics requested by CCPulse+ views.

- The number of monitored agents, agent groups, places, place groups, queues, groups of queues, routing points, and groups of routing points in the CCPulse+ workspace. CCPulse+ issues one current state request for each.

Moreover, if you request the extended current status for particular object, CCPulse+ requests current state information for the entire status tree of that object which includes subordinate objects. For the extended current status of an agent group, for example, CCPulse+ also receives the current state for all agents, places, DNs, and DN actions within that group. Thus, any formula measuring the number of requests is dependent on the content of a particular workspace and the objects for which you requested the extended current status.

The Number of Responses

The traffic specifically from Stat Server to CCPulse+ is dependent on the notification mode specified for each statistic along with each request. There are four types of notification modes:

- Change-based—Stat Server sends the current value whenever the value of the statistic significantly changes. A statistic's `Insensitivity` setting determines what a significant change in value is. Higher `Insensitivity` settings typically reduce network traffic, but they also reduce reporting accuracy because values are not updated as frequently.
- Time based—Stat Server sends the current value every *x* seconds, in accordance with the specified notification frequency. Lower notification frequencies reduce network traffic, but they also affect the availability of data in real time. For instance, instead of having updates every 2 seconds, a lower frequency specification may provide updates every 60 seconds.
- Reset based—Stat Server sends the current value right before setting it to `0`.
- No notification—Stat Server sends a response only when you issue a `Peek Statistic` request.

Depending on which notification mode you define for each statistic, Stat Server might send no response, one response, or constant responses. In addition to these four notification modes, the user can request updates at any time by using the Peek Statistics feature. Refer to the *Framework 8.0 Stat Server User's Guide* for further information about statistics.

The Size of One Request/Response

Except for extended current status requests and responses, the size of each statistical request and response is fixed and significantly less than 256 bytes. The response size is variable for extended current status requests and depends on the object type and subject parameter of the request.

Chapter

8 Troubleshooting

This chapter presents some scenarios that you might have to troubleshoot while using CCPulse+ and suggests possible solutions for each.

If you are experiencing difficulties with your CCPulse+ application, browse the following list for the resolutions to some commonly encountered issues.

If you continue to have problems after reviewing this chapter and verifying that your hardware is functioning properly and that your CCPulse+ configuration settings are correct, contact Genesys Customer Care.

These issues are covered in this chapter:

- Agents Having Trouble Logging In, page 93
- Wizards Are Disabled, page 94
- Import/Export Utility Is Disabled, page 94
- No DB Server Connection Error Message, page 95
- Agents Missing from Call Center Objects Pane, page 95
- General SQL Server Error Message, page 96
- Agents Cannot Monitor Agent Activity, page 96
- Sudden Stat Server Disconnection, page 96
- Stat Server Missing from New Workspace, page 97
- No Historical Associations Error Messages, page 97
- Cannot Recover Workspace, page 97
- Can't View Multi-Tenant Data, page 98

Agents Having Trouble Logging In

Confirm that the user name and password are correct, keeping in mind that both are case-sensitive. Confirm that the `Settings` fields on the CCPulse+ `Configuration Server` tab are entered correctly.

Wizards Are Disabled

In Configuration Manager, check that you have been assigned the rank of `Administrator`. CCPulse+ disables the Template, Threshold, and Action wizards for persons with no rank, or with the rank of `User`. Figure 13, for example, shows that `Administrator` ranking is being added to the properties of `Agent 7005`, who previously had no assigned ranking.

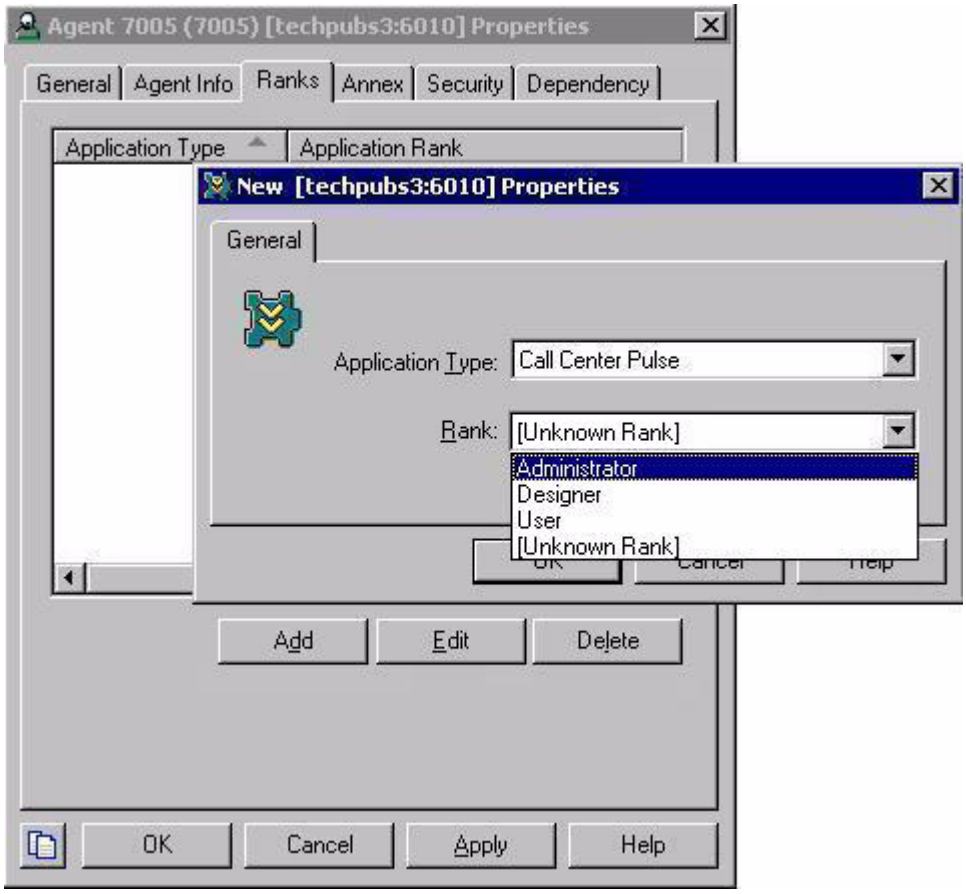

Figure 13: Adding Administrator Ranking to Agent Properties

Import/Export Utility Is Disabled

In Configuration Manager, check that you have been assigned the rank of `Administrator` (see Figure 13). CCPulse+ disables the Import/Export Utility for persons with no rank, or with the rank of `User`.

No DB Server Connection Error Message

If you receive the error message shown in Figure 14, verify in Configuration Manager that you have correctly configured the database access point to the Data Mart in the CCPulse+ `Application` object.

Figure 14: No DB Server Connection Error Message

Agents Missing from Call Center Objects Pane

When selecting object types in the Configuration Wizard, select the `IncludeAll (Type:Agent)` check box. See Figure 15 on page 95.

Be sure that agents have been set up properly in Configuration Manager. See the “Configuration Wizard” topic in *Reporting 7.5 CCPulse+ Help* for more information.

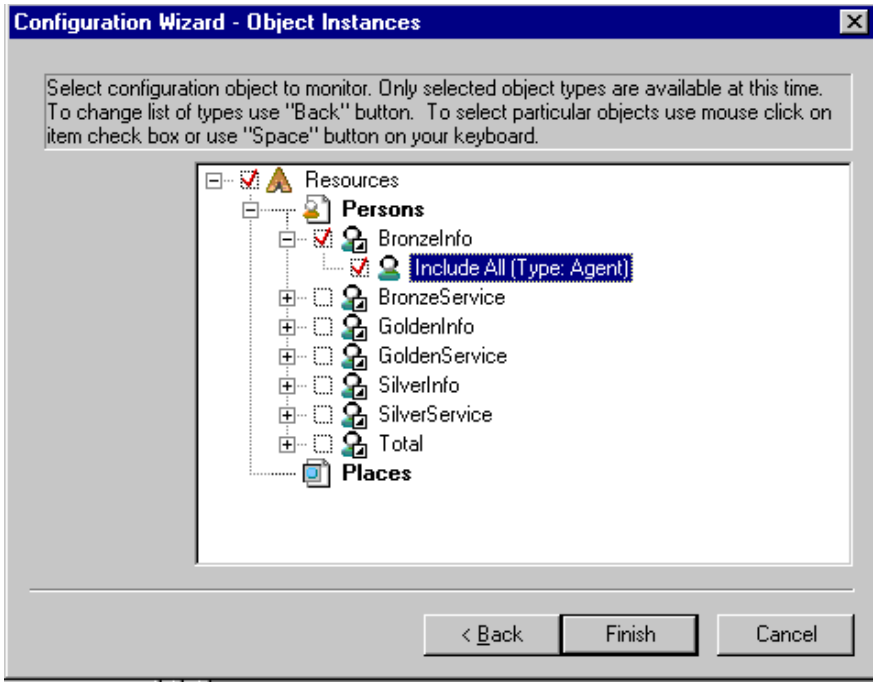

Figure 15: Object Instances

General SQL Server Error Message

"General SQL Server error: Check messages from the SQL Server. Invalid column name ' ... '."

The most probable cause of this error message is that your Data Mart has not been upgraded to release 6.5 or later. To use 6.5 (or later) CCPulse+ templates, you must upgrade all Reporting components to release 6.5 or later.

For more information, refer to the Reporting chapters of the *Genesys Migration Guide* or the "Configuring Genesys Solution Reporting" chapter of the *Reporting 8.0 Deployment Guide*.

Agents Cannot Monitor Agent Activity

Reinstall CCPulse+. At the Question dialog box, shown in Figure 16 on page 96, click Yes to enable operators to monitor agent statistics.

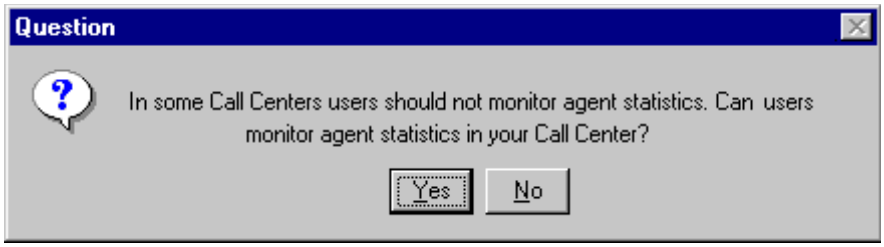

Figure 16: Allowing Users to Monitor Agent Statistics

Refer to "Installing Real-Time Reporting" in the *Reporting 7.5 Deployment Guide* for more information.

Sudden Stat Server Disconnection

Aberrant behavior may result when these events in CCPulse+ occur simultaneously:

- Stat Server is suddenly disconnected.
- Connection to the backup Stat Server cannot be established.
- The current workspace has been changed.

If the StatServer disconnected error message appears, click OK. When the Would you like to save your workspace? message appears, click one of the following:

- Yes, to save your current workspace but not your data.
- No, to clear both workspace and data.
- Cancel, to save the data but not the workspace.

Stat Server Missing from New Workspace

You must specify a Stat Server to which CCPulse+ is to connect. In Configuration Manager, check the `Connections` tab on the `Properties` dialog box of your CCPulse+ `Application` object to ensure the correct Stat Server is listed.

No Historical Associations Error Messages

If you are creating a historical view, and you receive the error message shown in Figure 17, verify that you have entered an associated historical statistic for each applicable real-time statistic you are using.

Figure 17: No Data Available Error Message

If you are trying to create a historical view, and you receive the error message in Figure 18, confirm that you have correctly configured your Data Mart database access point in the CCPulse+ `Application` object.

Figure 18: No Historical Associations Error Message

Cannot Recover Workspace

Under the very rare circumstance when your workspace becomes corrupt, CCPulse+ will not enable you to recover it. Instead, if you try to open the workspace, CCPulse+ displays the following error message:

```
Workspace [name] corrupted! It cannot be recovered. Please delete.
```

A workspace can become corrupt, for instance, if it was previously only partially saved as is the case when the related CCPulse+ process is killed

during the save operation or given file system problems. For this reason, Genesys recommends that you periodically back up your storage (*.stg) files.

Can’t View Multi-Tenant Data

The tenant to which a user belongs and the whether the user can view environment-level data is determined by how the user was originally defined to Configuration Server. CCPulse+ will display data for only those tenants that are monitored by the specific Stat Server application to which CCPulse+ is connected.

Supplements

Related Documentation Resources

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

Reporting

- *Reporting 8.0 CCPulse+ Help,* which describes how to navigate the CCPulse+ interface, operate the wizards, create templates and views, and associate actions to specific thresholds that you might set.
- The *Reporting 8.0 Deployment Guide,* which will help you configure, install, start, stop, and uninstall your Reporting environment. This guide also provides instructions for setting up CCPulse+ administrators and users.
- The *Framework 8.0 Stat Server User's Guide,* which will help you understand statistics configuration.
- The *Reporting 8.0 Reference Manual,* which provides general information about performance measurements, how Reporting behaves during time shifts, and how to set up custom reports for skills-based and partial-period reporting.
- The *Reporting 8.0 Master Index,* which will help you find where other related topics are documented.

Genesys

- *Genesys Technical Publications Glossary,* which ships on the Genesys Documentation Library DVD, 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, provides documented migration strategies for Genesys product releases. Contact Genesys Customer Care for more information.

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

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

Consult these additional resources as necessary:

- *Genesys Interoperability Guide,* which provides information on the compatibility of Genesys products with various Configuration Layer Environments; Interoperability of Reporting Templates and Solutions; and G*plus* Adapters Interoperability.
- *Genesys Database Sizing Estimator Worksheets,* which provides a range of expected database sizes for various Genesys products.

For additional system-wide planning tools and information, see the release-specific listings of System-Level Documents on the Genesys Documentation website (docs.genesys.com).

Genesys product documentation is available on the:

- Genesys Customer Care website at http://genesys.com/customer-care.
- Genesys Documentation site at http://docs.genesys.com/.
- Genesys Documentation Library DVD and/or the Developer Documentation CD, which you can order by e-mail from Genesys Order Management at orderman@genesys.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:

80rt_ad_ccpulse+_12-2011_v8.0.100.00

You will need this number when you are talking with Genesys Customer Care 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 1 describes and illustrates the type conventions that are used in this document.

Table 1: 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 102).	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 . . .

Table 1: Type Styles (Continued)

Type Style	Used For	Examples
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

[] (square brackets) . . . 102
< > (angle brackets) . . . 102

A

Action object properties
  Action . . . 66
  StartTime . . . 66
  type . . . 66
Action property . . . 66
ActionDelay object, in threshold scripts . . . 72
action-related information, obtaining . . . 65
actions
  ActivateWebBrowser . . . 85
  CCPulseNotifier in . . . 74
  CFGDbid object . . . 73
  CFGObjectID object . . . 73
  CFGTenantDbid object . . . 73
  CFGType object . . . 74
  color enumeration objects . . . 75
  configuration objects . . . 73
  example . . . 84
  ExecuteProcess . . . 85
  explanation of predefined actions . . . 85
  Hide . . . 85
  Hide object . . . 74
  PCSpeakerAudioAlert . . . 86
  ResetColor . . . 86
  ResetColor object . . . 74
  SendE-MailWithCDO . . . 86
  SetColor . . . 87
  SetColor object . . . 74
  Show . . . 87
  Show object . . . 74
  using with thresholds . . . 84
  WAVAudioAlert . . . 89
Actions property . . . 65
ActionsStorageFullPath configuration option . . . 30
ActivateWebBrowser action . . . 85
agent group capacity
  calculation . . . 41
  performance . . . 41
agent login
  troubleshooting problems . . . 93
AgentGroupID property . . . 63
AgentID property . . . 62
AgentLoggedIn threshold . . . 79
AgentLoggedOut threshold . . . 80
AgentStats, customizing in Registry . . . 45
angle brackets . . . 102
audience . . . 8

B

basic statistics
  icon . . . 59
  in formulas . . . 60
BinActionsStorageFullPath configuration option . . . 31
BinTemplatesStorage FullPath configuration option . . . 31
BinThresholdsStorageFullPath configuration option . . . 31
BinWorkspacesStorageFullPath configuration option . . . 31
brackets
  angle . . . 102
  square . . . 102

C

CampaignActivated threshold . . . 80
CampaignDeactivated threshold . . . 80
Capacities property
  CurrentInteractions . . . 64
  described . . . 63
  MaxInteractions . . . 64
  MediaType . . . 64
  RoutableInteractions . . . 64

capacity format
display . . . 68
predefined . . . 68
user-defined . . . 68
CCPulse+
overview . . . 13
server connections . . . 53
status bar . . . 54, 56
using composite metrics . . . 50
CCPulseNotifier predefined object . . . 74
CFGDbid object, in action scripts . . . 73
CFGObjectID object, in action scripts . . . 73
CFGTenantDbid object, in action scripts . . . 73
CFGType object, in action scripts . . . 74
color codes, RGB . . . 75
column names
in templates . . . 50
linking objects using . . . 50
Common and Security section . . . 18
composite metrics . . . 50
configuration objects, in action scripts . . . 73
configuration options
ActionsStorageFullPath . . . 30
BinActionsStorageFullPath . . . 31
BinTemplatesStorageFullPath . . . 31
BinThresholdsStorageFullPath . . . 31
BinWorkspacesStorageFullPath . . . 31
CurrentStateEnable . . . 38
CurrentStateRefreshInterval . . . 36
DefaultStorageType . . . 30
DelayBeforeRemoval . . . 36
DisableAgentCurrentState . . . 39
DisableCreateRemoveViews . . . 37
DisableHideShowActionsOnGraph . . . 37
DisableRowColVisibilityChange . . . 37
DisableThresholdsDlg . . . 33, 40
EnableAgentCapacity . . . 40
EnableAgentCapacityStatus . . . 23, 40
EnableAgentGroupCapacity . . . 41
EnableDNAliases . . . 33
enable-ipv6 . . . 18
EnableXMLExport . . . 33
ExtendedCurrentStatus . . . 21
FirstDayOfWeek . . . 22
IgnoreNewObjectNotifications . . . 41
inactivity-timeout . . . 18
local-options . . . 15
MaxNumberOfAgents . . . 38
MediaTypes . . . 21, 68
MinDayNumberInFirstWeek . . . 23
NoAggregationInterval . . . 23
SafeScriptingMode . . . 42
ShowAgentLoginID . . . 34
ShowAgentSkills . . . 34
ShowDurationBeforeStatusName . . . 34
ShowStatusReasons . . . 35
TemplatesStorageFullPath . . . 30
ThresholdsStorageFullPath . . . 30
TimeZone . . . 23
UseStatServerVirtualGroups . . . 42
WholeWeeks . . . 23
WorkspaceAutoLoad . . . 43
WorkspacesStorageFullPath . . . 30
XMLActionsStorageFullPath . . . 32
XMLTemplatesStorageFullPath . . . 32
XMLThresholdsStorageFullPath . . . 32
XMLWorkspacesStorageFullPath . . . 32
connection failures
Configuration Server . . . 54
Stat Server . . . 56
connections, Data Mart . . . 49
conventions
in document . . . 101
type styles . . . 101
current status, determining . . . 62
CurrentAgentTargetState statistic . . . 69
CurrentInteractions capacity item . . . 64
CurrentStateEnable configuration option . . . 38
CurrentStateRefreshInterval configuration option . . . 36
CurrentTargetState statistical category . . . 69
custom statistics . . . 62
determining current status . . . 62
formatting return values . . . 68
icon . . . 59
customizing
AgentStats . . . 45
DeltaAvgTime . . . 46
DeltaCntCalls . . . 46
DeltaTotalTime . . . 46
FormatAgentInfo . . . 47
GridSortingInterval . . . 47
HistoricalUpdateFrequency . . . 47
IntervalForCurrentUpdate . . . 47
NotifyEvery . . . 48
Windows Registry settings . . . 44
CustomStatistic section . . . 21

D

Data Mart
connections . . . 49
using with CCPulse+ . . . 49
DB Server, troubleshooting connection problems with . . . 95
DefaultStorageType configuration option . . . 30
DelayBeforeRemoval configuration option . . . 36
DeltaAvgTime, customizing in Registry . . . 46
DeltaCntCalls, customizing in Registry . . . 46
DeltaTotalTime, customizing in Registry . . . 46
DisableAgentCurrentState configuration option . . . 39
DisableCreateRemoveViews configuration option . . . 37
DisableHideShowActionsOnGraph

configuration option 37
DisableRowColVisibilityChange configuration option 37
DisableThresholdsDlg configuration option 33, 40
displaying agents, troubleshooting 95
DN
aliases in CCPulse+ displays 33
numbers in CCPulse+ displays 33
property 65
DN object properties
Actions 65
DN 65
DNType 65
StartTime 65
Status 65
SwitchID 65
type 65
DNs property 63
DNType property 65
document
conventions 101
version number 101

E

EnableAgentCapacity configuration option . . 40
EnableAgentCapacityStatus configuration option 23, 40
EnableAgentGroupCapacity configuration option 41
EnableDNAliases configuration option 33
enable-ipv6 configuration option 18
EnableXMLExport configuration option . . . 33
error messages
General SQL Server error 96
No connection to specified DB server... . . . 95
No data available for review... 97
No historical associations... 97
ExactValue threshold 80
ExactValueWDuration threshold 80
ExecuteProcess action 85
ExtendedCurrentStatus configuration option . 21

F

FirstDayOfWeek configuration option 22
font styles
italic 101
monospace 102
for referencing basic statistics 60
FormatAgentInfo, customizing in Registry . . 47

G

GreaterThanLimit threshold 81
GreaterThanLimitWDuration threshold 81
GridSortingInterval, customizing in Registry . . 47

H

Hide
action 85
object, in action scripts 74
historical associations
for custom statistics 51
troubleshooting problems with 97
historical reporting considerations
associations for statistics 51
column names 50
composite metrics 50
Data Mart 49
using the last_N_intervals parameter 51
Historical View section 22
HistoricalUpdateFrequency, customizing in Registry 47

I

IgnoreNewObjectNotifications configuration option 41
Import/Export Utility, troubleshooting 94
inactivity-timeout configuration option 18
IntervalForCurrentUpdate, customizing in Registry 47
italics 101

J

JScript language 60

L

last_N_intervals, using this parameter 51
LessThanLimit threshold 82
LessThanLimitWDuration threshold 82
LessThanSpecifiedLimit threshold 81
local-options configuration option 15
Login ID property 62

M

MaxInteractions capacities item 64
MaxNumberOfAgents configuration option . . 38
MediaType capacity item 64
MediaTypes configuration option 21, 68
metrics, using composite 50

MinDayNumberInFirstWeek configuration option . . . 23
monitoring agent activity, troubleshooting problems with . . . 96
monospace font . . . 102

N

NoAggregationInterval configuration option . . . 23
NotEqual threshold . . . 82
NotEqualWDuration threshold . . . 82
NotifyEvery, customizing in Registry . . . 48

O

OutOfRange threshold . . . 83
OutOfRangeWDuration threshold . . . 83

P

PCSpeakerAudioAlert action . . . 86
persistent connection . . . 54
PlaceID property . . . 62
predefined format string . . . 68, 70
predefined objects, in formulas . . . 62
Predefined Statistics window . . . 59

R

ResetColor
  action . . . 86
  object, in action scripts . . . 74
Result object, in threshold scripts . . . 72
return values, formatting . . . 68
RoutableInteractions capacity item . . . 64

S

SafeScriptingMode configuration option . . . 42
sections
  Common and Security . . . 18
  CustomStatistic . . . 21
  Historical View . . . 22
  Storage . . . 29
  UserInterface . . . 33
  View . . . 36
  Workspace . . . 38
SendE-MailWithCDO action . . . 86
servers, disconnecting from CCPulse+ . . . 53
SetColor
  action . . . 87
  object, in action scripts . . . 74
Show
  action . . . 87
  object, in action scripts . . . 74
ShowAgentLoginID configuration option . . . 34
ShowAgentSkills configuration option . . . 34
ShowDurationBeforeStatusName configuration option . . . 34
ShowStatusReasons configuration option . . . 35
SQL Server, troubleshooting problems with . . . 96
square brackets . . . 102
StartTime property . . . 62, 63, 65, 66
Stat Server
  troubleshooting connection problems . . . 97
  troubleshooting sudden disconnect . . . 96
StatAction object . . . 76
StatAlias object, in threshold scripts . . . 72
state object properties
  AgentGroupID . . . 63
  AgentID . . . 62
  Capacities . . . 63
  DNs . . . 63
  LoginID . . . 62
  PlaceID . . . 62
  StartTime . . . 62, 63
  Status . . . 62
  type . . . 62, 63
state predefined object . . . 66
statistics, historical associations for custom . . . 51
StatReasons object, in threshold scripts . . . 73
status bar, CCPulse+ . . . 54, 56
Status property . . . 62, 65
StatValue object, in threshold scripts . . . 72
Storage section . . . 29
storages, customizing location . . . 29
SwitchID property . . . 65
syntax . . . 60

T

templates, column names in . . . 50
TemplatesStorageFullPath configuration option . . . 30
thresholds
  ActionDelay object . . . 72
  AgentLoggedIn . . . 79
  AgentLoggedOut . . . 80
  CampaignActivated . . . 80
  CampaignDeactivated . . . 80
  ExactValue . . . 80
  ExactValueWDuration . . . 80
  example . . . 78
  GreaterThanLimit . . . 81
  GreaterThanLimitWDuration . . . 81
  LessThanLimit . . . 82
  LessThanLimitWDuration . . . 82
  LessThanSpecifiedLimit . . . 81
  NotEqual . . . 82

NotEqualWDuration 82
OutOfRange 83
OutOfRangeWDuration 83
predefined explained 79
Result object 72
StatAction objects 76
StatAlias object 72
StatReasons object 73
StatValue object 72
using with actions 77
WithinRange 83
WithinRangeOfValues 83
WithinRangeWDuration 84
ThresholdsStorageFullPath configuration option 30
TimeZone configuration option 23
troubleshooting
agent login issues 93
agent missing from displays 95
general SQL server error 96
Import/Export Utility disabled 94
missing Stat Server 97
monitoring agent activity 96
no DB Server connection 95
no historical associations 97
Stat Server disconnect 96
wizards disabled 94
type property 62, 63, 65, 66
type styles
conventions 101
italic 101
monospace 102
typographical styles 101

U

user-defined format string 68, 69
UserInterface section 33
UseStatServerVirtualGroups configuration option 42

V

version numbering, document 101
View section 36

W

WAVAudioAlert action 89
WholeWeeks configuration option 23
Windows Registry
customizing settings in 44
root path 45
WithinRange threshold 83
WithinRangeOfValues threshold 83
WithinRangeWDuration threshold 84
wizards, troubleshooting problems with 94
Workspace section 38
WorkspaceAutoLoad configuration option . . . 43
WorkspacesStorageFullPath configuration option 30

X

XMLActionsStorageFullPath configuration option 32
XMLTemplatesStorageFullPath configuration option 32
XMLThresholdsStorageFullPath configuration option 32
XMLWorkspacesStorageFullPath configuration option 32