GENESYS

Gplus **Adapter 8.0**

for Siebel CRM

Deployment Guide

Updated 5/2020:
- On page 32, references to Siebel 8.1.1.09 were changed to 8.1.1.9.
- On pages 104 and 107, the target path was updated in Step 2.

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 Documentation 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 *Genesys Care Support Guide for On-Premises* 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: 80gp_dep_slcrm_05-2020_v8.0.204.00

Table of Contents

Chapter 7

Chapter 8

G*plus* Adapter 8.0

List of Procedures

G*plus* Adapter 8.0

GENESYS

Preface

Welcome to the *G*plus *Adapter 8.0 for Siebel CRM Deployment Guide.* This document describes how to install and configure the components of the G*plus* Adapter.

This document is valid only for the 8.0 release of this product.

Note: For versions of this document that have been created for other releases of this product, 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:

For information about related resources and about the conventions that are used in this document, see the supplementary material starting on page 625.

About Gplus Adapter 8.0 for Siebel CRM

The G*plus* Adapter 8.0 for Siebel CRM is a software solution that provides seamless integration between Siebel CRM and Genesys solutions. This combination brings together Siebel's leading software applications and Genesys' contact center solutions.

The G*plus* Adapter provides a single point of access to contact information. The Adapter brings together multiple media and channels, and provides access to the power of Siebel software, promoting better contact relationships overall.

Intended Audience

This document is primarily intended for system administrators or other individuals who install and configure the G*plus* Adapter. It has been written with the assumption that you have a basic understanding of:

- Computer-telephony integration (CTI) concepts, processes, terminology, and applications.
- TCP/IP Internet-working fundamentals including routing and client /server- application communications via TCP sockets.
- Basic conceptual understanding of database systems, including SQL commands that are necessary to validate availability of your company's database environment. You should involve your company's DBA resources during the implementation of this project.
- The network configurations that are used in your company's computing environment.

You should also be familiar with the following Genesys solutions:

- Framework
- Universal Routing
- Outbound Contact
- eServices (formerly Multimedia)

Note: Refer to the *Genesys Interoperability Guide* for further information about the appropriate Genesys Applications version numbers.

Making Comments on This Document

If you especially like or dislike anything about this document, feel free to e-mail your comments to Techpubs.webadmin@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 scope of this document only and to the way in which the information is presented. Contact your Genesys Account Representative or 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 Support Guide for On-Premises* for complete contact information and procedures.

Document Change History

This section lists content that is new or that has changed significantly since the first release of this document. The most recent changes appear first.

New in Document Version v8.0.202.00

This document has been re-issued to support the G*plus* Adapter for Siebel CRM release 8.0.210. The following topics have been added or changed since the previous release of the document:

- A new chapter, "Deploying Basic HTTP Authentication" on page 557, describes principles and configuration steps to set up Basic HTTP Authentication.
- A new chapter, "Application Monitoring" on page 563, describes principles and configuration steps to enable application monitoring.
- Sections for the following components are updated to describe new features:
  - Communication Server
  - Multimedia
  - Media Routing
  - UCS Gateway
- A new subsection, "New in Document Version v8.0.202.00", summarizes the changes in this document.

New in Document Version v8.0.201.00

This document has been re-issued to support the G*plus* Adapter for Siebel CRM release 8.0.2. The following topics have been added or changed since the previous release of the document:

- Many sections are updated to describe support for Siebel 8.1.1.14 and 8.2.2.14 (IP2014).
- The chapter "Deploying the Multimedia Component" on page 391 is updated to describe Siebel Chat UI support.

- A new subsection, "New in Document Version v8.0.201.00", summarizes the changes in this document.

New in Document Version v8.0.102.00

This document has been re-issued to support the G*plus* Adapter for Siebel CRM release 8.0.110. The following topics have been added or changed since the previous release of the document:

- A new subsection, "Supported Siebel Versions" on page 32, describes supported Siebel versions.
- A new chapter, "Patching and Configuring Siebel CRM" on page 41, describes patching and configuring steps for specific Siebel versions.
- Many sections are updated to describe support for Siebel 8.1.1.11 and 8.2.2.4.
- A new subsection, "New in Document Version v8.0.102.00", summarizes the changes in this document.

New in Document Version v8.0.101.00

This document has been re-issued to support the G*plus* Adapter for Siebel CRM release 8.0.1. The following topics have been added or changed since the previous release of the document:

- A new subsection, "New in This Release" on page 120, summarizes the feature enhancements that the 8.0.1 version of the G*plus* Adapter for Siebel CRM Campaign Synchronization Component.
- Support for the new 8.0.1 version of the Campaign Synchronization Component that is comprised of the Siebel waves campaign management feature. See, Chapter 5, "Deploying the Campaign Synchronization Component," on page 119 for more information.
- A new subsection, "New in Document Version v8.0.101.00", summarizes the changes in this document.

New in Document Version v8.0.002.00

- Supported software versions have been updated throughout the document. Otherwise, there have been no significant changes to this document since the last release.

1

System Requirements

This chapter outlines the minimum system and software requirements for the G*plus* Adapter 8.0 for Siebel CRM (G*plus* Adapter).

Please review the system requirements before you install the G*plus* Adapter.

This chapter provides information on the following topics:

- Deployment Options, page 27
- Installation Options, page 28
- Universal Definition File, page 31
- Supported Siebel Versions, page 32
- System Requirements for Gplus Adapter Components, page 33
- Date and Time Synchronization, page 39

Deployment Options

If you are deploying the G*plus* Adapter for the first time, G*plus* Adapter migration issues are not your concern. If your organization has already deployed an earlier version of the G*plus* Adapter, your deployment options will vary depending on the version of the G*plus* Adapter that you use. Table 1 on page 28 describes, at a very high level, where to find the appropriate instructions for installing the G*plus* Adapter 8.0 for Siebel CRM, depending on whether your organization has a previous version of the Adapter.

Note: In the following table and throughout this document, the Voice, Multimedia, and Media Routing components of the G*plus* Adapter are identified as "driver-based components," and the Configuration Synchronization, Campaign Synchronization, Communication Server, and UCS Gateway components of the G*plus* Adapter are called "server-based components."

Table 1: Deployment Options

Customers	Details	Comments
New users of the G*plus* Adapter for Siebel CRM.	Follow the installation instructions that are detailed in the G*plus* *Adapter 8.0 for Siebel CRM Deployment Guide*.	New users are those who are deploying the G*plus* Adapter for Siebel CRM for the first time.
Existing users of G*plus* Adapter 6.5.2, 7.0, 7.1, 7.2, or 7.5 for Siebel CRM who want new (8.0) features and maintenance.	Follow the procedures for migrating from the G*plus* Adapter 6.5.2, 7.0, 7.1, 7.2, or 7.5 for Siebel to the G*plus* Adapter 8.0 for Siebel CRM in the appropriate section of the *Genesys Migration Guide*.	**Warning:** For information regarding the uninstall process, refer to the G*plus* Adapter for Siebel CRM documentation. Then if necessary, contact Genesys Customer Care.
Existing users of *pre*-6.5.2 G*plus* Adapter for Siebel who are seeking new (8.0) features and maintenance.	Contact Genesys Technical Support to discuss options that are specific to your current version and configuration.	

Note: Customers are responsible for migration of *all* customizations.

Installation Options

Some of the features of this release of the G*plus* Adapter 8.0 for Siebel CRM have a dependency upon other Siebel or Genesys products, and some also depend on other G*plus* Adapter components. To use these G*plus* Adapter features, you must implement the requirements.

The standard procedure, which this document assumes you are following, is to install all of the required external product requirements first, before you start the installation of the G*plus* Adapter.

G*plus* Adapter External Requirements

Table 2 on page 29 lists the Genesys and Siebel application software required for various types of G*plus* Adapter deployments. Depending on the G*plus* Adapter components and features installed at your site, certain pieces of Siebel and Genesys software must also be present to support your deployment, in addition to the fundamental requirements.

The underlying Genesys framework of the integrated Siebel and G*plus* Adapter solution is powered by Genesys Framework and T-Servers.

Table 2:   G*plus* Adapter Options and Related System Requirements

Fundamental requirements (required for any deployment):	
• Siebel Call Center Application • Genesys Framework	
Basic Voice • Fundamental requirements, as listed previously • Genesys T-Server	**Genesys E-mail** • G*plus* Adapter Multimedia Component • Genesys eServices
Expert Contact • Genesys CTI-Less T-Server	**Universal Callback** • Genesys Universal Callback server
Media Routing • G*plus* Adapter Multimedia Component • Interaction Server (part of Genesys eServices)	**Outbound Campaign** • Siebel Marketing • Genesys Outbound Contact Server
Genesys Chat • G*plus* Adapter Multimedia Component • Genesys eServices	**Communication Server** • Fundamental requirements only, as listed previously
UCS Gateway • Fundamental requirements, as listed previously • Sun Java Runtime Environment (JRE) 1.5.x or later	**Configuration Synchronization** • Fundamental requirements, as listed previously • Genesys T-Server
Campaign Synchronization • Fundamental requirements, as listed previously • Siebel Marketing • Genesys Outbound Contact Server	**iWD Routing** • G*plus* Adapter Multimedia Component • Genesys intelligent Workload Distribution (iWD)

The previously listed requirements are particularly significant if you want additional information about the software that supports a G*plus* Adapter feature. The G*plus* Adapter integrates many software applications. Detailed

information about some G*plus* Adapter features is only available in the proper documentation for the integrated software application that powers the feature.

G*plus* Adapter Internal Requirements

Some of the components of this 8.0 release of the G*plus* Adapter for Siebel CRM have a dependency upon another G*plus* Adapter component. This means that you must install some components before others. Table 3 lists each major G*plus* Adapter component and provides notes on its dependencies on other G*plus* Adapter components.

To see the external software requirements for each component, refer to the listing for that component under "System Requirements for Gplus Adapter Components" on .

Table 3: Internal Dependencies Among G*plus* Adapter Components

G*plus* Adapter Components/Features /Functionality	G*plus* Adapter (Internal) Requirements
Communication Server	No prerequisite G*plus* Adapter components. The Communication Server is a prerequisite for G*plus* Adapter driver-based components.
Configuration Synchronization Component	No prerequisite G*plus* Adapter components. No G*plus* Adapter components have dependencies on this component. However, Configuration Synchronization functionality is intended for use with the Voice and Multimedia components.
Campaign Synchronization Component	No prerequisite G*plus* Adapter components. No G*plus* Adapter components have technical dependencies on this component. However, Campaign Synchronization functionality is intended for use with the Outbound Campaign feature of the Voice Component.
Voice Component including: • Basic Voice • Expert Contact • Outbound Campaign • Universal Callback	G*plus* Communication Server for Siebel CRM should be installed first. You must first import the Basic Voice feature before importing the other G*plus* Adapter Voice Components. You must configure the Voice Component, including all of the features you intend to use, before adding any other driver-based Adapter Component.

Table 3: Internal Dependencies Among G*plus* Adapter Components (Continued)

G*plus* Adapter Components/Features /Functionality	G*plus* Adapter (Internal) Requirements
Multimedia Component including: • Genesys E-mail • Genesys Chat	The following components should be deployed: • G*plus* UCS Gateway for Siebel CRM • G*plus* Communication Server for Siebel CRM
Media Routing Feature	The following component should be deployed: • G*plus* Adapter for Siebel CRM Multimedia Component
iWD Routing Feature	The following component should be deployed: • G*plus* Adapter for Siebel CRM Multimedia Component

Universal Definition File

This release includes a new definition file, the GenComm_universal.def file. This GenComm_universal.def file is used by the G*plus* Communication Server, Voice Component, Multimedia Component, Media Routing Component, and iWD Routing Component. This text file provides sample Siebel configuration data for these components. The file contains configuration parameters, driver descriptions, profile descriptions, and event and command descriptions. The configurations created with the data from this *.def file incorporate the following drivers:

• Gplus_Universal

And the following profiles:

• G*plus* Universal Profile

For more information, see Chapter 3, "Deploying the Communication Server for Siebel CRM", Chapter 6, "Deploying the Voice Component", and Chapter 8, "Deploying the Multimedia Component".

Before Importing the Definition File

The GenComm_universal.def file includes commands and events to support the Voice, Multimedia, Media Routing, and iWD Routing Component components. Use this file to create the drivers and the profiles for these components.

To implement some configurations, you must modify the GenComm _universal.def file before importing it.

Migrations and Upgrades

If you are migrating or upgrading to the G*plus* Adapter 8.0 for Siebel CRM, refer to the *Genesys Migration Guide* for crucial information about migrating your customizations to the newer version of the Adapter. The *Genesys Migration Guide* includes information and examples that may simplify the migration of customizations coded in your old.def file.

G*plus* Adapter implementation libraries are common to all driver-based components (Voice, Multimedia, Media Routing) of the G*plu*s Adapter 8.0 for Siebel CRM.

This means that if any of the driver-based components is upgraded to the newer version, then all G*plus* Adapter for Siebel CRM components running in this Siebel Server will share and use the new upgraded Adapter implementation libraries.

Supported Siebel Versions

The G*plus* Adapter 8.0 for Siebel CRM supports the following versions of the Siebel CRM:

- 7.7
- 7.8
- 8.0
- 8.1

The 8.0.110 version of the G*plus* Adapter also supports Siebel versions 8.1.1.11 and 8.2.2.4 (IP2013) in both HI and Open UI modes.

Note: It is mandatory to deploy Patchset #5 or later for these Siebel versions. Therefore, full versions must be at least 8.1.1.11.5/8.2.2.4.5.

Note: Siebel versions from 8.1.1.9 to 8.1.1.11.4 inclusively, and from 8.2 to 8.2.2.4.4 inclusively, are not supported.

The 8.0.200 version of the Gplus Adapter also supports Siebel versions 8.1.1.14 and 8.2.2.14 (IP2014) in both HI and Open UI modes.

All G*plus* Adapter components use the following directory name to Siebel version mapping:

- 7.7 for Siebel version 7.7 and 7.8
- 8.0 for Siebel version 8.0
- 8.1 for Siebel version 8.1, but prior to 8.1.1.9
- 8.1_8.2_OUI for Siebel version 8.1.1.11.5/8.2.2.4.5 (IP2013) in both HI and Open UI modes

- IP2014 for Siebel version 8.1.1.14/8.2.2.14 (IP2014) in both HI and Open UI modes

There are significant changes in the deployment procedures for G*plus* Adapter for Siebel 8.1.1.11/8.2.2.4 and 8.1.1.14/8.2.2.14 as compared to earlier versions.

System Requirements for Gplus Adapter Components

The G*plus* Adapter is integrated with the Siebel Server, so all Siebel Server hardware and software requirements are applicable to the G*plus* Adapter and its components. The Siebel Server must have the Communications Session Manager component enabled in order to use the G*plus* Adapter. Refer to the Siebel documentation for current system requirements.

Note: Information about supported operating systems and platforms is available in the *Genesys Supported Operating Environment Reference Manual*.

Note: For the most up-to-date information about supported Genesys applications, consult the G*plus* Adapter for Siebel CRM chapter in the latest *Genesys Interoperability Guide*.

The following is a list of the G*plus* Adapter components:

- Communication Server Requirements, page 33
- Configuration Synchronization Component Requirements, page 34
- Voice Component Requirements, page 35
- UCS Gateway Requirements, page 37
- Multimedia Component Requirements, page 38
- Media Routing Component Requirements, page 39

Communication Server Requirements

Hardware Requirements

You need the following hardware to deploy the G*plus* Adapter for Siebel CRM Communication Server:

- Pentium III 700 Mhz CPU or faster
- 256 MB of RAM
- 64 MB of disk space

- 800 × 600 256-color monitor or higher
- Network adapter and network connection

Note: These are minimum hardware requirements. For large call centers and/or large call volumes, more hardware resources (especially RAM and CPU) will be required.

Software Requirements

You need the following software to deploy the G*plus* Communication Server.

Genesys Applications

- Consult the G*plus* Adapter for Siebel CRM chapter in the *Genesys Interoperability Guide*.

Configuration Synchronization Component Requirements

Hardware Requirements

You need the following hardware to deploy the G*plus* Adapter for Siebel CRM Configuration Synchronization Component:

- Pentium III 700 Mhz CPU or faster
- 256 MB of RAM
- 200 MB of disk space
- 800 × 600 256-color monitor or higher
- Network adapter and network connection

Note: These are the minimum hardware requirements. For large call centers and/or large call volumes, more hardware resources (especially RAM and CPU) will be required.

Software Requirements

You need the following software to deploy the G*plus* Adapter for Siebel CRM Configuration Synchronization Component:

Siebel 7.7/7.8/8.0/8.1/8.1.1.11/8.2.2.4 Applications

- Siebel Server
- Siebel Tools
- Siebel Web Engine

- Siebel Web Client

Genesys Applications

- Framework

Note: Please refer to the *Genesys Interoperability Guide* for further information about the appropriate Genesys Applications version numbers.

Voice Component Requirements

The Voice Component includes several features:

- Basic Voice
- Expert Contact
- Outbound Campaign
- Universal Callback

Most of these Voice features can be regarded as options, but to deploy any feature of the Siebel CRM Voice Component, you must first deploy the fundamental Basic Voice feature. The system requirements for this feature appear in the following lists.

Basic Voice Feature (Fundamental Voice Component) Requirements

The following minimum requirements apply to all Voice Component features.

Siebel 7.7/7.8/8.0/8.1/8.1.1.11/8.2.2.4 Applications

- Siebel Server
- Siebel Call Center
- Siebel Tools
- Siebel Web Client

Genesys Applications

- Framework
- *Gplus* Communication Server
- Configuration Server
- Configuration Manager

- T-Server (7.2 or higher is required, if you are using the Two-Step Network Transfer functionality)

Note: Please refer to the *Genesys Interoperability Guide* for further information about the appropriate Genesys Applications version numbers.

Expert Contact Feature Requirements

In addition to the previously listed requirements for the Basic Voice feature, you need the following software to deploy the Expert Contact feature of the Voice Component.

Genesys G*plus* Adapter Components

- Basic Voice

Genesys Applications

- Genesys Expert Contact

Note: Please refer to the *Genesys Interoperability Guide* for further information about the appropriate Genesys Applications version numbers.

Outbound Campaign Feature Requirements

In addition to the previously listed requirements for the Basic Voice feature, you need the following software to deploy and use the Outbound Campaign feature.

Siebel 7.7/7.8/8.0/8.1/8.1.1.11/8.2.2.4 Applications

- Siebel Marketing

Note: In addition to the fundamental Siebel application requirements (the requirements that are imposed by the Basic Voice feature), the Outbound Campaign feature also requires Siebel Marketing. Siebel Marketing is required to manage outbound campaign interactions.

Genesys Applications

- Outbound Contact

> **Note:** Please refer to the *Genesys Interoperability Guide* for further information about the appropriate Genesys Applications version numbers.

Genesys G*plus* Adapter Components

- G*plus* Configuration Synchronization Component
- G*plus* Campaign Synchronization Component

Universal Callback Feature Requirements

*I*n addition to the previously listed requirements for the Basic Voice feature, you need the following software to deploy the Universal Callback feature of the Voice Component:

Genesys Applications

- Universal Callback Server 7.1

UCS Gateway Requirements

Hardware Requirements

You need the following hardware to deploy the G*plus* UCS Gateway:

- Pentium III 700 Mhz CPU or faster
- 256 MB of RAM
- 64 MB of disk space
- 800 × 600 256-color monitor or higher
- Network adapter and network connection

> **Note:** These are minimum hardware requirements. For large call centers and/or large call volumes, more hardware resources (especially RAM and CPU) will be required.

Software Requirements

You need the following software to deploy the G*plus* UCS Gateway.

Genesys Applications

• Framework

> **Note:** Please refer to the *Genesys Interoperability Guide* for further information about the appropriate Genesys Applications version numbers.

Other Applications

• Sun Java Runtime Environment (JRE) 1.5.x

Multimedia Component Requirements

You need the following software to deploy the G*plus* Adapter 7.5 for Siebel CRM Multimedia Component.

Siebel 7.7/7.8/8.0/8.1/8.1.1.11/8.2.2.4 Applications

• Siebel Server
• Siebel Tools
• Siebel Call Center
• Siebel Web Client

Genesys Applications

• Framework
• Multimedia
• Universal Routing

> **Note:** For information about service packs and patches that are required by Siebel, see Siebel documentation and the Siebel Technical Support website.

> **Note:** Please refer to the *Genesys Interoperability Guide* for further information about the appropriate Genesys Applications version numbers.

Media Routing Component Requirements

You need the following software to deploy the G*plus* Adapter 8.0 for Siebel CRM Media Routing Component.

Siebel 7.7/7.8/8.0/8.1/8.1.1.11/8.2.2.4 Applications

*   Siebel Server
*   Siebel Tools
*   Siebel Call Center
*   Siebel Web Client

Genesys Applications

*   Multimedia
*   Framework
*   Universal Routing

Note: Please refer to the *Genesys Interoperability Guide* for further information about the appropriate Genesys Applications version numbers.

Genesys G*plus* Adapter Components

*   G*plus* Adapter 8.0 for Siebel CRM Multimedia Component
*   G*plus* Communication Server 8.0

Date and Time Synchronization

Interactions between client/server applications in a multi-tiered computing environments rely heavily on the proper configuration of the client and host system. An important consideration is the proper management of the Date/Time and Time Zone configuration.

The records exchanged among Genesys, Siebel, and database systems carry timestamps. Computing systems designed to work together can produce unpredictable results if the system dates and time are not synchronized. Therefore, please make every effort to adhere to the following time synchronization guidelines:

*   Enterprise servers that host Genesys, Siebel or database applications should have their Time Zone configured properly for their particular geographic region. To configure the timezone, please see the instructions from the hardware or software manufacturer for your particular operating system(s).

- As a best practice, all servers that process Genesys and Siebel data should be synchronized to the second. This synchronization should be maintained continuously and validated on a regular basis. For security reasons, Genesys strongly recommends the use and deployment of Network Time Protocol (NTP), version 4.1 or greater. The full protocol suite should be deployed, and derivative protocols like DAYTIME, SNTP, or RDATE should be avoided.

- For a large environment with multiple call centers or multiple contact points, the deployment of a network time appliance with stratum 1 access should be considered. For those deployments in WAN environments where packet latency may affect NTP operation, multiple NTP appliances should be deployed to individual LAN segments. Genesys does not recommend particular equipment at this time, however, we do want you to be aware of the following manufacturers of dedicated timeserver hardware:

Table 4: Dedicated Timeserver Hardware

Manufacturer	Hardware
EndRun Technologies http://www.endruntechnologies.com	Praecis Cntp
Datum Corporation http://www.datum.com	Syncserver S100
True Time http://www.truetime.com	TimeVault
TrueTime http://www.truetime.com	NTS-200
Lantronix http://www.lantronix.com	CoBox NTP

- For enterprise deployments in regulated environments (financial services, insurance, government, brokerage, and so on), check with appropriate legal resources. Certain jurisdictions have specific requirements that may supersede these suggestions.

- Deployment of timeserver resources can impact the operations of certain authentication technologies. You should coordinate the deployment of NTP with members of your systems security and database technologies department(s).

- For more information about NTP, see http://www.ntp.org.

Chapter

2

Patching and Configuring Siebel CRM

This chapter describes the procedures for patching and configuring the G*plus* Adapter 8.0 for Siebel CRM (G*plus* Adapter).

Overview

If you are using Siebel CRM versions 8.1.1.11/8.2.2.4 (IP2013) or 8.1.1.14/8.2.2.14 (IP2014), then you must complete the following procedures before deploying the G*plus* Adapter:

- Allowing file system access
- Patching the applicationcontext.js file

These procedures are described in the following sections.

Allowing file system access

Due to a Siebel security policy, file system access by the Siebel server script is disabled by default. For more details about this security policy, refer to Oracle Document ID 1594198.1.

Procedure:
Allowing file system access for the Siebel script

Purpose: To allow file system access for the Siebel server script.

Start of procedure

1. In the Siebel Web Client, log in as a Siebel administrator.

2. Select Site Map > Administration - Application > System Preferences.

3. Create a new record, specifying Allow File System Access as the System
    Preference Name and TRUE as the System Preference Value.

End of procedure

Next Steps

*   If applicable, complete the procedure for "Patching the
    applicationcontext.js file" on .

Patching the applicationcontext.js file

Due to a current defect in Siebel version 8.1.1.11.5, you must add a fix to the
applicationcontext.js file. Refer to Oracle Service Request 3-8755729071 for a
list of defects and their current status.

If you are using a Siebel version that includes a fix for this defect, you can skip
this procedure.

Procedure:
Patching the applicationcontext.js file

Purpose: To patch the applicationcontext.js file.

Start of procedure

1. On your system, locate the applicationcontext.js file. It is typically in the
    following directory: siebsrvr/WEBMASTER/siebel_build/scripts/Siebel

2. Make a backup of the applicationcontext.js file.

3. At the top of the RemoveApplet function, insert the following code:
    if (this.GetExternalActiveApplet() === ax) { this.SetExternalActiveApplet(null); }

> **Note:** The ax variable in the condition must correspond to the first input
> argument of the function. If the first input argument is not ax, change it
> to match. The code should look like the following example:
> ...aq.prototype.RemoveApplet=function(ax){**if(this.GetExternalActiveApp**
> **let()===ax){this.SetExternalActiveApplet(null);**}delete
> i[ax.GetName()];...

End of procedure

Next Steps

* You can now proceed with deployment of the G*plus* Adapter.

3 Deploying the Communication Server for Siebel CRM

This chapter describes how to configure and install the G*plus* Communication Server for Siebel CRM. It includes the following sections:

Overview

G*plus* Communication Server for Siebel CRM (the Adapter) reduces in-process coupling between Genesys and Siebel code and increases the isolation level among communication components. The result is:

- decreased risk of failure
- more flexible deployment possibilities for a heterogeneous environment (platforms, operating system).

The Adapter's communication component consists of two major parts:

1. **G*plus* Communication Driver:** The G*plus* Communication Driver is a thin layer that delivers messages between Siebel Communication Server and the G*plus* Communication Server. The G*plus* Communication Driver must be deployed on a Siebel Server host.

2. **G*plus* Communication Server:** The G*plus* Communication Server acts as a host process for pluggable remote drivers for underlying Genesys communication middleware (voice T-Servers and Open Media Interaction Server). The Communication Server may run on different hosts with different platforms and operating systems.

Notes: • The G*plus* Communication Server is a prerequisite for all driver-based components.

• The Adapter is not compatible with the Siebel CTI parameter, MaxCommToolbar, if the value of that parameter is more than 1. For more details about this parameter, refer to the Siebel documentation.

The process of configuring and installing the Adapter consists of:

1. Planning the configuration of the Adapter. See, Planning, page 47.

2. Configuring the Genesys section of the Adapter. See, Configuring Genesys, page 49.

3. Installing Communication Server. See, Installation, page 62.

4. Deployment of the Communication Driver on a Siebel host. See, Configuring Siebel, page 72.

5. Configuring Siebel. See, Configuring Siebel, page 72.

New in This Release

This section provides information about new features or functionality in the G*plus* Communication Server.

Release 8.0.210   The following new features or functionality are included in this release:

• Protection of sensitive data in logs in Genesys Communication Driver.

• HTTP Authentication. See "Deploying Basic HTTP Authentication" on page 557 for details.

• One way SSL between the Communication Driver and Communication Server.

• Application monitoring. See "Application Monitoring" on page 563 for details.

Release 8.0.2   The following new features or functionality are included in this release:

- The communication protocol between the server and driver is improved (now uses Bidirectional-stream Over Synchronous HTTP, or BOSH. See http://xmpp.org/extensions/xep-0124.html).
- Support for SIP Business Continuity.
- Support for Genesys Framework 8.5.
- Support for Red Hat Enterprise Linux 6.
- Support for Red Hat Enterprise Linux 7.
- Support for AIX 7.1.
- Support for Solaris 11.

Release 8.0.1   No new features were introduced in this release.

Release 8.0.0   The following new features or functionality are included in this release:

- Support for the Red Hat Enterprise Linux (RHEL) operating system.
- Support for the secured connections to Interaction Server, T-Server, Configuration Server, and Message Server.
- Support for improved connection management.

Planning

When planning, you must consider the following:

Will the Adapter be deployed on the *same* host as the Siebel Server, or will it be deployed on a *different* host?

Will the Adapter support the SIP Business Continuity feature, provided that Genesys switch and server components are already mirrored at two separate sites (a two-site deployment; see *SIP Server 8.1 High-Availability Deployment Guide*) in an active-active configuration?

The following are also important considerations:

- The average CPU load on the Siebel server host.
- The network throughput, latency, and reliability.

The Adapter is implemented as a Genesys server, and is represented by an Application object in Genesys Configuration Manager.

The installation of the component delivers both the G*plus* Communication Server and the G*plus* Communication Driver, which may be considered as an installation of a Genesys server that delivers the added ability of a G*plus* Communication Driver for a given platform (operating system).

Possible Deployment Scenarios

The following three Adapter deployment scenarios may be considered:

1. Both the G*plus* Communication Server and G*plus* Communication Driver parts of the component may be deployed on the same host as the Siebel Server. In this case, one installation delivers both parts to the required locations.

2. The G*plus* Communication Server part of the component may be deployed on a dedicated host, but on the same platform (operating system) as the Siebel host. The installation is performed on the dedicated host, and the G*plus* Communication Driver part of the component should be moved manually to the Siebel Server host after that.

   a. This deployment scenario is the same as the one described above, but includes support for SIP Business Continuity. The installation for the G*plus* Communication Server part of the component is performed on two dedicated Genesys hosts that belong to the sites (that is, two instances of the G*plus* Communication Server part of the component are deployed on two dedicated hosts that represent DR Peers in SIP Business Continuity architecture. See *SIP Server 8.1 High-Availability Deployment Guide*). The G*plus* Communication Driver part of the component is identical on both hosts, so one of them should be manually moved to the Siebel Server host after deployment. A final deployment could look like "Gplus Communication Server with SIP Business Continuity":

Figure 1: G*plus* Communication Server with SIP Business Continuity
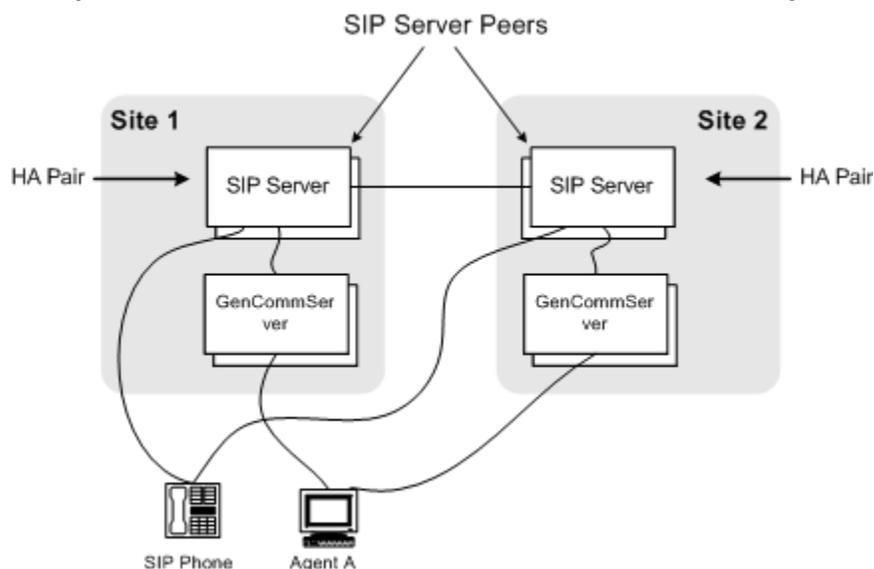
3. The G*plus* Communication Server part of the component may be deployed on a dedicated host, which uses an operating system that is different than that of the Siebel host. Technically, this method of deployment can be

performed, but it requires the following installations to deliver the G*plus* Communication Server and Driver parts of the component to the different platforms:

- The first installation is performed on the dedicated host to install the G*plus* Communication Server part.
- The second installation is performed on the Siebel host to install the G*plus* Communication Driver part. This installation requires an additional Application object in Configuration Manager that is used only during installation. The installation also installs the G*plus* Communication Server part of the component, along with the required G*plus* Communication Driver part. It can be removed manually.
  a. This deployment scenario is the same as the one described above, but includes support for the Business Continuity feature. The first installation is performed twice on both dedicated hosts. (See the similar deployment described above in 2a.)

Configuring Genesys

This section describes how to configure the Genesys section of the G*plus* Communication Server.

Configuring the Genesys section of the G*plus* Communication Server consists of the following sections:

Importing the Communication Server Application Template

This section describes how to import the Communication Server Application Template.

Recommendations

Genesys recommends using an Application Template when you are configuring your Adapter. The Application Template for your Adapter contains the most important configuration options set to the values recommended for the majority of environments. When modifying configuration options for your Adapter later in the process, you can change the values inherited from the template rather than create all the options by yourself.

Procedure:
Communication Server: Importing the Communication Server Application Template

Purpose: To import the Communication Server Application Template.

Start of procedure

1. In Configuration Manager, under Environment, right-click the Application Templates folder.

2. Select Import Application Template.

3. Browse to and select the Application Template for the Communication Server. Depending on the version that you are using, the name of this template is:
   - Release 8.0.0: Gplus_Comm_Server_for_SiebelCRM_800.apd
   - Release 8.0.1: Gplus_Comm_Server_for_SiebelCRM_801.apd
   - Release 8.0.2: Gplus_Comm_Server_for_SiebelCRM_802.apd

4. Click Open.
   The Properties dialog box for the Application Template displays.

5. Optional: Edit the Application Template name.

6. Click OK to accept the default values.
   The Application Template has been imported to the Genesys Configuration Layer.

End of procedure

Next Steps

- Create the Configuration Layer Application object for the Communication Server. See, Procedure: Communication Server: Creating the Communication Server Application object, on page 51.

Creating the Communication Server Application Object

This section describes how to create the Communication Server Application object.

Procedure:
Communication Server: Creating the Communication Server Application object

Purpose: To create the Application object for the Communication Server.

Start of procedure

1. In Configuration Manager, under Environment, right-click the Application Templates folder.

2. Select New > Application.

3. Select the Application Template that you just created, depending on the version that you are using:
   - Release 8.0.0: Gplus_Comm_Server_for_SiebelCRM_800.apd
   - Release 8.0.1: Gplus_Comm_Server_for_SiebelCRM_801.apd
   - Release 8.0.2: Gplus_Comm_Server_for_SiebelCRM_802.apd

4. Click OK. The Properties dialog box for the Application object appears.

End of procedure

Next Steps

- Configure the tabs in the Properties dialog box for the Communication Server. See, Procedure: Communication Server: Configuring the tabs in the Properties dialog box, on page 51.

Configuring the Tabs in the Properties Dialog Box

This section describes how to configure the tabs in the Properties dialog box, arranged in the order in which they display.

Procedure:
Communication Server: Configuring the tabs in the Properties dialog box

Purpose: To configure the tabs in the Properties dialog box for the Communication Server.

Start of procedure

General Tab   **1.** Click the General tab in the Properties dialog box. See Figure 2.
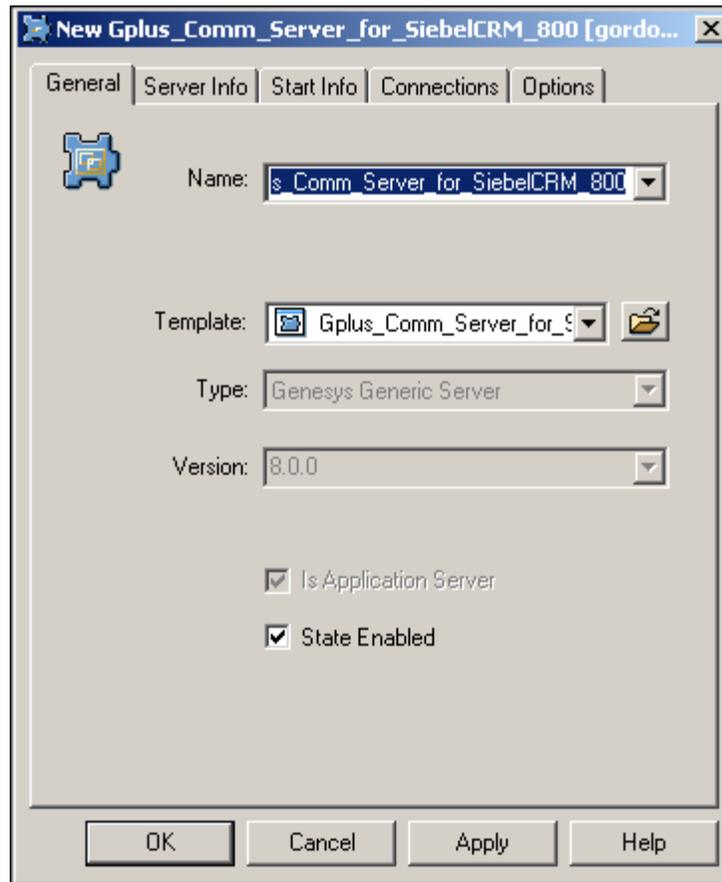
Figure 2: View of General Tab in a Single-Tenant Environment

2. In the Name list, click the name of the Application object you are configuring.

3. In the Template list, click the name of the template you are configuring or use the Browse button to select the template you are configuring.

4. Click Apply.

5. Next, if you are working in a multi-tenant environment, go to the Tenants Tab; otherwise, go to the Server Info Tab.

Note: The Tenants tab only displays, if you are working in multi-tenant environment.

Tenants Tab   **6.** Select the Tenants tab in the Properties dialog box. See Figure 3 on page 53.
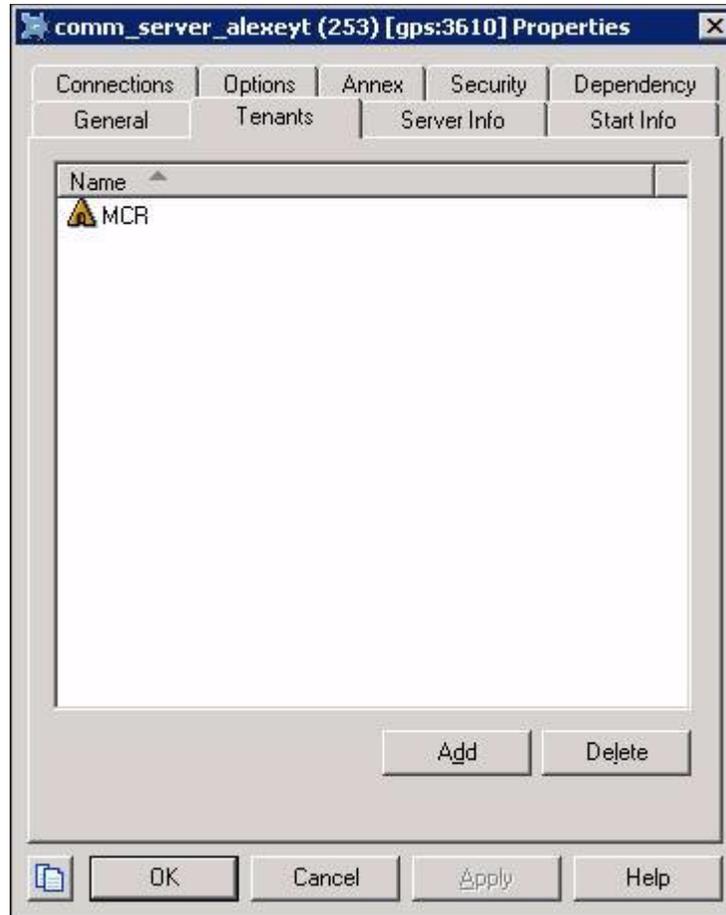
Figure 3: View of Tenants Tab

7. Click Add.

8. Select the Genesys Tenants under which the objects that are exported from Siebel are created. For example, in Figure 3 above, the MCR tenant has been added.

Server Info Tab **9.** Select the Server Info tab in the Properties dialog box.

10. In the Host field, use the Browse button to select the host where you are installing the G*plus* Communication Server and click OK.

11. In the Communications Port field, enter any valid port number for the port with the ID of default. Genesys recommends setting this to 18001, since this corresponds to the value provided by the installation sample configuration file for Siebel (the *.def file).

Note: This value determines the HTTP port number used by the G*plus* Communication Server to receive HTTP packets from the G*plus* Communication Driver. Use the value you enter here later when you configure the Siebel portion of the G*plus* Communication Server.

Start Info Tab   **12.** Select the Start Info tab in the Properties dialog box.

13. In the Working Directory field, enter the full name of the Component installation directory on the host you specified on the Server Info tab. The value you enter in this field is used as the default destination folder during installation.

14. Enter any valid value into each of the following fields:
- Command Line
- Command Line Arguments

The values you enter in these fields are overwritten during installation; however, the data must be present in these fields during the configuration process.

15. Leave the default values for the remaining fields.

Connections Tab   **16.** Select the Connections tab in the Properties dialog box.

Note: This step is *mandatory* for configuring the server to work with the Voice and Multimedia components. You may also configure a connection to the Genesys Message Server.

The content of the Connections tab depends on the Adapter components being used.
- For the Voice component: It is necessary to add a connection to the corresponding T-Server. For backward compatibility, the Voice component also supports a way to define connection to the T-Server without adding the T-Server to the Connections list, as in Release 7.2.
- For the Multimedia and Media Routing components: It is necessary to add a connection to the Interaction Server.

17. Click Add to attach a connection to the Interaction Server and/or T-Server. See Figure 4 on page 55.
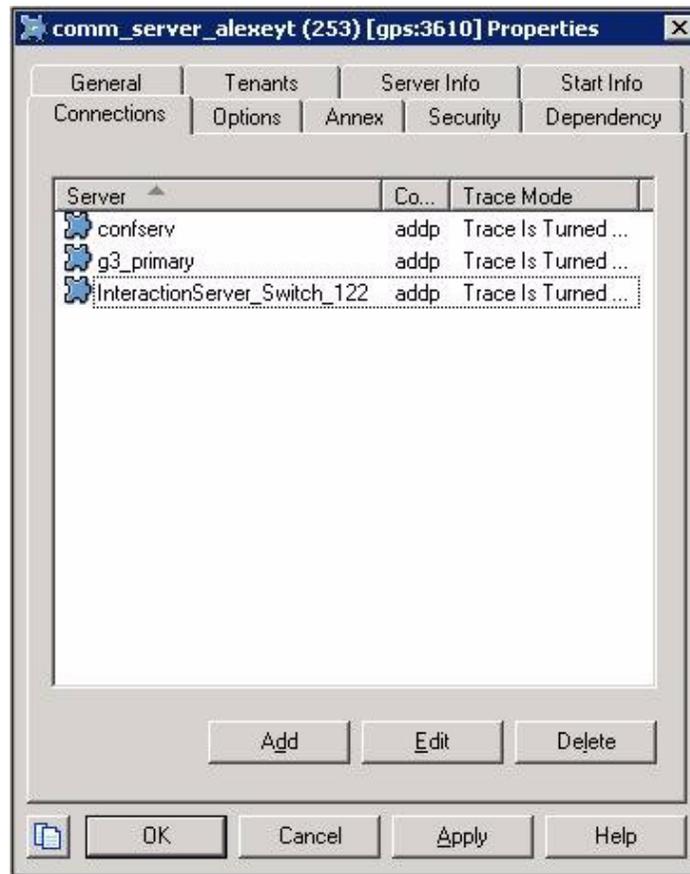
Figure 4: View of Connections Tab in Multi-Tenant Environment

- Optionally, you can add a connection to the Genesys Message Server.
- Optionally, you can add a connection to the Genesys Configuration Server.

Options Tab **18.** Select the Options tab. See Figure 5 on page 56.
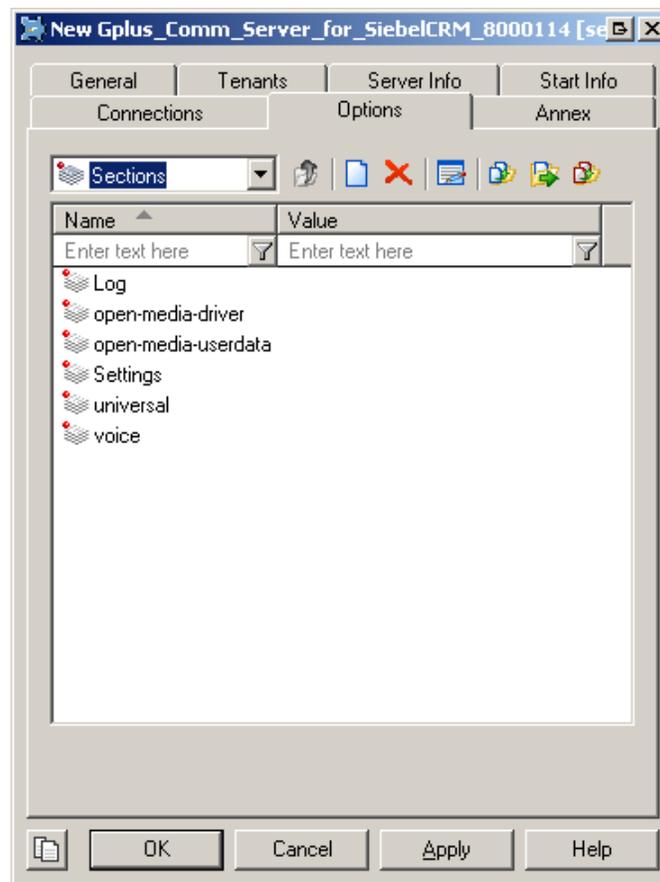
Figure 5: View of Options Tab

19. In the Sections pane, the following sections are listed:

- Log
- open-media-driver
- open-media-userdata
- Settings
- universal
- voice

20. Double-click the Settings section to start.

21. Set the configuration options that are described in the following section, "Setting the Genesys Configuration Options for the Communication Server" on page 57.

End of procedure

Next Steps

- Set the configuration options for the Communication Server. See the section, "Setting the Genesys Configuration Options for the Communication Server" on page 57.

Setting the Genesys Configuration Options for the Communication Server

Unless specified otherwise, set the Communication Server configuration options in the Options tab of the Application object using the following navigation path:

- In Genesys Administrator—Application object > Options tab > Advanced View (Options)

- In Configuration Manager—Application object > Properties dialog box > Options tab.

For ease of reference, the options have been arranged in alphabetical order within their corresponding sections:

Changes to configuration options take effect after the Communication Server restarts.

Log Section

The G*plus* Communication Server supports the unified set of log options (common log options) to allow precise configuration of the log file output. For a complete list of unified log options and their descriptions, see the "Common Log Options" chapter of the *Framework Configuration Options Reference Manual*.

Selective Protection of Sensitive Data in Logs

To prevent displaying the *private* parameters included in UserData, Extensions, and Reasons in the Adapter's log, a number of options can be configured in the Options section of the G*plus* Communication Server Application object in Configuration Manager.

These options are described in the Log-Filter and Log-Filter-Data sections of the *Framework Configuration Options Reference Manual*.

The Log-Filter section defines the parameter default-filter-type with valid values copy, hide, or skip. This parameter specifies the default method of presenting KVList information (including UserData, Extensions, and Reasons) in the log. For example—see Figure 6.
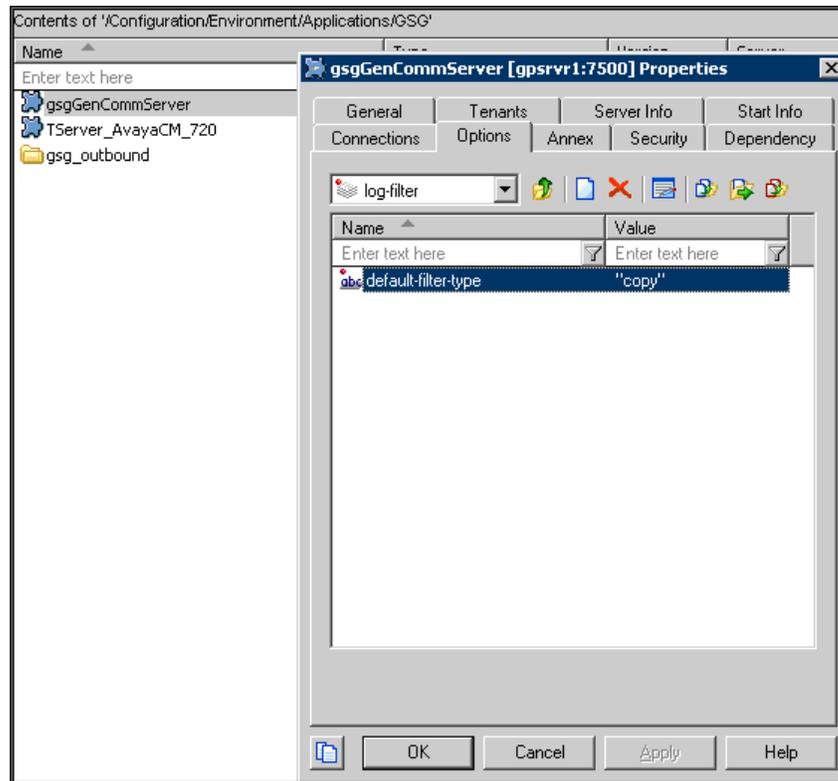
Figure 6: Log-Filter Parameters in Configuration Manager

The Log-Filter-Data section defines a list of key-value pairs and specifies the way of presenting each individual KVList pair defined by the key name in the log. The key defines the parameter name, while the value (which can be defined as copy, hide, or skip) specifies how this parameter is presented in the log—for example, see Figure 7 on page 59.
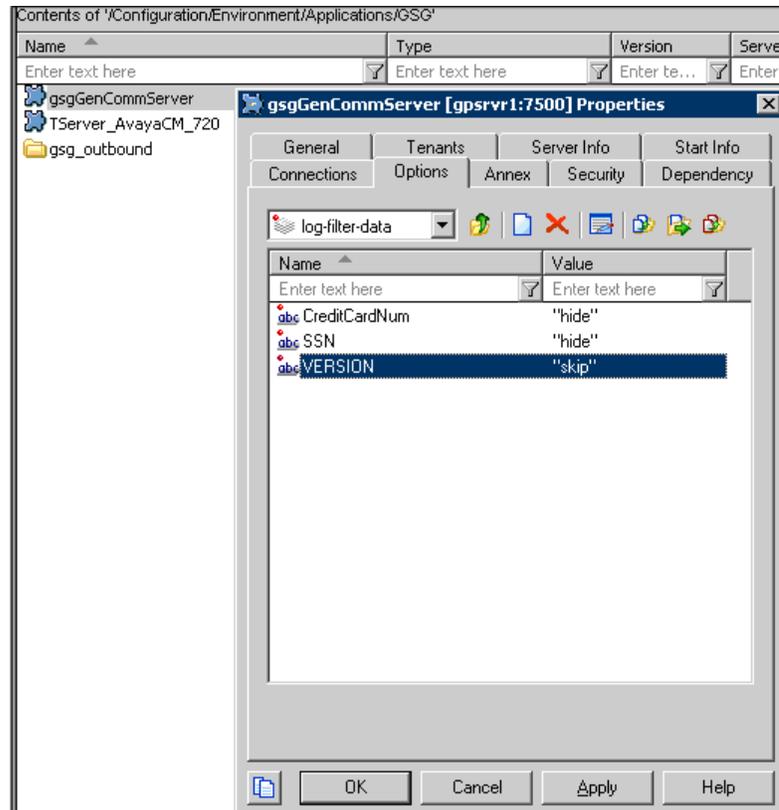
Figure 7: Log-Filter-Data Parameters in Configuration Manager

open-media-driver Section

The open-media-driver section has various options that can be configured. The
following section describes the possible options:

internal-port

Default Value: -1
Valid Value: Any positive integer that defines the exact port value

Defines the port number that is used for interthread communications. If the
value of this configuration option is set to -1, this option is disabled. If the
value of this configuration option is set to 0, the port is selected automatically.

on-connect-delay

Default Value: 0
Valid Value: 10000 (10 seconds)

Determines the delay, measured in milliseconds (ms), between connecting to
Interaction Server and registering the first client. This delay is used to allow

Interaction Server to restore the interactions state after an abnormal termination.

Note: This parameter value must be less than the registration-timeout Interaction Server parameter value.

service-duplicate-overwrite

Default Value: false
Valid Values: true, false

If the value of this option is set to true, subsequent agent sessions with the same user name are allowed, and the previous agent session are disabled. If the value of this option is set to false, only one session per agent is allowed. This option is needed as a workaround when a session is not closed by Siebel, but when the browser is forcibly closed.

open-media-userdata Section

The open-media-userdata section has the following option that can be configured:

ThirdPartyId

Default Value: top

Defines how particular user data is sent to Siebel. All data is normally sent encoded as a single UserData field. It may be convenient to have part of the user data delivered to Siebel as a top-level key. The user-data key should be added to the section with the value of top, indicating that the key will be sent as a top-level key in the data set.

Settings Section

The Settings section has various options that can be configured. The following section describes the possible options:

ProcessingThreadPoolSize

Default Value: 30
Valid Value: Any positive integers that are greater than or equal to (>=) 4

Provides the value for the maximum number of threads that process tasks per pool. The following two thread pools are present:

1. The first thread is used for processing incoming requests.

2. The second thread is used for processing responses on outgoing requests.

RequestTimeout

Default Value: 30000

Valid Value: Any positive integer that is greater than (>) or equal to (=) 3000.

Provides the timeout value in milliseconds (ms) for handling a request that is submitted by the Communication Server to the Communication Driver.

Universal Section

The universal section has various options that can be configured. The following section describes the possible options:

preview-media-type

Default value: outboundpreview
Valid Values: Any non-empty string containing media type for push preview media.

Defines the media type for push preview media. If the value of the option is different from the default value (outboundpreview), you must change it before compiling the *.srf file. In the MCR Session Manager Business Service, change the following Business Service User Properties value:
OpenMediaReadyCommand_outboundpreview to the following new value:
OpenMediaReadyCommand_<new value>.

preview-park-view

Default value: Empty string
Valid Values: Empty string or name of the view (For more information about the view name, see the Universal Routing Server documentation.)

Defines the view that is used to transfer push preview interactions. If the value of this option is an empty string, the transfer of push preview interactions is performed without the use of the special intermediate queue that goes directly to the agent.

Note: For the sample strategy (Park Queue Restore) of the park view/queue, see the SampleBPforMMrecovery.wie file in the installation directory.

Voice Section

The Voice section has the following option that can be configured:

CheckDNStateOnLogin

Default value: false
Valid Values: TRUE, true, True, FALSE, false, False

Determines whether or not the checking of the DN state is enabled upon agent login. If the checking of the DN state is enabled, the agent login is rejected, if that DN is already registered.

Metrics Section

The metrics section has the following option that can be configured:

snmp-trap-receiver

Default value: Empty string
Valid Values: A trap receiver address in the format of host:port.

Determines the address of the SNMP trap receiver. If a value is specified, the adapter enables monitoring functionality. See "Application Monitoring" on page 563 for more details.

Creating and configuring the SIP Business Continuity DR Peer Communication Server Application Object

This section is applicable if the G*plus* Adapter is intended to support the SIP Business Continuity (BC) feature.

The second Communication Server Application Object must be created and configured on the SIP BC alternative site. This can be done similar to the procedure for installing the adapter for a singular site, as described in "Configuring Genesys" on page 49.

The options of both Communication Server Application Objects must be fully synchronized, with one exception.

Note: This step is mandatory for configuring the server to work with the Voice component. You must add a connection to the SIP Server that is the second peer in the SIP Business Continuity deployment. This can be performed similar to how the first peer of the SIP BC DR pair (another SIP Server) is configured in "Configuring the Tabs in the Properties Dialog Box" on page 51. For more details, refer to the *SIP Server 8.1 High-Availability Deployment Guide*.

Installation

This section describes the installation process for the G*plus* Adapter 8.0 Communication Server for Siebel CRM and consists of the following steps:

• Installing the Communication Server, page 63
• Configuring the GenCommDrv.ini File (Optional), page 64

For further information about your deployment options, see the section "Planning" on page 47.

Installing the Communication Server

To run an installation that delivers the G*plus* Communication Server for Siebel CRM driver part of the component directly under the Siebel Server (see Deployment Scenario1and the second installation section of Deployment Scenarios 3 and 3a on page 48), the system administrator should have appropriate privileges and stop the Siebel Server before installation.

Note: Do not use special symbols in any destination directory name when installing in a UNIX environment.

Procedure:
Communication Server: Installing the Communication Server

Purpose: To install the G*plus* Communication Server.

Start of procedure

Starting the Communication Server Installation Process

1. To start the installation process, run the setup.exe file (for Windows), or install.sh file (for UNIX) from the G*plus* Communication Server installation package.

2. Answer the installation questions according to your selected deployment scenario.

Deploying the G*plus* Communication Driver on the Siebel Host

3. Stop the Siebel Server.

4. Move the files from the G*plus* Communication Driver subdirectory of the target installation directory on the dedicated host to the bin folder (for Windows), or to the lib folder (for UNIX) in the Siebel directory hierarchy of the Siebel host.

End of procedure

Next Steps

- Configure the GenCommDrv.ini file. This step is optional. See the section, "Configuring the GenCommDrv.ini File (Optional)".

Configuring the GenCommDrv.ini File (Optional)

The GenCommDrv.ini file is delivered to the bin folder (for Windows), or the lib folder (for UNIX) in the Siebel directory hierarchy of the Siebel host in two ways:

1. The file is delivered to the bin folder directly through the installation.

2. The files is manually delivered to the bin folder, as described in Deployment Scenario 2 on .

The GenCommDrv.ini file describes the default configuration of the G*plus* Communication Driver part of the component.

Note: If you are an advanced system administrator, you may change the predefined configuration of the G*plus* Communication Driver, but you should take responsibility for coordinating the configuration options for both the Driver and Server parts of the component.

Table 5 describes the properties associated with the GenCommDrv.ini file.

Table 5: GenCommDrv.ini Parameters

Property name	Values	Description
General section		
CheckServiceTimeout	Default Value: 60000 Valid Values: A positive integer that is greater than or equal to (>=)1000 **Note:** If an invalid value is applied, the default value (60000) is used	(Optional) This property provides the timeout value in milliseconds (ms) for CheckService activities between G*plus* Communication Driver and G*plus* Communication Server.
ClientConnectionsPer Server	Default Value: 1 Valid Value: An integer that is greater than or equal to (>=) 1. **Note:** If an invalid value is applied, the default value (1) is used.	(Optional) This property provides the value for the maximum number of allowed client connections to a particular endpoint (listening port on the server side).

Table 5: GenCommDrv.ini Parameters (Continued)

Property name	Values	Description
ProcessingThreadPool Size	Default Value: 30 Valid Value: An integer that is greater than or equal to (>=) 4 **Note:** If an invalid value is applied, the default value (30) is used.	(Optional) This property provides the value for the maximum number of threads that process tasks per pool. There are two thread pools present: one for processing incoming requests, and one for processing responses on outgoing requests.
RequestTimeout	Default Value: 30000 Valid Value: A positive integer that is greater than (>) the defined minimum value, where the minimum value is 3000. **Note:** If an invalid value is applied, the default value (30000) is used.	(Optional) This property provides the timeout value in milliseconds (ms) for handling an HTTP request submitted by the G*plus* Communication Driver to the G*plus* Communication Server.
SiebelAPI	Default Value: 2 Valid Values: 1, 2 **Note:** If an invalid value is applied, the default value (2) is used.	(Optional) This property defines the Siebel API version to use.
OnReconnectDelay	Default Value: 4000 Valid Values: A positive integer that is greater than or equal to (> =) 1000. **Note:** If an invalid value is applied, the default value (4000) is used.	(Optional) This property defines the timeout value in milliseconds from the moment when the connection is established and the first request to the server is sent. It allows the server to clean up an agent session before its re-creation.
LogFiltering	Default Value: false Valid Values: false, true	(Optional) Enables or disables log filtering.

Table 5: GenCommDrv.ini Parameters (Continued)

Property name	Values	Description
LogFilterType	Default Value: copy Valid Values: copy|hide|skip| **Note:** Refer to the *Genesys Framework Configuration Options Reference Manual* for more details (the default-filter-type option).	(Optional) Specifies the default way of presenting KV pairs information in the log. Refer to the "Hide Selected Data in Logs" chapter in the *Genesys Security Deployment Guide* for information about how to use this option. The option applies to command and event parameters.
LogFilterData	Default Value: Empty String Valid Values: "<item_1>,<item_2>,...<item_N>" where <item> = <key>:<value>, and where <key> and <value> are any string, except special symbols, such as : (a colon),\ (a backward slash), and , (a comma). **Note:** Values must be in double quotes, as shown above.	(Optional) The treatment for specific keys. This value overrides the default treatment specified by LogFilterType.
Log Subsystem section		
LogDir	Default Value: ./ Valid Values: A valid and accessible directory path. **Note:** If an empty string is applied, then the default (./) is used. If an invalid directory path is provided or is not accessible, then the file is not created.	(Optional) This property provides the directory name where the logs will be stored.

Table 5: GenCommDrv.ini Parameters (Continued)

Property name	Values	Description
LogExt	Default Value: log Valid Value: A non-empty string. **Note:** If an invalid value is applied, the default value (log) is used.	(Optional) This property provides the GenCommDrv log file name extension.
LogFile	Default Value: GenCommDrv Valid Value: Any non-empty string. **Note:** If an invalid value is applied, the default value (GenCommDrv) is used.	(Optional) This property provides the GenCommDrv log file name prefix.
LogFilesCount	Default Value: 10 Valid Value: A positive integer that is greater than or equal to (>=) 2 **Note:** If an invalid value is applied, the default value (10) is used.	(Optional) This property provides the number of log files.
LogLevel	Default Value: 2 Valid Values: 0, 1, 2, 3, 4, 5, 6 **Note:** If an invalid value is applied, the default value (2) is used.	(Optional) This property defines the trace level. Valid values for this parameter are: • 0—All messages (log events of Standard, Trace, Interaction, and Debug levels) are logged. • 1—The same as 0 (zero). • 2—Messages of low priority are logged. • 3—Messages of normal priority are logged. • 4—Messages of high priority are logged. • 5—Log alarm messages only are logged. • 6—No messages are logged.

Table 5: GenCommDrv.ini Parameters (Continued)

Property name	Values	Description
MaxLogKB	Default Value: 10240 Valid Value: A positive integer that is greater than or equal to (>=) 100. **Note:** If an invalid value is applied, the default value (10240) is used.	(Optional) This property provides the maximum GenCommDrv log file size in kilobytes.
HTTP Client Connection section		
ClientPortRangeStart	Default Value: 0 Valid Values: A positive integer greater than or equal to (>=) 0 (zero). **Note:** If an invalid value is applied, the default value (0) is used.	(Optional) This property provides the value for the start of the port range used to open a client connection to the Gplus Communication Driver. A value of 0 (zero) means that the server relies on the operating system to assign a unique port number to the application.
ServerReconnectDelay	Default Value: 10000 Valid Value: A positive integer that is greater than or equal to (>=) 1000 **Note:** If an invalid value is applied, the default value (10000) is used.	(Optional) This property provides the timeout value in milliseconds (ms) between attempts to reconnect to the *Gplus* Communication Server.

Table 5: GenCommDrv.ini Parameters (Continued)

Property name	Values	Description
ServerConnectTimeout	Default Value: 0 Valid Value: A positive integer that is greater than or equal to (>=) 0 **Note:** If an invalid value is applied, the default value (0) is used.	(Optional) This property defines the timeout value in milliseconds for establishing a connection. A default value of 0 (zero) indicates that the predefined operating system value will be used.
Security properties of HTTP Connection section		
Certificate	Default Value: "" (an empty string) Valid Value: For Windows this should be a base 64 or hex encoded string. For Unix, this should be the absolute path to the certificate file (see also "T-Server Connections" on page 223 in Chapter 6). **Note:** If the value is an empty string (""), an insecure connection is used.	(Optional) This property identifies the certificate to be used.

Table 5: GenCommDrv.ini Parameters (Continued)

Property name	Values	Description
CertificateKey	Default Value: "" (an empty string) Valid Value: For Windows, this property is not used. Therefore, no valid values are defined. For Unix, this is the full path to the private key file: <serial_#>_<host_name>_priv_key.pem (see "T-Server Connections" on page 223 in Chapter 6).	(Optional) This property identifies the Certificate Key to be used.
TrustedCA	Default Value: "" (an empty string) Valid Values For Windows, this property is not used. Therefore, no valid values are defined. For Unix, this is the full path to the certificate authority file: ca_cert.pem (see "T-Server Connections" on page 223 in Chapter 6). Default value is "" (empty string).	(Optional) This property identifies the full path to the certificate authority file.

By default, if the Adapter is configured to support SIP Business Continuity, the instructions described above are sufficient. One disadvantage to this approach is that all options defined in the GenCommDrv.ini file will uniformly affect the G*plus* Communication Driver in both directions, towards the first and the second instances of G*plus* Communication Server that are installed on DR Peer Sites in SIP BC topology.

If you are an advanced user and want to change or distinguish certain options that are relevant to the preferred SIP BC Site or the alternative SIP BC Site, then you can specify three separate .ini files:

• GenCommDrv.ini - default file, as described above. Communication Driver accepts them and applies them to work with the both instances of Communication Server.

• GenCommDrv2.ini - this file provides options for controlling connections and logging for the Communication Server located on the alternative SIP BC Site. Options in this file override the ones defined in GenCommDrv.ini. In this configuration, the options defined in GenCommDrv2.ini affect the Communication Driver and the second Communication Server (the alternative SIP BC Site) whereas the options defined in GenCommDrv.ini are only applicable to the Communication Driver and the first Communication Server pair (the preferred SIP BC Site).

• GenCommBC.ini - this file provides options for controlling logging messages during changeover between SIP BC Sites. Only six of them are applicable: LogFile, LogDir, LogExt, MaxLogKB, LogFilesCount, LogLevel.

The Contents of the GenCommDrv.ini File

The following code snippet shows the contents of the GenCommDrv.ini file:

```
// ============================
// General
// ============================
SiebelAPI              = 2
ProcessingThreadPoolSize   = 30
ClientConnectionsPerServer = 1
RequestTimeout = 30000
OnReconnectDelay = 4000


// ============================
// Log subsystem properties
// ============================
LogFile          = GenCommDrv
LogDir         = ./
LogExt         = log
MaxLogKB          = 10240
LogFilesCount      = 10
LogLevel         = 2


// ============================
// Log filtering properties
// ============================

//Enables or disables log filtering
//Values: true|false, default value - false
LogFiltering = false
```

//Defines the treatment for all KV pairs by default
//Values: copy|hide|skip|..., default value - copy, refer to Genesys documentation for more details
LogFilterType = copy
//The treatment for specific keys, overrides the default treatment specified by LogFilterType
//Values: <item_1>,<item_2>,...<item_N>, default value - empty string
// where <item> = <key>:<value>
// where <key>,<value> are any string excepting special symbols like ':','\',','
LogFilterData = ""

// =============================
// HTTP Client connection properties
// =============================

ServerReconnectDelay = 10000
ClientPortRangeStart = 0
ServerConnectTimeout = 0
// =============================
// Security properties of HTTP connection
// (used when https is configured in
PrimaryGenCommServerURL/BackupGenCommServerURL URLs)
// =============================
Certificate   = ""
CertificateKey   = ""
TrustedCA   = ""

Uninstalling the Communication Server

Complete information about uninstalling the G*plus* Adapter and its components is provided in Chapter 14, "Uninstallation Instructions," on page 565.

Configuring Siebel

This section describes how to configure the Siebel part of the G*plus* Communication Server for Siebel CRM.

Configuring the Siebel CTI for the Communication Server

To start using the Communication Server, it is necessary to create a corresponding communication configuration and communication profile for the GenCommDrv library file. Genesys provides the GenComm_universal.def configuration file as an example of the configuration.

Communication Driver Mandatory Parameter

Setting the following parameter is mandatory for the Communication Driver:

Note: Additional parameters may be mandatory to use with the voice or multimedia component. Refer to Chapter 6, "Deploying the Voice Component," on page 175 and Chapter 8, "Deploying the Multimedia Component," on page 391 for more information on these parameters.

PrimaryGenCommServerURL

This parameter should specify the URL of the Communication Server. For example:

PrimaryGenCommServerURL=http://cti.mycompany.com:9121

where cti.mycompany.com is the host name where the G*plus* Communication Server is installed (must be resolvable from the Siebel host), and 9121 is an example of a port as configured in Genesys Configuration Manager.

Note: In Configuration Manager the port must be empty (the default) or have the setup value for the Connection Protocol option.

BackupGenCommServerURL

If you have a backup G*plus* Communication Server, enter the information for the BackupGenCommServerURL parameter. Otherwise, leave it empty.

BCPrimaryGenCommServerURL

This parameter should specify the URL of the G*plus* Communication Server residing on the alternative DR site in SIP BC architecture. If this feature is not supported, leave this parameter empty.

BCBackupGenCommServerURL

This parameter should specify the URL of a backup G*plus* Communication Server residing on the alternative DR site in SIP BC architecture. If this feature is not supported, leave this parameter empty.

BCChangeoverTimeout

This parameter should be set if you intend to use the SIP Business Continuity feature. Otherwise, leave it empty. It indicates the time duration (in seconds) the G*plus* Communication Driver should wait after the current G*plus* Communication Server becomes unavailable before trying to connect to the second instance of the G*plus* Communication Server located on the alternative SIP BC site.

BCPreferredPingTimeout

This parameter should be set if you intend to use SIP Business Continuity feature, otherwise leave it empty. It indicates the time duration (in seconds) between two contiguous attempts to check availability of the instance of a working G*plus* Communication Server located on the preferred site in SIP BC topology.

> **Note:** If support for SIP Business Continuity is enabled, the G*plus* Voice Component is a mandatory part of deployment. The following voice relevant options are also mandatory in this section: TServerAppName and BCTServerAppName (see "Configuring Siebel" on page 196 in "Deploying the Voice Component").

Communication Server Device Events

This section provides the detailed descriptions of the following Communication Server component device events:

Table 6: G*plus* Communication Server Device Events

Communication Server is in service	
Device Event	GCSInService
Description	The G*plus* Communication Server is back in service-running and connected to the G*plus* Communication Driver.
TrackingID	Not applicable
Attributes	Not applicable
Communication Server is out of service	
Device Event	GCSOutOfService
Description	The G*plus* Communication Server appears to be out of service. The Genesys Communication Server is not running or there is no connection to it from the Gplus Communication Driver.
TrackingID	Not applicable
Attributes	Not applicable
Siebel Session is required to make "Changeover SIP BC DR peer"	
Device Event	BCChangeoverRequiredNotify

Table 6: G*plus* Communication Server Device Events (Continued)

Description	This event is sent to Siebel if the current SIP BC peer becomes unavailable. The Siebel agent session is asked to make a changeover to another SIP BC Peer i.e. connect to the second Communication Server.
TrackingID	Not applicable.
Attributes	Not applicable.
The Preferred SIP BC Peer is in operation	
Device Event	BCPreferredSiteInOperationNotify
Description	The preferred SIP BC side (SIP Server and Communication Server) is back in operation.
TrackingID	Not applicable.
Attributes	Not applicable.
Siebel Session is required to make "AgentLogin"	
Device Event	BCAgentLoginRequiredNotify
Description	After switchover between SIP DR peers on Genesys side, the existing Siebel session is notified to repeat AgentLogin action.
TrackingID	Not applicable.
Attributes	Not applicable.
Siebel Session is required to make "AgentReady"	
Device Event	BCAgentReadyRequiredNotify
Description	After switchover between SIP DR peers on Genesys side, the existing Siebel session is notified to repeat "AgentReady" action.
TrackingID	Not applicable.
Attributes	Not applicable.

Communication Server Device Commands

This section provides the detailed descriptions of the following Communication Server component device commands:

Table 7: G*plus* Communication Server Device Commands

Siebel Session requires a changeover to another SIP BC Peer	
Device Command	BCDoChangeoverRequest
Description	The current agent session initiates to establish connection to the second Communication Server which belongs to another SIP BC peer.
Fields	Not applicable.
Siebel Session checks the preferred SIP BC Peer operability	
Device Command	BCPingPreferredSideRequest
Description	The current agent session initiates a ping of the preferred SIP BC peer site and its operability (SIP Server and Communication Server).
Fields	Not applicable.
Siebel Session requests "AgentLogin" action	
Device Command	BCAgentLogin
Description	The current agent session initiates a voice device command "AgentLogin".
Fields	Not applicable.
Siebel Session requests "AgentReady" action	
Device Command	BCAgentReady
Description	The current agent session initiates a voice device command "AgentReady".
Fields	Not applicable.

In the GenComm.def file, Business Continuity Commands and Events are shown in the example below:

```
[EventHandler:BCChangeoverRequiredNotify]
  Comments           = ""
  DeviceEvent        = "BCChangeoverRequired"
  Profile            = "Gplus Universal Profile"
```

```
            Order             = "0"
            Response          = "BCChangeoverRequiredResponse"

         [EventResponse:BCChangeoverRequiredResponse]
            Comments = ""
            Command  = "BCDoChangeoverRequest"

      [Command:BCDoChangeoverRequest]
         DeviceCommand = "BCDoChangeover"
         Description   = "trying to changeover middleware CTI environment ..."
         CmdData       = "BCDoChangeoverRequestData"
         Hidden        = "FALSE"
         MenuPosition  = "100"
         Profile       = "Gplus Universal Profile"

      [EventHandler:BCPingPreferredSiteRequiredNotify]
         Comments             = ""
         DeviceEvent          = "BCPreferredSitePingRequired"
         Profile              = "Gplus Universal Profile"
         Order                = "0"
         Response             = "BCPingPreferredSiteRequiredResponse"

         [EventResponse:BCPingPreferredSiteRequiredResponse]
            Comments = ""
            Command  = "BCPingPreferredSiteRequest"

      [Command:BCPingPreferredSiteRequest]
         DeviceCommand = "BCPingPreferredSite"
         Description   = "ping the preferred site availability ..."
         CmdData       = "BCPingPreferredSiteRequestData"
         Hidden        = "FALSE"
         MenuPosition  = "100"
         Profile       = "Gplus Universal Profile"

      [EventHandler:BCPreferredSiteInOperationNotify]
         ServiceMethod        = "Communications Client.ShowStatusText"
         ServiceParam.Text     = "Genesys framework on the preferred site is In Operation"
         Comments             = ""
         DeviceEvent          = "BCPreferredSiteInOperation"
         Profile              = "Gplus Universal Profile"
         Order                = "0"

      [EventHandler:BCAgentLoginRequiredNotify]
         Comments             = ""
         DeviceEvent          = "BCAgentLoginRequired"
         Profile              = "Gplus Universal Profile"
         Order                = "0"
         Response             = "BCAgentLoginRequiredResponse"

         [EventResponse:BCAgentLoginRequiredResponse]
            Comments = ""
```

```
                          Command  = "BCAgentLoginRequest"

                  [Command:BCAgentLoginRequest]
                    DeviceCommand = "BCAgentLogin"
                    Description   = "auto Agent Login ..."
                    Hidden        = "FALSE"
                    MenuPosition  = "100"
                    Profile       = "Gplus Universal Profile"

                  [EventHandler:BCAgentReadyRequiredNotify]
                    Comments               = ""
                    DeviceEvent            = "BCAgentReadyRequired"
                    Profile                = "Gplus Universal Profile"
                    Order                  = "0"
                    Response               = "BCAgentReadyRequiredResponse"

                    [EventResponse:BCAgentReadyRequiredResponse]
                      Comments = ""
                      Command  = "BCAgentReadyRequest"

                  [Command:BCAgentReadyRequest]
                    DeviceCommand = "BCAgentReady"
                    Description   = "auto Agent Ready ..."
                    Hidden        = "FALSE"
                    MenuPosition  = "100"
                    Profile       = "Gplus Universal Profile"
```

List of Requirements to Support SIP Business Continuity

The following is a list of requirements to support SIP Business Continuity:

- Confirm that SIP Business Continuity is already deployed in the environment to which you want to deploy the G*plus* Adapter and is working properly (see "Planning" on page 47).

- Create two Communication Server Application Objects: one for the Preferred Site (see "Configuring Genesys" on page 49), and one for the Alternate Site that represents a mirrored deployment in SIP BC architecture (see page "Creating and configuring the SIP Business Continuity DR Peer Communication Server Application Object" on page 62).

- Install the G*plus* Communication Server parts of the Adapter to each of the dedicated hosts that represent a Disaster Recovery (DR) pair in SIP BC architecture (the Preferred and Alternative Sites). Install the G*plus* Communication Driver part of the Adapter to the Siebel host. (See 2a and 3a in "Possible Deployment Scenarios" on page 48).

- Confirm that both G*plus* Communication Servers in the DR pair are configured identically, so that their options are synchronized (see "Creating and configuring the SIP Business Continuity DR Peer Communication Server Application Object" on page 62). The Voice component is also a mandatory part of deployment to support SIP Business Continuity. Confirm that the Voice components on both DR peers are configured identically.

- (Optional) You can create extra GenCommDrv2.ini and files GenCommBC.ini on the Siebel host (see "Configuring the GenCommDrv.ini File (Optional)" on page 64) to customize the connection options and logging subsystem for each direction to the SIP BC peer.

- Confirm that the SIP Business Continuity related events and commands are configured on the Siebel host (see "Communication Server Device Events" on page 74 and "Communication Server Device Commands" on page 76).

- Confirm that the SIP Business Continuity related Driver options are configured on the Siebel host. Ensure that the names of the first and second SIP-Server instances are properly reflected in the options for Driver: TServerAppName and Driver:BCTServerAppName (see "Configuring Siebel" on page 72).

- Confirm that Library Name = GenCommBC in the CTI driver configuration.

GENESYS

Chapter

4

Deploying the Configuration Synchronization Component

This chapter describes how to configure and install the G*plus* Adapter for Siebel CRM Configuration Synchronization Component (the Adapter), which is a server-based component.

This chapter includes the following sections:

Overview

The G*plus* Adapter for Siebel CRM Configuration Synchronization Component (the Adapter) is a server-based component. The process for configuring and installing this component is significantly different than the process for driver-based components.

The process of configuring and installing the Adapter includes three general procedures:

- • Configuring Genesys

- Installation of the Configuration Synchronization Component
- Configuring Siebel

Each of these procedures is explained in detail under a separate heading as follows:

Before importing agent configuration data from Siebel to Genesys, see the *Gplus Adapter 8.0 for Siebel CRM User's Guide* and read the section "Synchronizing Siebel Extensions and ACD Queues with Genesys Configuration."

New in This Release

This section provides information about new features or functionality in the Configuration Synchronization Component:

Release 8.0.210 No new features were introduced in this release.

Release 8.0.2 The following new features or functionality are included in this release:

- Support for Siebel versions 8.1.1.14/8.2.2.14 (IP2014) for both HI and Open UI modes.
- Support for Genesys Framework 8.5.
- Support for Red Hat Enterprise Linux 6.
- Support for Red Hat Enterprise Linux 7.
- Support for AIX 7.1.
- Support for Solaris 11.

Release 8.0.110 The following new features or functionality are included in this release:

- Support for Siebel versions 8.1.1.11/8.2.2.4 (IP2013) and later for both HI and Open UI modes.

Release 8.0.1 No new features were introduced in this release.

Release 8.0.0 The following new features or functionality are included in this release:

- Support for the Red Hat Enterprise Linux (RHEL) operating system.

Configuring Genesys

This section describes how to configure the Genesys section of the Configuration Synchronization Component.

Configuring the Genesys section of the Configuration Synchronization Component consists of the following sections:

Prestart Information

Before starting the configuration process you should have the following Genesys Framework applications running:

- Configuration Database
- Configuration Server
- Configuration Manager

As a Configuration Manager user, you should have sufficient privileges to make changes to Configuration Layer objects.

Importing the Configuration Synchronization Component Application Template

This section describes how to import the Configuration Synchronization Component Application Template.

Recommendations

Genesys recommends using an Application Template when you are configuring your Adapter. The Application Template for your Adapter contains the most important configuration options set to the values recommended for the majority of environments. When modifying configuration options for your Adapter later in the process, you can change the values inherited from the template rather than create all the options by yourself.

Procedure:
Configuration Synchronization Component: Importing the Configuration Synchronization Component Application Template

Purpose: To import the Configuration Synchronization Component Application Template.

Start of procedure

1. In Configuration Manager, under Environment, right-click the Application Templates folder.

2. Select Import Application Template.

3. Browse to and select the Application Template for the Configuration Synchronization Component. Depending on the version of Configuration Manager that you use, this name varies slightly, as follows:
   - For versions 7.0 and earlier, use the following:
     - Release 8.0.0: Gplus_SiebelCRM_Config_Synch_800_for_CL_70.apd
     - Release 8.0.1: Gplus_SiebelCRM_Config_Synch_801_for_CL_70.apd
     - Release 8.0.2: Gplus_SiebelCRM_Config_Synch_802_for_CL_70.apd
   - For versions 7.1 and later, use the following:
     - Release 8.0.0: Gplus_SiebelCRM_Config_Synch_800_for_CL_71_and_higher.apd
     - Release 8.0.1: Gplus_SiebelCRM_Config_Synch_801_for_CL_71_and_higher.apd
     - Release 8.0.2: Gplus_SiebelCRM_Config_Synch_802_for_CL_71_and_higher.apd

4. Click Open.
   The Properties dialog box for the Application Template object displays.

5. Click OK to accept the default values.

   The Application Template object has been imported to the Genesys Configuration Layer.

End of procedure

Next Steps

- Create the Configuration Layer Application object for the Configuration Synchronization Component. See, Procedure: Configuration Synchronization Component: Creating the Configuration Synchronization Component Application object.

Creating the Configuration Synchronization Component Application Object

This section describes how to create the Configuration Synchronization Component Application object.

Procedure:
Configuration Synchronization Component: Creating the Configuration Synchronization Component Application object

Purpose: To create the Application object for the Configuration Synchronization Component.

Start of procedure

1. In Configuration Manager, under Environment, right-click the Applications folder.

2. Select New > Application.

   Select the Application template that you just created:
   - Release 8.0.0:
     Gplus_SiebelCRM_Config_Synch_800_for_CL_71_and_higher.apd
   - Release 8.0.1:
     Gplus_SiebelCRM_Config_Synch_801_for_CL_71_and_higher.apd
   - Release 8.0.2:
     Gplus_SiebelCRM_Config_Synch_802_for_CL_71_and_higher.apd

   or
   - Release 8.0.0: Gplus_SiebelCRM_Config_Synch_800_for_CL_70.apd
   - Release 8.0.1: Gplus_SiebelCRM_Config_Synch_801_for_CL_70.apd
   - Release 8.0.2: Gplus_SiebelCRM_Config_Synch_802_for_CL_70.apd

3. Click OK.

   The Properties dialog box for the Application object appears.

End of procedure

Next Steps

- Configure the tabs in the Properties dialog box for the Configuration Synchronization Component. See, Procedure: Configuration Synchronization Component: Configuring the tabs in the Properties dialog box, on page 86.

Configuring the Tabs in the Properties Dialog Box

This section describes how to configure the tabs in the Properties dialog box, arranged in the order in which they display.

Procedure:
Configuration Synchronization Component:
Configuring the tabs in the Properties dialog box

Purpose: To configure the tabs in the Properties dialog box for the Configuration Synchronization Component.

Start of procedure

General Tab   1. Click the General tab in the Properties dialog box (see Figure 8 on page 86).
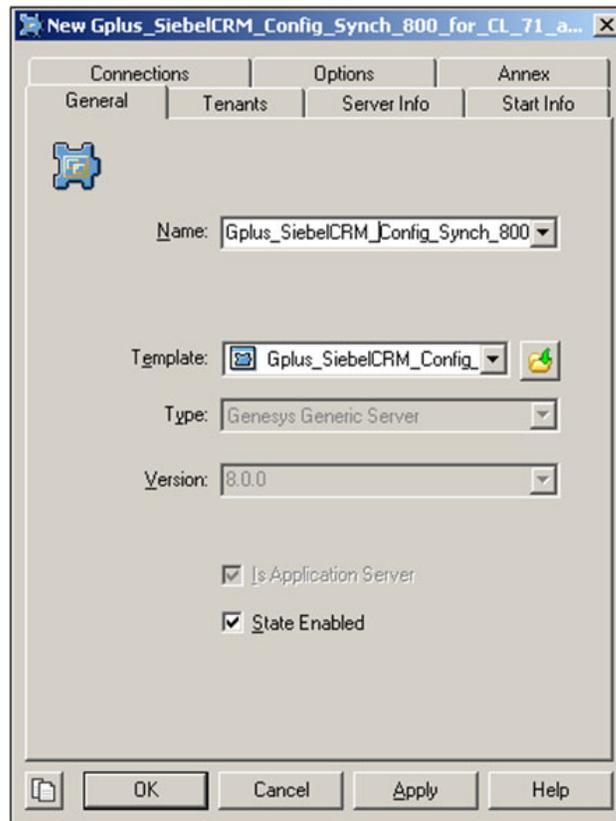
Figure 8: View of General Tab in Multi-Tenant Environment

2. In the Name list, click the name of the Application object that you are configuring.

3. In the Template list, click the name of the template that you are configuring or use the Browse button to select the template that you are configuring.

4. Click Apply.

5. Next, if you are working in a multi-tenant environment, go to the Tenants Tab; otherwise, go to the Server Info Tab.

Note: The Tenants tab only displays, if you are working in a multi-tenant environment.

Tenants Tab   **6.** Select the Tenants tab.

7. Select the Genesys Tenants under which the objects that are exported from Siebel are created.

8. Click Add.

Server Info Tab   **9.** Select the Server Info tab.

10. In the Host field, use the Browse button to select the host on which you are installing the Configuration Synchronization Component, and click OK.

11. In the Port list, add a port with any valid port number.

Note: This option determines the HTTP port number that is used by the Configuration Synchronization Component to receive HTTP packets from Siebel. You will use the value that you enter for this option when you configure the Siebel part of the Configuration Synchronization Component.

Start Info Tab   **12.** Select the Start Info tab.

13. In the Working Directory field, enter the full name of the Component installation directory on the host that you specified on the Server Info tab. The value that you enter in this field is used as the default destination folder during installation.

14. Enter any valid value into each of the following fields:
- Command Line
- Command Line Arguments

The values that you enter in these fields are overwritten during installation; however, data must be present in these fields during the configuration process.

15. Leave the default values for the remaining fields.

Options Tab   **16.** Select the Options tab (see Figure 9 on page 88).
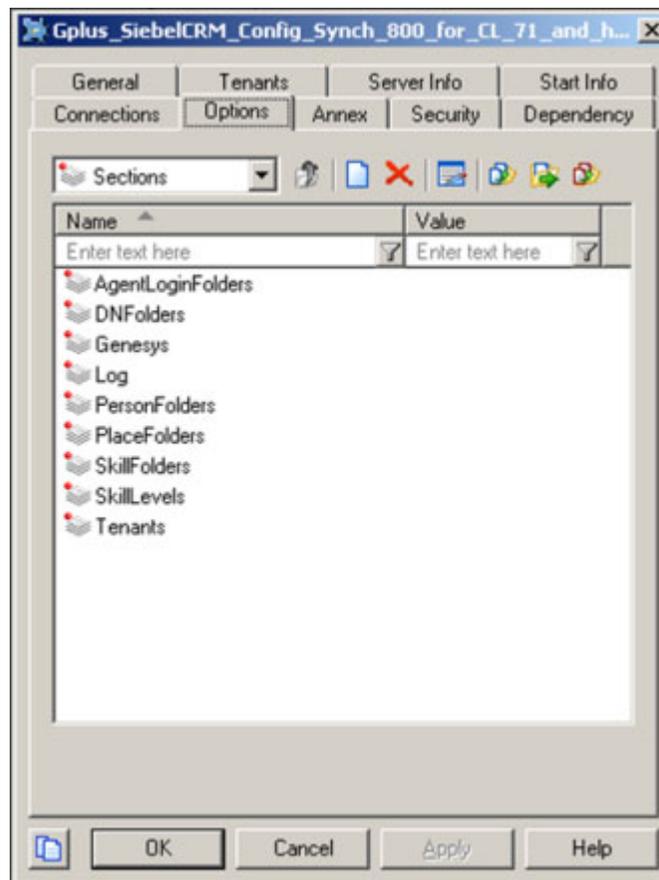
Figure 9: View of Options Tab in a Multi-Tenant Environment

In the Sections pane, the following sections are listed:

- AgentLoginFolders
- DNFolders
- Genesys
- Log
- PersonFolders
- PlaceFolders
- SkillFolders
- SkillLevels
- Tenants

17. Double-click the Genesys section to start.

18. Configure the configuration options that are described in the following section, "Setting the Genesys Configuration Options for the Configuration Synchronization Component" on , starting with the Log section.

End of procedure

Next Steps

- Set the configuration options for the Configuration Synchronization Component. See, "Setting the Genesys Configuration Options for the Configuration Synchronization Component".

Setting the Genesys Configuration Options for the Configuration Synchronization Component

Unless specified otherwise, set the Configuration Synchronization Component configuration options in the Options tab of the Application object using the following navigation path:

In Genesys Adminstrator—Application object > Options tab > Advanced View (Options)

In Configuration Manager—Application object > Properties dialog box > Options tab.

For ease of reference, the options have been arranged in alphabetical order within their corresponding sections:

AgentLoginFolders Section

Each option in the AgentLoginFolders section controls the mapping of a Siebel User ID to a Configuration Manager folder, where the Siebel user utilizes this User ID to create an Agent Login object when importing the information from Siebel.

Example: AgentLogin Folders Option

You want to create an AgentLoginFolders option in the following scenario:

- You have a switch named mySwitch defined in Genesys Configuration Manager.
- You want the Siebel user with a User ID of user1, to create an Agent Login object under the Configuration Manager folder, Folder1, which is a subfolder of the Agent Logins folder under the switch, mySwitch.

Then create a new option in the AgentLoginFolders section with user1 as the option name and /Agent Logins/Folder1 as the option value.

All Configuration Manager folders referenced in this section should be created in Genesys Configuration Manager prior to importing the information from Siebel.

The ratio of Siebel User IDs to Genesys Configuration Manager folders in this section is many-to-one.

Note: You can configure the import function, using the options in this section, to assign different Configuration Manager folders in Genesys to various Siebel users. However, if the same Siebel data is imported into different Configuration Manager folders, errors during import will occur due to uniqueness constraints imposed by the Genesys Configuration Layer. These errors result in Siebel and Genesys data being desynchronized. To avoid such errors, different Siebel users should import the same Siebel data into the same set of Configuration Manager folders. Alternatively, if each Siebel user has a unique folder assignment, different Siebel users should not import the same Siebel data into Genesys.

DNFolders Section

Each option in the DNFolders section controls the mapping of a Siebel User ID to a Configuration Manager folder, where the Siebel user with this User ID creates a DN object when importing information from Siebel.

**Example:
DNFolders Option**

You want to create a DNFolders section option in the following scenario:

- You have a switch named mySwitch defined in Genesys Configuration Manager.

- You want the Siebel user with the User ID of user1, to create a DN object under the Configuration Manager folder, Folder1, which is a subfolder of the DNs folder under the switch, mySwitch.

Then create a new option in the DNFolders section with user1 as the option name and /DNs/Folder1 as the option value.

All Configuration Manager folders referenced in this section should be created in Genesys Configuration Manager prior to importing information from Siebel.

The ratio of Siebel User IDs to Genesys Configuration Manager folders in this section is many-to-one.

Note: You can configure the import function, using options in this section to assign different Configuration Manager folders in Genesys to various Siebel users. However, if the same Siebel data is imported into different Configuration Manager folders, errors during import will occur due to uniqueness constraints imposed by the Genesys Configuration Layer. These errors result in Siebel and Genesys data being desynchronized. To avoid such errors, different Siebel users should import the same Siebel data into the same set of Configuration Manager folders. Alternatively, if each Siebel user has a unique folder assignment, different Siebel users should not import the same Siebel data into Genesys.

Genesys Section

The Genesys section has various options that can be configured. The following section describes the possible options.

password

Default Value: none
Valid Value: none

Specifies the password for the login user name above, for example—GenesysAdapter. This user name should be created in the Siebel environment. The user must have access to the Siebel database as a Siebel Administrator. For more information about Siebel user administration, see your Siebel documentation.

Note: Genesys does not recommend using a common login for this process.

ServerURL

Default Value: none
Valid Value: none

Specifies the URL of the web server that has the Siebel Web Engine installed. Initially, this option contains the following template string:

ServerURL=<ServerUrl>/<path>/start.swe

where:

<ServerUrl>:	is the first part of the URL that you use to access your Siebel Server through the Siebel Web Client.
<path>:	specifies a virtual path on the server referring to the specific Siebel Web Extension configuration. The default value for this option is eai_enu.

You must edit this URL template string according to your Siebel Web Extension configuration—for example:

http://www.myserver.com/eai_enu/start.swe

SkipSkillItemTypeName

Default Value: false
Valid Value: false, true

Specifies how skill names are generated, and specifically if Skill Item Type names (Skill names) are written or not. By default, this value is set to false, which means that the Skill Item Type names (Skill names) are written in Genesys Configuration Manager during synchronization. If you need to omit Skill Item Type names (Skill names), set the value of this option to true.

SWEExtSource

Default Value: GplusConfSynchExportAlldata
Valid Value: Any valid string value

Refers to an entry in the [HTTP Services] section of the eai.cfg file found within your Siebel Server installation. This entry provides the mapping to a Siebel Named Subsystem record, which in turn describes the Business Service call.

username

Default Value: <required>
Valid Value: A valid username created in a Siebel environment

Specifies a Siebel user name for the Siebel Object Manager login to process a request from the Adapter, for example—GenesysAdapter.

This user name must be created in the Siebel environment. The user must have access to the Siebel database as Siebel Administrator.

Note: Genesys does not recommend using a common login ID for this process.

For more information about Siebel user administration, refer to your Siebel documentation.

UseSeparatorsForSkillName

Default Value: false, true
Valid Value: false, true

Specifies how skill names are generated, and specifically how the names are separated. By default, this value is set to false, signifying that separators are not used. If you need to generate skill names as they were in version 6.5, set the value of this option to true.

xsl-transformer (optional)

Default Value: <optional>

Valid Value: none

This configuration option is optional. Use this option, if you want to customize the values of the synchronized data, such as Phone number, using your own rules.

The Adapter provides a mechanism to customize the transformation rules through an *.xsl file. In this case, you must add this option to the Genesys section. The value of this option should be equal to the name of the.xsl file.

Log Section

The Configuration Synchronization Component supports a common set of Log options to allow the precise configuration of the log file output. For a complete list of Common Log options and their descriptions, see the "Common Log Options" chapter of the *Framework 8.0 Configuration Options Reference Manual*.

Note: If you do not specify any log options, the default values apply.

PersonFolders Section

Each option in the PersonFolders section controls the mapping of a Siebel User ID to a Configuration Manager folder, where the Siebel user with this User ID create a Person object when importing information from Siebel.

Example: PersonFolders Option

You want to create a PersonFolders section option in the following scenario:

*   You have a Tenant named myTenant defined in Genesys Configuration Manager.

*   You want the Siebel user with the User ID of user1, to create a Person object under the Configuration Manager folder, Folder1, which is a subfolder of the Persons folder under the Tenant, myTenant.

Then create a new option in the PersonFolders section with user1 as the option name and /Persons/Folder1 as the option value.

Make sure that the Siebel Organization or Division whose objects Siebel user user1 will import is mapped to the Tenant, myTenant (see "Tenants Section" on).

All Configuration Manager folders referenced in this section should be created in Genesys Configuration Manager prior to importing information from Siebel.

The ratio of Siebel User IDs to Genesys Configuration Manager folders in this section is many-to-one.

Note: You can configure the import function using the options in this section, to assign different Configuration Manager folders in Genesys to various Siebel users. However, if the same Siebel data is imported into different Configuration Manager folders, errors during import will occur due to uniqueness constraints imposed by the Genesys Configuration Layer. These errors result in Siebel and Genesys data being desynchronized. To avoid such errors, different Siebel users should import the same Siebel data into the same set of Configuration Manager folders. Alternatively, if each Siebel user has a unique folder assignment, different Siebel users should not import the same Siebel data into Genesys.

PlaceFolders Section

Each option in the PlaceFolders section controls the mapping of a Siebel User ID to a Configuration Manager folder, where the Siebel user with this User ID will create Place objects when importing information from Siebel.

Example: PlaceFolders Option

You want to create a PlaceFolders section option in the following scenario:

- You have a Tenant named myTenant defined in Genesys Configuration Manager.

- You want the Siebel user with the User ID of user1, to create a Place object under the Configuration Manager folder, Folder1, which is a subfolder of the Persons folder under the Tenant, myTenant.

Then create a new option in the PlaceFolders section with user1 as the option name and /Places/Folder1 as the option value.

Make sure that the Siebel Organization or Division whose objects the Siebel user user1 will import is mapped to the Tenant, myTenant (see "Tenants Section" on).

All Configuration Manager folders referenced in this section should be created in the Genesys Configuration Manager prior to importing information from Siebel.

The ratio of Siebel User IDs to Genesys Configuration Manager folders in this section is many-to-one.

Note: You can configure the import function, using the options in this section, to assign different Configuration Manager folders in Genesys to various Siebel users. However, if the same Siebel data is imported into different Configuration Manager folders, errors during import will occur due to uniqueness constraints imposed by the Genesys Configuration Layer. These errors result in Siebel and Genesys data being desynchronized. To avoid such errors, different Siebel users should import the same Siebel data into the same set of Configuration Manager folders. Alternatively, if each Siebel user has a unique folder assignment, different Siebel users should not import the same Siebel data into Genesys.

SkillFolders Section

Note: If the SkillFolders Application option contains an incorrect value (for example—a folder which does not exist), the Person, even without any skill, will not be exported from Siebel to Genesys.

Each option in the SkillFolders section controls the mapping of a Siebel User ID to a Configuration Manager folder, where the Siebel user with this User ID will create a Skill object when importing information from Siebel.

Example: SkillFoldersOption

You want to create a SkillFolders section option in the following scenario:

- You have a Tenant named myTenant defined in Genesys Configuration Manager.

- You want the Siebel user with the User ID of user1, to create a Skill object under the Configuration Manager folder, Folder1, which is a subfolder of the Skills folder under the Tenant, myTenant.

Then create a new option in the SkillFolders section with user1 as the option name and /Skills/Folder1 as the option value.

Make sure that the Siebel Organization or Division whose objects Siebel user user1 will import is mapped to the Tenant, myTenant (see "Tenants Section" on).

All Configuration Manager folders referenced in this section should be created in Genesys Configuration Manager prior to importing information from Siebel.

The ratio of Siebel User IDs to Genesys Configuration Manager folders in this section is many-to-one.

Note: You can configure the import function, using the options in this section, to assign different Configuration Manager folders in Genesys to various Siebel users. However, if the same Siebel data is imported into different Configuration Manager folders, errors during import will occur due to uniqueness constraints imposed by the Genesys Configuration Layer. These errors result in Siebel and Genesys data being desynchronized. To avoid such errors, different Siebel users should import the same Siebel data into the same set of Configuration Manager folders. Alternatively, if each Siebel user has a unique folder assignment, different Siebel users should not import the same Siebel data into Genesys.

SkillLevels Section

Each option in the SkillLevels section define the mapping of Siebel Skill Expertise values to Genesys Skill Level values. To map a Siebel Skill Expertise value to a Genesys Skill Level value, create a new option within this section, where the option name is the value of the Siebel Skill Expertise and the option value is the value of the Genesys Skill Level. The default mapping is shown in Figure 10.
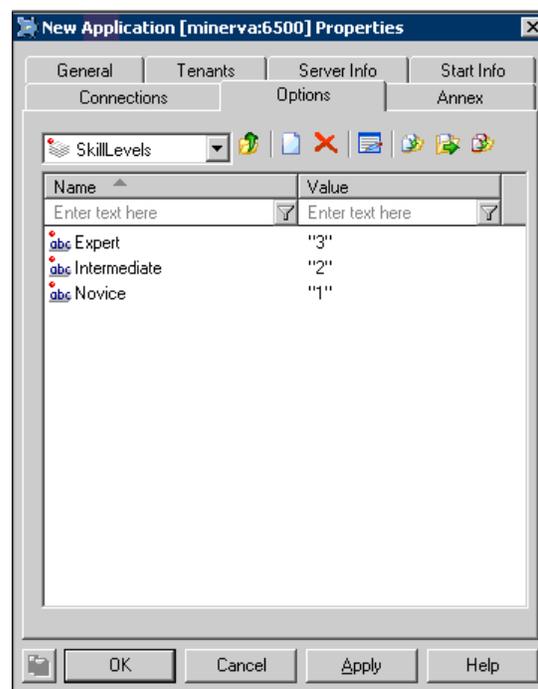
Figure 10: The Default Mapping of Siebel Skill Expertise Values to Genesys Skill Levels

After configuring the Application options, click OK to save the Application object.

Tenants Section

Options in the Tenants section define the mapping of Siebel Organizations or Divisions to Genesys Tenants. To map a Siebel Organization or Division to a Genesys Tenant, create a new option within this section, where the option name is the name of the Genesys Tenant and the option value is the name of the Siebel Organization or Division.

Example: TenantsOption

If you have an Organization named myOrganization in Siebel and a Tenant named myTenant in Genesys, enter myTenant as the option name and myOrganization as the option value. Then all Siebel objects that belong to myOrganization Organization in Siebel are created under the Tenant, myTenant in Genesys.

The relationship of Siebel Organizations or Divisions to Genesys Tenants is one-to-one.

Note: If you are working in a single-tenant environment, this section contains one option, where the option name is Resources and the option value is the name of your Siebel Organization or Division.

Saving the Application Object

After setting the Application options, click OK to save the Application object.

Configuring the Security Settings

This section describes how to configure the security settings for the Configuration Synchronization Component. The security settings must be configured, so that the Component can make changes to the Genesys Configuration Layer objects.

Follow the procedures outlined in this section to configure the security settings for the Application object that you created in the section, "Creating the Configuration Synchronization Component Application Object" on .

Procedure:
Configuration Synchronization Component: Creating a new Person object

Purpose: To add a new Person object to the environment.

Start of procedure

1. In Configuration Manager:
   - Right-click the Persons folder under Environment, if you are working in a multi-tenant environment.
   - Right-click the Persons folder under Resources, if you are working in a single-tenant environment.

2. Select New > Person.
   The new Person window displays.

3. Select the General tab and enter the following parameters:
   - Employee ID = siebel7gplus
   - User Name = siebel7gplus
   - Is Agent = Clear

4. Click OK.

End of procedure

Next Steps

- Add a Person object to the Super Administration Access Group. See, Procedure: Configuration Synchronization Component: Adding a Person object to the Super Administrators Access Group.

Procedure:
Configuration Synchronization Component: Adding a Person object to the Super Administrators Access Group

Purpose: To add a Person object to the Super Administrators Access Group.

Start of procedure

1. In Configuration Manager, to display the existing Access Groups:
   - Right-click the Access Groups folder under Environment, if you are working in a multi-tenant environment.
   - Right-click the Access Groups folder under Resources, if you are working in a single-tenant environment.

2. Double-click the Super Administrators Access Group to display its properties.

3. Click the Add button to add the Person object that you just created to the Users list.

4. Click OK.

End of procedure

Next Steps

- Associate the newly created Person object account with the Application object. See, Procedure: Configuration Synchronization Component: Associating a Person object account with the Application object.

Procedure:
Configuration Synchronization Component: Associating a Person object account with the Application object

Purpose: To associate a Person object account with the Application object.

Start of procedure

In Configuration Manager, under Environment:

1. Click the Applications folder to display the existing applications.

2. Double-click the Application object that you just created to display its properties.

3. Select the Security tab, in the Log On As section, and select This Account. The Add User window displays.

4. Select the Person account that you just created and click Add.

5. Click OK to close the Add User window.

6. Click OK to save the Application object.

This completes the Genesys part of the configuration process. Now you are ready to install the Configuration Synchronization Component.

End of procedure

Next Steps

- Install the Configuration Synchronization Component. See the section, "Installation" on page 100

Installation

This section describes the installation process for the G*plus* Adapter 8.0 for the Siebel CRM Configuration Synchronization Component. Select one of the following sections, depending on your environment:

-
-

Installing the Configuration Synchronization Component in a Windows Environment

This section describes how to install the Configuration Synchronization Component on a Windows system.

Procedure:
Configuration Synchronization Component: Installing the Configuration Synchronization Component in a Windows environment

Purpose: To install the Configuration Synchronization Component on a Windows operating system.

Start of procedure

1. To start the installation process, run the setup.exe file from the Configuration Synchronization Component installation package.

2. In the Welcome window, click Next.

3. In the Configuration Parameters for the Genesys Configuration Server window, enter the following:
   - The name of the host on which Configuration Server is running.
   - The communication port that client applications must use to connect to Configuration Server.
   - The user name used to access Configuration Server.
   - The password used to access Configuration Server.

4. Click Next.

5. In the Select Application window, select the Application object by its corresponding number that you configured in the previous procedure, .

6. Click Next.

7. In the Choose Destination Location window, click Next to accept the default destination folder or use the Browse button to select a different destination folder.

Note: If you want to deploy both the Configuration Synchronization Component and the Campaign Synchronization Component, you must specify a unique destination folder for each component. This prevents the installation package of one component from being overwritten by the installation package of the other component.

---

8. In the Ready to Install window, click Install.

9. In the Setup Complete window, click Finish.

The Configuration Synchronization Component is now installed.

In the G*plus* Adapter for Siebel CRM program folder in the Start menu, you can see that the installer created a shortcut for the Configuration Synchronization Component.

End of procedure

Next Steps

• Configure the Siebel part of the Configuration Synchronization Component. See the section, "Configuring Siebel" on page 103.

Installing the Configuration Synchronization Component in a UNIX Environment

This section describes how to install the Configuration Synchronization Component on an UNIX operating system.

Note: Do not use special symbols in any destination directory name when installing in an UNIX environment.

Procedure:
Configuration Synchronization Component: Installing the Configuration Synchronization Component in a UNIX environment

Purpose: To install the Configuration Synchronization Component on a UNIX system.

Start of procedure

1. In the directory in which the Configuration Synchronization Component installation package was copied, locate a shell script named: install.sh.

2. Run this script from the command prompt by typing sh and the file name:

    sh install.sh

3. When prompted, specify the host name, port, user name, and password of the computer on which the Component will be installed—for example:

    a. Enter the host name of Configuration Server.

    b. Enter the port of Configuration Server.

    c. Enter the user name for Configuration Server.

    d. Enter the password for the user name.

4. Choose the Configuration Server environment by its corresponding number.

5. Specify the Application object you configured in the previous procedure, Procedure: Configuration Synchronization Component: Creating the Configuration Synchronization Component Application object, on page 85.

6. Specify the full path to the destination directory in which you want the Component to be installed.

Note: If you want to deploy both the Configuration Synchronization Component and Campaign Synchronization Component, you must specify a unique destination folder for each component. This prevents the installation package of one component from being overwritten by the installation package of the other component.

The Configuration Synchronization Component is now installed.

In the G*plus* Adapter for Siebel CRM program folder in the Start menu, you can see that the installer created a shortcut for the Configuration Synchronization Component.

End of procedure

Next Steps

•   Configure the Siebel side of the Configuration Synchronization Component. See the section, "Configuring Siebel".

Uninstalling the Configuration Synchronization Component

Complete information about uninstalling the G*plus* Adapter and its components is provided in Chapter 14, "Uninstallation Instructions," on page 565.

Configuring Siebel

This section describes how to configure the Siebel part of the Configuration Synchronization Component for Siebel CRM.

Configuring the Siebel section of the Configuration Synchronization Component consists of the following sections:

Prestart Information

Before starting this part of the configuration process, you must make sure to do the following:

- You must have the Siebel Tools application running.

- You should be connected to the local copy of the Siebel Server database.

- You should have sufficient privileges to check-in and check-out projects from the Siebel Server repository.

- You should ensure that all required steps from the "Patching and Configuring Siebel CRM" section are performed.

Configuring Siebel Using Siebel Tools

You will use Siebel Tools to compile an updated version of the Siebel repository file for one, or more, of the Siebel applications that you use on your Siebel Server, which you will then deploy on the server. For more information about using Siebel Tools, see the Siebel documentation.

Configuring Siebel using the Siebel Tools consists of the following sub-sections:

Importing the GenesysConfigSynchronization.sif Archive File

During this step, you will import objects into the Siebel repository that are part of the Siebel implementation of the Configuration Synchronization Component. The objects are added to the Genesys Config Synchronization project in Siebel Tools.

Note: If the Genesys Config Synchronization project already exists in your Siebel repository file, lock it before importing the GenesysConfigSynchronization.sif archive.

If this project does not exist before export, lock it after the export.

Procedure:
Configuration Synchronization Component: Importing the GenesysConfigSynchronization.sif archive file for the Configuration Synchronization Component

Purpose: To import the GenesysConfigSynchronization.sif archive file for the Configuration Synchronization Component.

Start of procedure

1. In Siebel Tools, select Tools > Import from Archive... .

2. In the Select Archive to Import window, navigate to the GenesysConfigSynchronization.sif archive file. This file was created by the installation program in: <target directory>/objects/<Siebel Version> where <Siebel Version> is the subfolder name (7.7, /8.0, /8.1, /8.1_8.2_OUI or IP2014), depending on the Siebel version that you use.

3. Click Open.
   The Import Wizard–Preview window displays.

4. In the Import Wizard–Preview window, in the Conflict Resolution section, select Overwrite the object definition in the repository.

5. Click Next.
   The Import Wizard–Review Conflicts and Actions window displays.

6. Click Next.
   The Do you wish to proceed? window displays.

7. Click Yes.
   The objects from the archive are imported into the Siebel repository.

8. Click Finish to complete the Import Wizard.

Note: It is possible for siebel_assert_xxx.txt file(s) to be generated by the Siebel environment after importing the archive. This does not affect the functionality of the Configuration Synchronization Component in any way.

End of procedure

Next Steps

- Check-out an existing project from the Siebel repository. See, Procedure: Configuration Synchronization Component: Checking out an existing project from the Siebel Repository for the Configuration Synchronization Component, on page 105

Checking Out an Existing Project from the Siebel Repository for the Configuration Synchronization Component

The Siebel implementation of the Configuration Synchronization Component makes use of runtime events. To automate the process of creating runtime events, which are required for Configuration Synchronization Component, you must modify the Personalization Action Set business component. To modify this object, you must check-out the corresponding project from the Siebel repository.

Procedure:
Configuration Synchronization Component: Checking out an existing project from the Siebel Repository for the Configuration Synchronization Component

Purpose: To check out an existing project from the Siebel repository.

Start of procedure

1. In Siebel Tools, select Tools > Check Out...
   The Check Out dialog box displays.

2. Select Project category.

3. In the Projects list, select the Personalizations project.

4. Click Check Out.

End of procedure

Next Steps

- Exporting the Siebel extension of type S as DNs of type Extension or ACD Positions in Genesys. See, Procedure: Configuration Synchronization Component: Exporting Siebel extensions of type S as DNs of type Extensions or ACD Positions in Genesys.

Defining the Export of Siebel Extensions of Type S as DNs of Type Extensions or ACD Positions in Genesys

The Configuration Synchronization Component can export Siebel extensions of type S to Genesys as DNs of type Extensions or ACD Positions. By default, the Siebel extensions of type S are exported as Genesys DNs of type ACD Position.

Procedure:
Configuration Synchronization Component: Exporting Siebel extensions of type S as DNs of type Extensions or ACD Positions in Genesys

Purpose: To set up the export of Siebel extensions of type S as DNs of type Extensions in Genesys.

Start of procedure

1. In Siebel Tools, navigate to Object Explorer \ Business Service folder/applet.
2. Select <Genesys Config Synchronization> Business Service.
3. For this Business Service, go to the Business Service User Props applet.
4. Select the record with Name <ConvertExt2ACD>.
5. Instead of True (default), set False as a value for this record.

End of procedure

Next Steps

- Import the PersonalizationActioSet.sif archive into the Siebel repository. See, Procedure: Configuration Synchronization Component: Importing the PersonalizationActionSet.sif archive file into the Siebel Repository.

Importing the PersonalizationActionSet.sif Archive File into the Siebel Repository for the Configuration Synchronization Component

This section describes how to import the PersonalizationActionSet.sif archive file into the Siebel repository.

Procedure:
Configuration Synchronization Component: Importing the PersonalizationActionSet.sif archive file into the Siebel Repository

Purpose: To import the PersonalizationActionSet.sif archive file into the Siebel repository.

Start of procedure

1. In Siebel Tools, select Tools > Import from Archive... .

2. In the Select Archive to Import window, navigate to the PersonalizationActionSet.sif archive file. This file was created by the installation program in: <target directory>/objects/<Siebel Version> where <Siebel Version> is the subfolder name (7.7, /8.0, /8.1, /8.1_8.2_OUI or IP2014), depending on the Siebel version that you use.

3. Click Open.
   The Import Wizard–Preview window displays.

4. In the Import Wizard–Preview window, in the Conflict resolution section, select Merge the object definition from the archive file with the definition in the repository.

5. Click Next.
   The Do you wish to proceed? window displays.

6. Click Yes.
   The objects from the archive are imported into the Siebel repository.

7. Click Finish to complete the Import Wizard.

Note: It is possible for siebel_assert_xxx.txt file(s) to be generated by the Siebel environment after importing the archive. This does not affect the functionality of the Component in any way.

End of procedure

Next Steps

- Compile the Siebel Repository File. See, Procedure: Configuration Synchronization Component: Compiling the Siebel Repository File.

Compiling the Siebel Repository File for the Configuration Synchronization Component

This section describes how to compile the Siebel repository file for the Configuration Synchronization Component.

Procedure: Configuration Synchronization Component: Compiling the Siebel Repository File

Purpose: To compile the Siebel repository File.

Start of procedure

1. In Siebel Tools, select Tools > Compile Projects...

2. Select Locked projects.

3. In the Siebel Repository File edit box, enter the name of the repository file.

4. Click Compile.
   The status bar at the bottom of the Object Compiler window indicates when the compilation is finished.

5. When the compilation is finished, close Siebel Tools.

Note: It is possible for siebel_assert_xxx.txt file(s) to be generated by the Siebel environment after compiling the Siebel repository file. This does not affect the functionality of the Component in any way

End of procedure

Next Steps

- Configure the Siebel Server to work with the Configuration Synchronization Component. See the section, "Configuring the Siebel Server".

Configuring the Siebel Server

Complete the following procedure, described in this section, to configure the Siebel Server to work with the Configuration Synchronization Component.

Procedure:
Configuration Synchronization Component:
Configuring the Siebel Server

Purpose: To configure the Siebel Server to work with the Configuration Synchronization Component.

Start of procedure

Updating the Configuration Files

1. To update the eai.cfg file in the Siebel Server installation, open the file and add the following line to the [HTTP Services] section:

    GplusConfSynchExportAllData = GplusConfSynchExportAllData.

Configuring the Siebel Inbound HTTP Transport

2. The Configuration Synchronization Component used the Siebel Inbound HTTP Transport. For instructions about how to configure the Siebel Inbound HTTP Transport, refer to the Siebel documentation.

Deploying the Siebel Repository File

3. Deploy the compiled Siebel repository file on your Siebel Server. Additionally, you might have to generate and deploy the browser scripts for the new repository file.

    For further information on deploying an updated repository file to the Siebel Server, refer tot he Siebel documentation.

End of procedure

Next Steps

*   Configure Siebel using the Siebel Web Client. See the section, "Configuring Siebel Using the Siebel Web Client".

Configuring Siebel Using the Siebel Web Client

You must connect to your Siebel Server using the Siebel Web Client.

Note: You must be logged in as a Siebel Administrator.

This part of the configuration process includes the following steps:

Checking the Status of the EAI Server Component Group

You must check the status of the EAI (Enterprise Application Integration) Server Component Group on your Siebel Server. There are slight variations in the navigation required by this process, depending on the version of Siebel with which you are working.

Procedure:
Configuration Synchronization Component: Checking the EAI Server Component Group status

Purpose: To check the status of the EAI Server Component Group on your Siebel Server.

Start of procedure

1. Navigate through the Site Map to Administration - Server Configuration > Component Groups.

2. Select the Siebel Server on which you are working. For the selected server, locate the Enterprise Application Integration Component Group in the Server Component Groups applet. Verify that the value for the Enable State group is Enabled.

End of procedure

Next Steps

*   Create a Siebel Enterprise Server named subsystem. See the section, "Creating a Siebel Enterprise Server Named Subsystem".

Creating a Siebel Enterprise Server Named Subsystem

You must configure one connection subsystem that is used by the Configuration Synchronization Component.There are slight variations in the navigation that is required by this process, depending on the version of Siebel with which you are working.

Procedure:
Configuration Synchronization Component: Navigating to the Siebel Server configuration

Purpose: To navigate through the site map to get to your Siebel Server configuration.

Start of procedure

1. Navigate through the site map to Administration - Server Configuration.

2. Select Profile Configuration in the applet within the view.

3. Complete the process by completing the procedures that follow.

End of procedure

Next Steps

*   Create a EAI Transport Data Handling subsystem. See, "Configuration Synchronization Component: Creating the EAI Transport Data Handling subsystem".

Procedure:
Configuration Synchronization Component: Creating the EAI Transport Data Handling subsystem

Purpose: To create a EAI Transport Data Handling subsystem.

Start of procedure

1. In the Profile Configuration applet, create a new record.

2. Specify the following parameters for the new record:
    *   Alias: GplusConfSynchExportAllData
    *   Description: Gplus Configuration Synchronization Component: ExportAllData
    *   Profile: Gplus Configuration Synchronization Component ExportAllData Parameters
    *   Subsystem Type: EAITransportDataHandlingSubsys

3. Save the new record.

4. For the Named Subsystem record that you just created, enter the following parameters in the Profile Parameters applet:
    *   DispatchMethod: ExportAllData

• DispatchService: Genesys Config Synchronization

End of procedure

Next Steps

• Create a connection subsystem. See, on .

Procedure:
Configuration Synchronization Component: Creating a connection subsystem

Purpose: To create and configure one connection subsystem that is used by the Configuration Synchronization Component.

Start of procedure

Creating a Connection Subsystem

1. In the Profile Configuration applet, create a new record.

2. Specify the following parameters for the new record:
   • Alias: GplusConfSyncConnectionPrimary
   • Description: Gplus Configuration Synchronization Component Primary Connection
   • Profile: Gplus Configuration Synchronization Component Primary Connection Parameters
   • Subsystem Type: HTTPSubSys

3. Save the new record.

4. For the Named Subsystem record that you just created, enter the following parameters in the Profile Parameters applet:
   • HTTPRequestMethod: POST
   • HTTPRequestURLTemplate: http://<host>:<port>

   where:

   <host> is the host on which you installed the primary Configuration Synchronization Component.

   <port> is the port that you specified with the Communications Port option on the Server Info tab of the primary Configuration Synchronization Component Application object.

Creating a Backup Connection Subsystem

5. In the Profile Configuration applet, create a new record.

6. Specify the following parameters for the new record:
   • Alias in Siebel: GplusConfSyncConnectionBackup

- • Description: Gplus Configuration Synchronization Component Backup Connection
- • Profile: Gplus Configuration Synchronization Component Backup Connection Parameters
- • Subsystem Type: HTTPSubSys

7. Save the new record.

8. For the Named Subsystem record that you just created, enter the following parameters in the Profile Parameters applet:
   - • HTTPRequestMethod: POST
   - • HTTPRequestURLTemplate: http://<host>:<port>

   where:

   <host> is the host on which you installed the backup Configuration Synchronization Component.

   <port> is the port that you specified with the Communications Port option on Server Info tab of the backup Configuration Synchronization Component Application object.

End of procedure

Next Steps

- • Create the runtime events that are required by the Configuration Synchronization Component. See the section, "Creating Runtime Events".

Creating Runtime Events

To create the runtime events that are required by the Configuration Synchronization Component, you must run the InstallRunTimeEvents method of the Genesys Configuration Synchronization Siebel Business Service using the Siebel Business Service Simulator applet.

Note: The runtime events that are used by the Configuration Synchronization Component replace functions that are performed by scripts in the older versions of the Component (7.0, and older).

Procedure:
Configuration Synchronization Component: Creating the runtime events

Purpose: To create the runtime events that are required by the Campaign Synchronization Component.

Start of procedure

1. Navigate through the site map to Business Service Administration > Business Service Simulator applet.

2. In the Business Service Simulator applet, create a new record.

3. Specify the following parameters for the new record:
    *   Service Name: Genesys Config Synchronization
    *   Method Name: InstallRunTimeEvents

4. Click the Run button.

5. In the Siebel client, navigate to Administration - Runtime Events > Events.

6. Select the Reload Runtime Events system menu item.

End of procedure

Next Steps

*   You have completed the configuration and installation of the Configuration Synchronization Component.

Business Service—User Properties

This section describes the Genesys Configuration Synchronization User Properties of this Business Service.

Customers can redefine the default values of the User Properties that are specified in Table 8. These values are used during configuration synchronization. Default values are useful when all configuration objects must have the same values.

Table 8: User Properties of Genesys Config Synchronization Business Service

Name	Default Value	Comments
ConvertExt2ACD	true	
DefaultAgentLoginOverride		Empty string
DefaultAgentLoginState	1	CfgObjectState (1 = Enabled)
DefaultAgentLoginSwitchSpecific Type	1	
DefaultAgentLoginUseOverride	2	CfgFlag (2 = true)

Table 8: User Properties of Genesys Config Synchronization Business Service (Continued)

Name	Default Value	Comments
DefaultAgentLoginWrapupTime	1	
DefaultDnAssociation		Empty string
DefaultDnLoginID		Empty string
DefaultDnNumberOfTrunks	0	
DefaultDnOverride		Empty string
DefaultDnRegisterAll	2	CfgDNRegisterFlag (2 = true)
DefaultDnRouteType	1	CfgRouteType (1 = Default)
DefaultDnState	1	CfgObjectState (1 = Enabled)
DefaultDnSwitchSpecificType	1	
DefaultDnUseOverride	2	CfgFlag (2 = true)
HTTP Connection Subsystem (Primary)	GplusConfSync Connection Primary	
HTTP Connection Subsystem (Backup)	GplusConfSync Connection Backup	
Library Name	GenCommDrv, GenCommBC	Comma-separated list of driver library names.

Data-Type Definitions

The following are definitions of data types that are used in Table 8:

- CfgObjectState
- CfgDNRegisterFlag
- CfgRouteType

CfgObjectState

0 - No Object State/CFGNoObjectState

1 - Enabled/CFGEnabled

2 - Disabled/CFGDisabled

3 - Deleted/CFGDeleted

CfgDNRegisterFlag

0 - Unknown/CFGDRUnknown

1 - False/CFGDRFalse

2 - True/CFGDRTrue

3 - OnDemand/CFGDROnDemand

CfgRouteType

0 - Unknown/CFGNoRoute

1 - Default/CFGDefault

2 - Label/CFGLabel

3 - Overwrite DNIS /CFGOverwriteDNIS

4 - DDD/CFGDDD

5 - IDDD/CFGIDDD

6 - Direct/CFGDirect

7 - Reject/CFGReject

8 - Announcement/CFGAnnouncement

9 - Post feature/CFGPostFeature

10 - Direct Agent/CFGDirectAgent

11 - Use external protocol/CFGUseExternalProtocol

12 - Get from DN/CFGGetFromDN

13 - Default/CFGXRouteTypeDefault

14 - Route/CFGXRouteTypeRoute

15 - Direct/CFGXRouteTypeDirect

16 - Reroute/CFGXRouteTypeReroute

17 - Direct UUI/CFGXRouteTypeDirectUUI

18 - Direct ANI/CFGXRouteTypeDirectANI

19 - Direct No Token/CFGXRouteTypeDirectNoToken

20 - DNIS pooling/CFGXRouteTypeDNISPooling

21 - Direct DNISn ANI/CFGXRouteTypeDirectDNISnANI

22 - Ext protocol/CFGXRouteTypeExtProtocol

23 - Direct digits/CFGXRouteTypeDirectDigits

24 - Forbidden/CFGXRouteTypeForbidden

25 - ISCC protocol/CFGXRouteTypeISCCProtocol

26 - Pullback/CFGXRouteTypePullBack

For more detailed information about these data types, see the *Genesys Framework 8.1 Deployment Guide*.

Starting the Adapter

After the Configuration Synchronization Component (the Adapter) has been deployed, it must always be kept running to ensure that the Genesys environment stays up-to-date with the Siebel environment. Running the Component constantly prevents error messages in the Siebel Web Client and guarantees that the Genesys environment reflects the latest configuration updates made in the Siebel environment.

The Configuration Synchronization Component can be started from the command line. Depending on the operating system that you use, the name of the component varies slightly, as follows:

Windows • GplusConfSynch.exe

UNIX • GplusConfSynch

The component supports the following command line options:

-host <host> -port <port> -app <application> [-clean_batch | -batch]

where:

<host> is the name of the host on which the Genesys Configuration Server is running.

<port> is the port of the Genesys Configuration Server.

<application> is the name of the Configuration Synchronization Component application.

The -batch option should be used to synchronize Siebel and Genesys agent data when the Configuration Synchronization Component is started for the first time. To remove all of the agent data from the Configuration Synchronization Component folders in Configuration Manager and then export agent data from Siebel, use the -clean_batch option.

If you are using Windows, you can also start the Configuration Synchronization Component from the Start menu by going to Programs > Genesys Solutions > Gplus Adapter for Siebel CRM and selecting the component's shortcut (the shortcut has the same name as the Configuration Synchronization Component Application object). This starts the component with the default command line options (without the -batch and -clean_batch options).

Notes: • Before starting the Configuration Synchronization Component with either the -batch or -clean_batch options, you should make sure that all of the latest changes that you made in the Siebel Web Client have been saved in the Siebel database. Usually, this can be done by switching to a view that is different from the one you that used to make the last modification.

• For the -batch and -clean_batch options to work correctly, different Siebel users should create Configuration Manager objects of the same type under the same Configuration Manager folder. When you use either the -batch or -clean_batch options, the Configuration Synchronization Component uses folder mapping for the Siebel user, specified by the username option in the Genesys section of the Configuration Synchronization Component application.

• You can use Genesys Management Layer, and specifically the Solution Control Interface, to start, or stop, or switch between primary and backup Configuration Synchronization Components. If you want to do this, make sure that the Command Line Arguments in the application Start Info tab do not include the -service argument.

5 Deploying the Campaign Synchronization Component

This chapter describes how to configure and install the Siebel CRM Campaign Synchronization Component of the G*plus* Adapter.

This chapter includes the following sections:

Overview

The 8.0.1 version of the G*plus* Adapter for Siebel CRM Campaign Synchronization Component is a new product in comparison to the earlier versions (7.5 and 8.0) of the Campaign Synchronization Component.

The major new features and enhancements are:

- Support for the Siebel waves campaign management feature, which means that instead of mapping (synchronizing) the Siebel calling lists to the Genesys calling lists, the Siebel campaign waves are now mapped to the Genesys calling lists.

- Support for the synchronization of multiple parallel campaigns, including single or multiple-tenants.

- Improved synchronization algorithms to achieve better performance.

- The communication between Siebel and the Gplus Adapter Campaign Synchronization Server is now unidirectional.

- The implementation of a Siebel GUI to simplify the configuration and synchronization procedures.

Each of these procedures is explained in detail under a separate heading as follows:

See the G*plus Adapter for Siebel CRM User's Guide* for further details about how to directly manage the campaign synchronization. Also, refer to the G*plus Adapter for Siebel CRM Developer's Guide* for details on customizable solutions.

New in This Release

This section provides information about new features or functionality in the Campaign Synchronization Component.

Release 8.0.210   No new features were introduced in this release.

Release 8.0.2   The following new features or functionality are included in this release:

- Support for Siebel versions 8.1.1.14/8.2.2.14 (IP2014) for both HI and Open UI modes.

- Support for Multiple Phone Numbers by Phone Type.

- Support for Genesys Framework 8.5.

- Support for Red Hat Enterprise Linux 6.

- Support for Red Hat Enterprise Linux 7.

- Support for AIX 7.1.

- Support for Solaris 11.

Release 8.0.110   The following new features or functionality are included in this release:

- Support for Siebel versions 8.1.1.11/8.2.2.4 (IP2013) for both HI and Open UI modes.

Release 8.0.1   The following new features or functionality are included in this release:

- The 8.0.1 version of the Gplus Adapter for Siebel CRM Campaign Synchronization Component is a new product in comparison to the earlier versions (7.5 and 8.0) of the Campaign Synchronization Component.

  The major new features and enhancements are:

  - Support for the Siebel waves campaign management feature.
  - Support for the synchronization of multiple parallel campaigns, including single or multiple-tenants.
  - Improved synchronization algorithms to achieve better performance.

◆   The communication between Siebel and the Gplus Adapter Campaign Synchronization Server is now unidirectional.

◆   The implementation of a Siebel GUI to simplify the configuration and synchronization procedures.

See the *Gplus Adapter for Siebel CRM User's Guide* for further details about how to manage campaign synchronization with the new approach. Also, refer to the *Gplus Adapter for Siebel CRM Developer's Guide* for details on customizable solutions.

Configuring Genesys

This chapter describes how to configure and install the 8.0.1 version of the Genesys part of the Campaign Synchronization Component.

Configuring the 8.0.1 version of the Genesys side of the Campaign Synchronization Component consists of the following sections:

Prestart Information

Before starting the configuration process you should have the following Genesys Framework applications running:

- Configuration Database
- Configuration Server
- Configuration Manager

As a Configuration Manager user, you should have sufficient privileges to make changes to Configuration Layer objects.

Importing the Campaign Synchronization Server Application Template

This section describes how to import the Campaign Synchronization Server Application Template.

Recommendations

Genesys recommends using an Application Template when you are configuring your Adapter. The Application Template for your Adapter contains the most important configuration options set to the values recommended for the majority of environments. When modifying configuration options for your Adapter later in the process, you can change the values inherited from the template rather than create all the options by yourself.

Procedure:
Campaign Synchronization Component: Importing the Campaign Synchronization Server Application Template

Purpose: To import the Campaign Synchronization Server Application Template.

Start of procedure

1. In Configuration Manager, under Environment, right-click the Application Templates folder.

2. Select Import Application Template.

3. Browse to select the Application Template for the Campaign Synchronization Server.

4. Select the proper template based on the version of Configuration Server used in your environment. Depending on the version of Configuration Manager that you use, this name varies slightly, as follows:
    - For versions 7.0 and earlier, use the following:
       Gplus_SiebelCRM_Camp_Synch_802_for_CL_70.apd
    - For versions 7.1 and later, use the following:
       Gplus_SiebelCRM_Camp_Synch_802_for_CL_71_and_higher.apd

5. Click Open.

    The Properties dialog box for the Application Template object displays.

6. Optional: Edit the Application Template name.

7. Click OK to accept the default values.

    The Application Template object has been imported to Genesys Configuration Layer.

End of procedure

Next Steps

- Create the Configuration Layer Application object for the Campaign Synchronization Server. See Procedure: Campaign Synchronization Component: Creating the Application object.

Creating the Campaign Synchronization Server Application Object

This section describes how to create the Campaign Synchronization Server Application object.

Procedure:
Campaign Synchronization Component: Creating the Application object

Purpose: To create the Application object for the Campaign Synchronization Server.

Start of procedure

1. In Configuration Manager, under Environment, right-click the Applications folder.

2. Select New > Application.

3. Select the Application template that you just created.

4. Click OK.

   The Properties dialog box for the Application displays.

End of procedure

Next Steps

- Configure the tabs in the Properties dialog box for the Campaign Synchronization Server. See, Procedure: Campaign Synchronization Component: Configuring the tabs in the Properties dialog box, on page 124.

Configuring the Tabs in the Properties Dialog Box

This section describes how to configure the tabs in the Properties dialog box, arranged in the order in which they display.

Procedure:
Campaign Synchronization Component: Configuring the tabs in the Properties dialog box

Purpose: To configure the tabs in the Properties dialog box for the Campaign Synchronization Server.

Start of procedure

General Tab   1. Click the General tab in the Properties dialog box (see Figure 11).
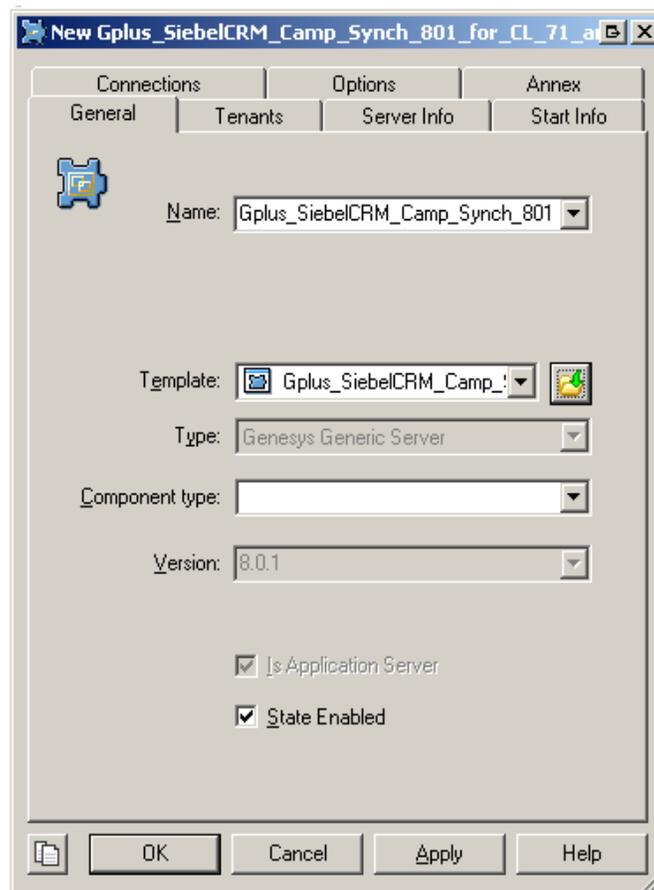
Figure 11: View of General Tab in a Multi-Tenant Environment

2. In the Name field, enter the name of the Application object you are configuring.

3. In the Template field, enter the name of the template that you are configuring or use the Browse button to select the template that you are configuring.

4. Click Apply.

5. Next, if you are working in a multi-tenant environment, go to the Tenants Tab; otherwise, go to the Server Info Tab.

Note: The Tenants tab only displays if you are working in a multi-tenant environment.

Tenants Tab

6. Select the Tenants tab.

7. Select the Genesys Tenants under which the objects that are exported from Siebel are created.

8. Click Add.

Server Info Tab

9. Select the Server Info tab.

10. In the Host field, use the Browse button to select the host where you are installing the Campaign Synchronization Component and click OK.

11. In the Ports list, add a port with any valid port number.

Note: This option determines the HTTP port number used by the Campaign Synchronization Server to receive HTTP packets from Siebel. You will use the value that you enter for this option when you configure the Siebel part of the Campaign Synchronization Server.

Start Info Tab

12. Select the Start Info tab.

13. In the Working Directory field, enter the full name of the Server installation directory on the host you specified on the Server Info tab. The value you enter in this field is used as the default destination folder during installation.

14. Enter any valid value into each of the following fields:
    - Command Line
    - Command Line Arguments

    The values you enter in these fields are overwritten during installation; however, the data must be present in these fields during the configuration process.

15. Leave the default values for the remaining fields.

Connections Tab

16. Select the Connections tab.

17. On the Connections tab, add the Applications from the following list to which this Application connects:

- Database Access Point—used to provide access to the Outbound Contact database for storing Calling List tables (mandatory).
- Outbound Contact Server—required for Do Not Call notifications and call results synchronization in real-time mode.
- T-Server—used by the Outbound Contact Server to communicate through a communication DN.
- Message Server—required, if you plan to use the Management Layer for alarm-signaling and centralized-logging capabilities.

Options Tab 18. Select the Options tab (see Figure 12 on page 126).

In the Sections pane, the following sections are listed:

- do-not-call
- Log
- outbound
- xml
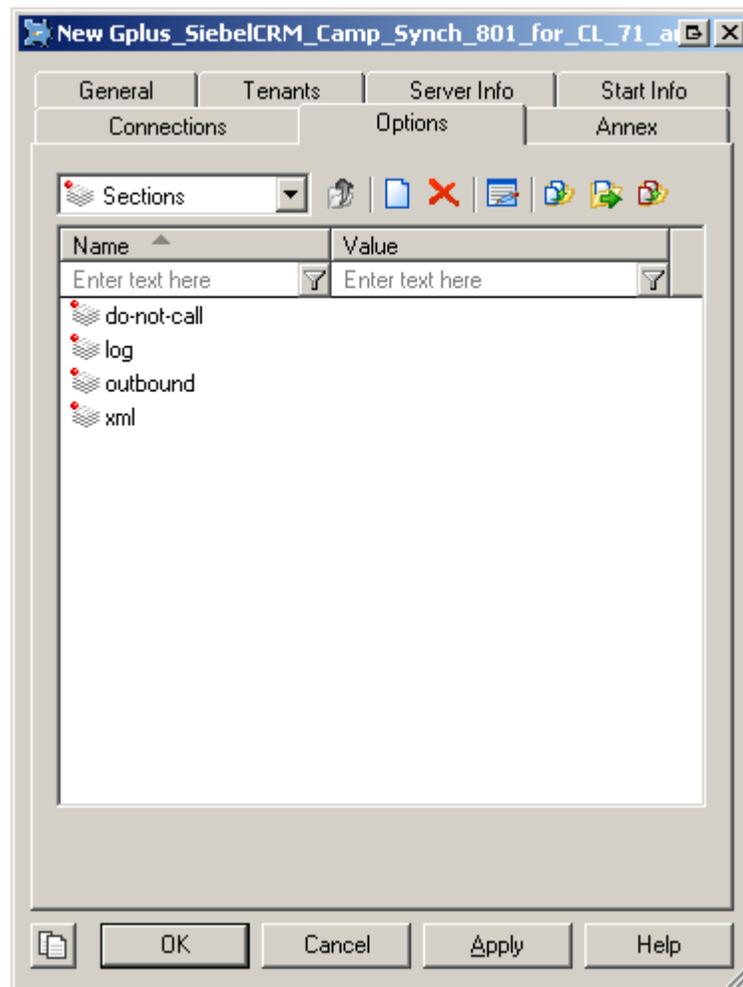
Figure 12: View of Options Tab in a Multi-Tenant Environment

19. Double-click a section to access its options.

20. Configure the configuration options that are described in the following section, "Setting the Genesys Configuration Options for the Campaign Synchronization Server" on page 127.

End of procedure

Next Steps

• Set the configuration options for the Campaign Synchronization Server. See the section, "Setting the Genesys Configuration Options for the Campaign Synchronization Server".

Setting the Genesys Configuration Options for the Campaign Synchronization Server

Unless specified otherwise, set the Campaign Synchronization Server configuration options in the Options tab of the Application object using the following navigation path:

• In Genesys Adminstrator—Application object > Options tab > Advanced View (Options).

• In Configuration Manager—Application object > Properties dialog box > Options tab.

For ease of reference, the options have been arranged in alphabetical order within their corresponding sections:

do-not-call Section

The do-not-call section has the following option that can be configured:

communication-dn

Default Value: <optional>
Valid Value: Any string value containing the correct Communication DN

Determines the name of the Communication DN that is used for interactions with the Outbound Contact Server, and specifically for the Do Not Call option.

Log Section

The Campaign Synchronization Server supports a common set of log options to allow the precise configuration of the log file output. For a complete list of

Common Log options and their descriptions, see the "Common Log Options" chapter of the *Framework 8.0 Configuration Options Reference Manual*.

Note: If you do not specify any log options, the default values apply.

---

outbound Section

The outbound section has various options that can be configured. The following section describes the possible options:

format

Default Value: GplusCampSynch
Valid Value: Any string value containing a Format object name

Specifies the name of a list import Format object in the Genesys Configuration Layer that is used by the Campaign Synchronization Server to create Genesys Calling List tables. The default value for this option is GplusCampSynch.

ocs-request-timeout

Default Value: 10
Valid Value: Any positive numeric value

Specifies the time interval (in seconds) that the Adapter waits for a reply from Outbound Contact Server for a Do Not Call request.

xml Section

The xml section has the following option that can be configured:

xsl-transformer (Optional)

Default Value: <optional>
Valid Value: A string value containing the *.xsl file name

This configuration option is optional. Transforms the values of various synchronized data (for example—phone number) using your own customized rules. The Adapter provides a mechanism to customize the transformation rules through an *.xsl file. The value of this option should be equal to the name of the *.xsl file.

Configuring the Security Settings

This section describes how to configure the security settings for the Campaign Synchronization Server. The security settings must be configured for the Campaign Synchronization Server, so that the Server can make changes to the Genesys Configuration Layer objects.

Follow the procedures outlined in the following sections to configure the security settings for the Application object that you created in the section,

"Creating the Campaign Synchronization Server Application Object" on page 123.

Procedure: Campaign Synchronization Component: Creating a new Person object

Purpose: To add a new Person object to the environment.

Start of procedure

1. In Configuration Manager:
   - Right-click the Persons folder under Environment, if you are working in a multi-tenant environment.
   - Right-click the Persons folder under Resources, if you are working in a single-tenant environment.

2. Select New > Person.

   The new Person window displays.

3. Select the General tab and enter the following parameters:
   - Employee ID = siebel7gplus
   - User Name = siebel7gplus
   - Is Agent = Clear

4. Click OK.

End of procedure

Next Steps

- Add a Person object to the Super Administrators Access Group. See, Procedure: Campaign Synchronization Component: Adding a Person object to the Super Administrators Access Group.

Procedure: Campaign Synchronization Component: Adding a Person object to the Super Administrators Access Group

Purpose: To add a Person object to the Super Administrators Access Group.

Start of procedure

1. In Configuration Manager, to display the existing Access Groups:
    - Right-click the Access Groups folder under Environment, if you are working in a multi-tenant environment.
    - Right-click the Access Groups folder under Resources, if you are working in a single-tenant environment.

2. Double-click the Super Administrators Access Group to display its properties.

3. Click the Add button to add the Person object that you just created to the Users list.

4. Click OK.

End of procedure

Next Steps

- Associate the newly created Person object account with the Application object. See, Procedure: Campaign Synchronization Component: Associating a Person object account with the Application object.

Procedure:
Campaign Synchronization Component: Associating a Person object account with the Application object

Purpose: To associate a Person object account with the Application object.

Start of procedure

1. In Configuration Manager, under Environment, click the Applications folder to display existing applications.

2. Double-click the Application object you just created to display its properties.

3. Select the Security tab, in the Log On As section, and select This Account. The Add User window displays.

4. Select the Person account you just created and click Add.

5. Click OK to close the Add User window.

6. Click OK to save the Application object.

End of procedure

Next Steps

- Create the list import Format object by following the procedures in the section, "Creating the List Import Format Object" on page 131.

Creating the List Import Format Object

This section describes how to create a List Import Format object. The List Import Format object determines the structure of the calling list tables that the Campaign Synchronization Component creates when importing Siebel campaigns. Creating the List Import Format object involves the following processes:

- Creating a Field object. See, Procedure: Campaign Synchronization Component: Creating Field objects, on page 132.
- Updating the Field object. See, Procedure: Campaign Synchronization Component: Updating the Field object, on page 135.
- Creating a Format object. See, Procedure: Campaign Synchronization Component: Creating the Format object, on page 136.
- Copying the Format object. See, Procedure: Campaign Synchronization Component: Copying the Format object to a Tenant, on page 138.

The first step of the process is to create several Field objects (which are mandatory) for each List Import Format object.

Displaying the Annex Tab
However, before you create the Field objects, ensure that the Annex tab is displayed in the Configuration Object properties window: See, Procedure: Campaign Synchronization Component: Configuring the Annex Tab to Display in Object Properties.

Procedure:
Campaign Synchronization Component: Configuring the Annex Tab to Display in Object Properties

Purpose: To ensure the Annex tab is displayed in the object properties.

Start of procedure

1. In Configuration Manager, from the View menu, select Options.
2. In the Object Properties dialog box, select the Show Annex tab.
3. To close the window, click OK.

End of procedure

Next Steps

- Create the Field objects. See, Procedure: Campaign Synchronization Component: Creating Field objects, on page 132.

Procedure:
Campaign Synchronization Component: Creating Field objects

Purpose: To create a new Field object in Configuration Manager.

Start of procedure

1. In Configuration Manager:
   - Right-click the Fields folder under Environment, if you are working in a multi-tenant environment.
   - Right-click the Fields folder under Resources, if you are working in a single-tenant environment.

   The new Field Properties window appears.

2. Select New > Field.

   The new Field window appears.

3. On the General tab for the Field you are creating, enter the Name field value from Table 9, "Field Object Properties," on page 134. For example, to complete the first instance, enter crm_campaign_id (see Figure 13).
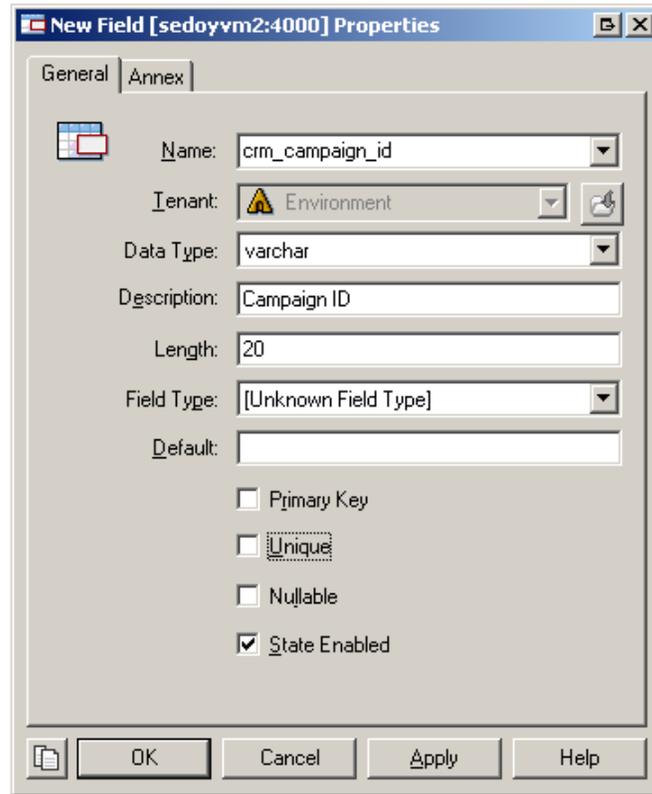
Figure 13: Creating a New Field Object

4.   On the ANNEX tab, create a new section called DEFAULT (see Figure 14).
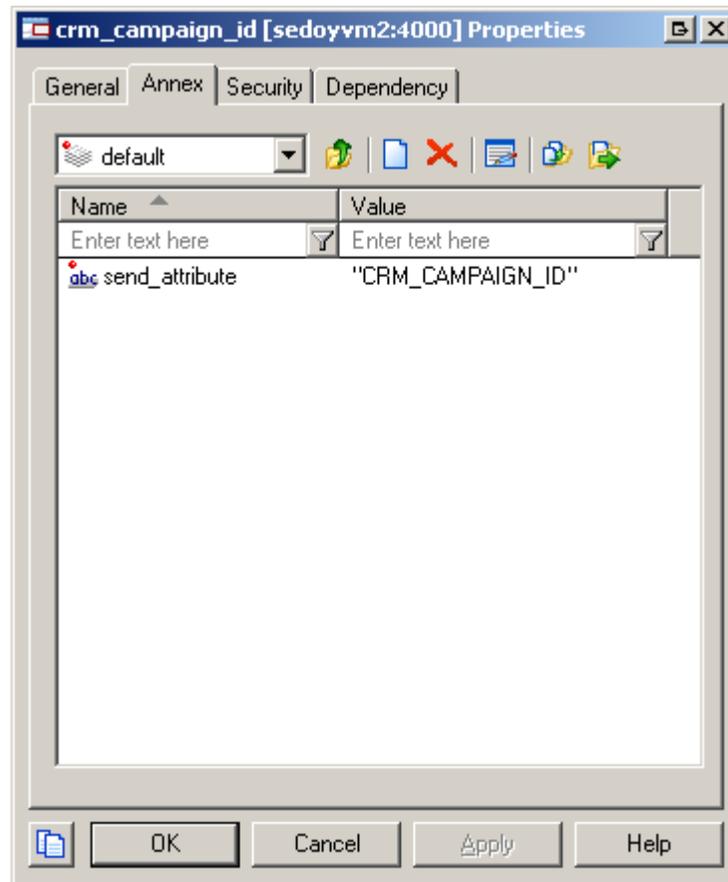
Figure 14: Creating the send_attribute Option

5. In the default section, create an option with the option name, send_attribute, and the option value from Table 10 on page 135, which corresponds to the Field object you are creating.

6. Click OK.

7. Repeat steps 1–6 for each Field shown in Table 9 on page 134.

Table 9 contains properties you need to enter on the General tab for each Field object.

Table 9: Field Object Properties

Field Name	Data Type	Description	Length	Field Type	Primary Key	Nullable
crm_campaign_id	varchar	Campaign ID	20	User-defined field	Cleared	Cleared

Table 9: Field Object Properties (Continued)

Field Name	Data Type	Description	Length	Field Type	Primary Key	Nullable
crm_camp_con_id	varchar	Campaign List Contact ID	20	User-defined field	Selected	Cleared
crm_contact_id	varchar	Contact ID	20	User-defined field	Cleared	Cleared
crm_phone_name	varchar	Phone Name	64	User-Defined	Cleared	Cleared

Table 10 shows the values for the send_attribute option, which you use when creating Field objects.

Table 10: Values for the send_attribute Option

Field Name	send_attribute Option Value
crm_campaign_id	CRM_CAMPAIGN_ID
crm_camp_con_id	CRM_CAMP_CON_ID
crm_contact_id	GSW_CUSTOMER_ID

End of procedure

Next Steps

• Update the Field objects. See, Procedure: Campaign Synchronization Component: Updating the Field object.

Procedure: Campaign Synchronization Component: Updating the Field object

Purpose: To update the Field object.

Start of procedure

1. If you need to synchronize the call results from Genesys to Siebel in real-time mode, update the following Field objects:
   • call_result
   • call_time
   • attempt
   • crm_campaign_id

- crm_camp_con_id
- crm_contact_id

Add the new option, icon_attribute to the default section of all of the Fields, and set the value of the option to 1 (see Figure 15).
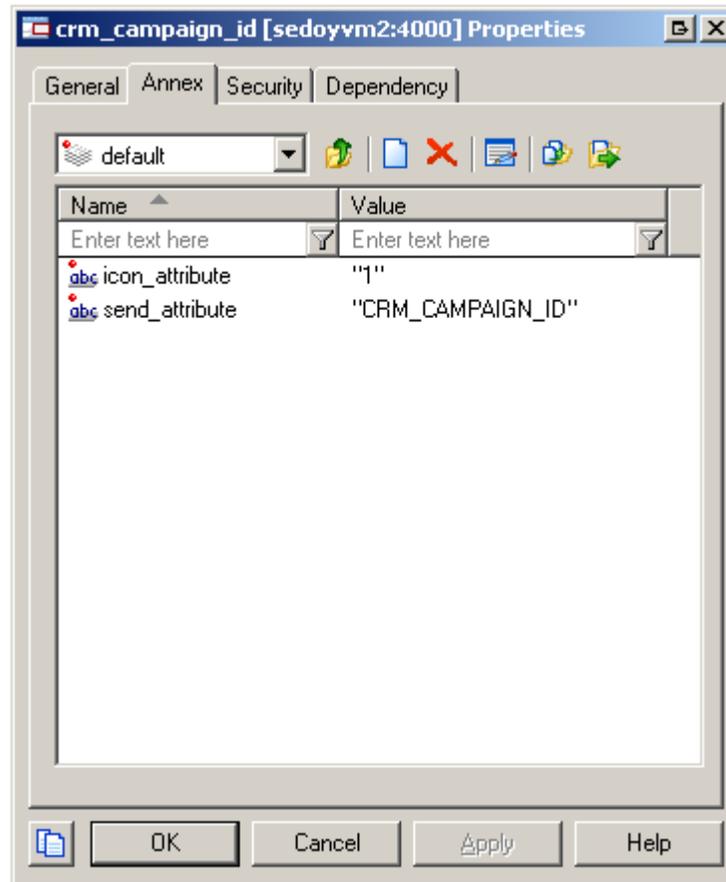
Figure 15: Adding the icon_attribute Option

End of procedure

Next Steps

- Create the Format object. See Procedure: Campaign Synchronization Component: Creating the Format object.

Procedure:
Campaign Synchronization Component: Creating the Format object

Purpose: To create the Format object.

Start of procedure

1. In Configuration Manager:
   - Right-click the Format folder under Environment, if you are working in a multi-tenant environment.
   - Right-click the Format folder under Resources, if you are working in a single-tenant environment.

   The new Format Properties window appears.

2. Select New > Copy of existing Format.

3. In the Browse window, select the default format for Outbound Contact, which is Default_Outbound_70.

4. Click OK.

   The Properties dialog box for the copy of the existing format that you are about to create displays.

5. Enter the following values on the General tab:
   - In the Name field, enter GplusCampSynch
   - In the Description field, enter the default List Import Format object for the G*plus* Adapter 8.1 for Siebel CRM Campaign Synchronization component.

6. Click OK.

7. In Configuration Manager, in the object tree view, right-click the Format object you just created.

8. Select New > Shortcut to Field.

   The Browse window displays.

9. Select the Field objects you just created.

10. Click OK.

End of procedure

Next Steps

- If you work in a multi-tenant environment, copy the Format object to a Tenant. See, Procedure: Campaign Synchronization Component: Copying the Format object to a Tenant.

- If you work in a single-tenant environment, synchronize the time zones. See the section, "Synchronizing the Time Zones" on page 138.

Procedure:
Campaign Synchronization Component: Copying the Format object to a Tenant

Purpose: To create a copy of the List Import Format object under each Tenant that is mapped to a Siebel Organization or Division.

Note: Complete this procedure *only* if you work in a multi-tenant environment

Start of procedure

1. In Configuration Manager, under a Tenant that is mapped to a Siebel Organization or Division, right-click the Formats folder.

2. Select New > Copy of existing Format.

   The Browse window displays.

3. Select the Format object you just created.

4. Click OK.

   The Properties dialog box appears for the Format copy that you are about to create.

5. Click OK to copy the Format.

End of procedure

Next Steps

• No further steps are required.

Synchronizing the Time Zones

This section describes how to synchronize the names of the time zones in the Genesys and Siebel environments. This means that you must make all of the Time Zone names used in your Genesys environment match those used in your Siebel environment. When you are finished, all of the time zones that are used in reference to the contacts or prospects in your Siebel environment must be present in the Genesys environment.

The procedure below explains how to copy the time zone names from the Siebel environment to Genesys Configuration Manager, and specifically to an individual Tenant that is mapped to a Siebel Organization or Division. For additional information about setting and using the time zone definitions in Configuration Manager, refer to the *Framework 8.1 Deployment Guide*.

Procedure:
Campaign Synchronization Component: Defining the time zones for a Tenant

Purpose: To define the time zones for a Tenant.

Note: It is also possible to modify the names of the time zones in the Siebel environment, so that they match the time zone names in the Genesys environment. The instructions above assume that the Siebel time zones have been defined first.

If the time zones have been imported into the Genesys environment while the Genesys Adapter is running (that is, after the deployment of this Adapter), then you must restart the Genesys Adapter to accept these changes.

For more information about Time Zone Administration in Siebel, see your Siebel documentation.

Start of procedure

1. Open both the Siebel and Configuration Manager applications.

2. In Configuration Manager, under a Tenant that is mapped to a Siebel Organization or Division, right-click the Time Zones folder.

3. Select New > Time Zone.

    The new Time Zone window displays.

4. In Siebel, navigate through the Site Map to get to Time Zone administration, as follows:

    •   Administration - Data > Time Zone Administration

5. In Configuration Manager, select the General tab and enter the values for the parameters listed below: (To minimize errors, use a cut-and-paste approach.)

    a. In the Name field, enter the exact name of the time zone from the Siebel environment.

    b. In the Description field, enter a brief description of the time zone.

    c. In the Offset field, enter the time difference between the local time and Greenwich Mean Time (GMT).

    d. In the Specific Zone Name in Netscape field, enter the time zone name used by the Netscape Navigator browser.

    e. In the Specific Zone Name in Microsoft Explorer field, enter the time zone name used by Microsoft Explorer.

6. Click OK.

End of procedure

Next Steps

- No further steps are required.

Installation

This section describes the installation process for the G*plus* Adapter 8.0.1 for the Siebel CRM Campaign Synchronization Component. Select one of the following sections, depending on your environment:

- Installing the Campaign Synchronization Component in a Windows Environment, page 140
- Installing the Campaign Synchronization Component in a UNIX Environment, page 141

Installing the Campaign Synchronization Component in a Windows Environment

This section describes how to install the Campaign Synchronization Component on a Windows system.

Procedure:
Campaign Synchronization Component: Installing the Component in a Windows environment

Purpose: To install the Campaign Synchronization Component on a Windows operating system.

Start of procedure

1. To start the installation process, run the setup.exe file from the Campaign Synchronization Component installation package.

2. In the Welcome window, click Next.

3. In the Configuration Parameters to the Genesys Configuration Server window, enter the following:

   a. The name of the host on which Configuration Server is running.

   b. The communication port that client applications must use to connect to Configuration Server.

   c. The user name used to access Configuration Server.

    d. The password used to access Configuration Server.

4. Click Next.

5. In the Select Application window, select the Application object you configured in the previous procedure, Procedure: Campaign Synchronization Component: Creating the Application object, on page 123.

6. Click Next.

7. In the Choose Destination Location window, click Next to accept the default destination folder, or click Browse to select a different destination folder.

8. In the Ready to Install window, click Install.

9. In the Setup Complete window, click Finish.

    The Campaign Synchronization Component is now installed.

    In the G*plus* Adapter for Siebel CRM program folder in the Start menu, you can see that the installer created a shortcut for the Campaign Synchronization Component.

End of procedure

Next Steps

• Configure the Siebel part of the Campaign Synchronization Component. See, "Configuring Siebel" on page 142.

Installing the Campaign Synchronization Component in a UNIX Environment

This section describes how to install the Campaign Synchronization Component on a UNIX operating system.

Note: Do not use special symbols in any destination directory name when installing in a UNIX environment.

Procedure:
Campaign Synchronization Component: Installing the Component in a UNIX environment

Purpose: To install the Campaign Synchronization Component on a UNIX system.

Note: Do not use special symbols in any destination directory name when installing in a UNIX environment.

Start of procedure

1. In the directory where the Component installation package was copied, locate a shell script named: install.sh.

2. Run this script from the command prompt by typing sh and the file name:

   sh install.sh

3. When prompted, specify the host name, port, user name, and password of the computer on which the Component will be installed—for example:

   a. Enter the host name of Configuration Server.

   b. Enter the port of Configuration Server.

   c. Enter the user name for Configuration Server.

   d. Enter the password for the user name.

4. Choose the Configuration Server environment by its corresponding number.

5. Specify the Application object that you configured in the previous procedure, Procedure: Campaign Synchronization Component: Creating the Application object, on page 123.

6. Specify the full path to the destination directory where you want the Component to be installed.

   The Campaign Synchronization Component is now installed.

   In the G*plus* Adapter for Siebel CRM program folder in the Start menu, you can see that the installer created a shortcut for the Campaign Synchronization Component.

End of procedure

Next Steps

- Configure the Siebel part of the Campaign Synchronization Component. See, "Configuring Siebel" on page 142.

Uninstalling the Campaign Synchronization Component

Complete information about uninstalling the G*plus* Adapter and its components is provided in Chapter 14, "Uninstallation Instructions," on page 565.

Configuring Siebel

This section describes how to configure the Siebel part of the Campaign Synchronization Component.

Configuring the Siebel section of the Campaign Synchronization Component consists of the following sub-sections:

Prestart Information

Before starting this part of the configuration process, you must make sure to do the following:

- You must have the Siebel Tools application running.

- You should be connected to the local copy of the Siebel Server database.

- You should have sufficient privileges to check-in and check-out projects from the Siebel Server repository.

- You should ensure that all required steps from "Patching and Configuring Siebel CRM" are completed.

Note: Campaign Synchronization functionality is intended for use with the Outbound Campaign feature, which requires Siebel Marketing. Siebel Marketing is required to manage outbound campaign interactions. As a logical consequence, Campaign Synchronization effectively requires Siebel Marketing.

Configuring Siebel Using the Siebel Tools

Use the Siebel Tools to compile an updated version of the Siebel repository file (SRF or *.srf file) for one, or more, of the Siebel applications that you use on your Siebel Server, which you will then deploy on the server. For more information about using Siebel Tools, see the Siebel documentation.

Configuring Siebel using the Siebel Tools consists of the following sub-sections:

Overview of the Siebel Repository File Preparation

When compiled, the Siebel repository file integrates information from the various Campaign Synchronization Component archives files (the *.sif files) that you import.

For the Campaign Synchronization Component, you must import all of the Campaign Synchronization Component *.sif files and resolve any data conflicts introduced by the imported file.

Checking Out Existing Projects from the Siebel Repository for the Campaign Synchronization Component

The Siebel implementation of the Campaign Synchronization Component makes use of a number of objects provided by Siebel. Use the procedure to enable the modifications of these objects. To do so, you will need to check-out the corresponding projects from the Siebel repository.

Procedure:
Campaign Synchronization Component: Checking out existing projects

Purpose: To check-out existing projects from the Siebel repository.

Start of procedure

1. In Object Explorer, start Siebel Tools against the local database.
2. In Siebel Tools, select Tools > Check Out...

   The Check Out dialog box displays.
3. In the Projects list, select the Campaign project to be checked-out.

> **Note:** When you first open the Projects list, the Genesys Symbolic Strings and Genesys Campaign Synchronization2.sif projects may not yet exist, so they may not be available to be checked-out and locked.
>
> - The project named Genesys Symbolic Strings is created when you import the GenSymbolicStrings.sif file, as explained in Procedure: Campaign Synchronization Component: Importing the GenSymbolicStrings.sif archive file, on page 146. Lock the project after importing the GenSymbolicStrings.sif file.
> - The project named Campaign is modified when you import the Campaign.sif file, as explained in Procedure: Campaign Synchronization Component: Importing the Campaign.sif Archive File, on page 147 below. Check-out and lock the project before importing the Campaign.sif file.
> - The project named Genesys Campaign Synchronization2 is created when you import the GenesysCampaignSynchronization.sif file, as explained in Procedure: Campaign Synchronization Component: Importing the GenesysCampaignSynchronization.sif archive file, on page 148. Lock the project after importing the GenesysCampaignSynchronization.sif file.

End of procedure

Next Steps

- Import the GenesysCampaignSynchronization.sif archive files. See the section, "Importing the GenSymbolicStrings.sif Archive File" on page 145.

Importing the GenSymbolicStrings.sif Archive File

The GenSymbolicStrings.sif archive contains the localized symbolic strings definition used for all of the G*plus* Adapter for Siebel CRM Components. The objects are added to the Genesys Symbolic Strings project.

Procedure:
Campaign Synchronization Component: Importing the GenSymbolicStrings.sif archive file

Purpose: To import the mandatory archive file with symbolic strings, GenSymbolicStrings.sif, that is used across the Siebel archive files provider within the Adapter.

Notes: • The GenSymbolicStrings.sif file must be imported first.
• If the Genesys Symbolic Strings project already exists in your Siebel repository file, lock it or check it out before importing the GenSymbolicStrings.sif archive. Otherwise, lock the project after the import.

Start of procedure

1. In Siebel Tools, select Tools > Import from Archive...

2. In the Select Archive to Import window, navigate to the GenSymbolicStrings.sif archive file.

   The Campaign Synchronization Component installation program created this file in the following directory: <target directory>/ objects/<LANG> where <LANG> is the subfolder name (ENU (English) by default) depending on the Siebel locale you use.

3. Click Open.

   The Import Wizard–Preview window displays.

4. In the Import Wizard–Preview window, in the Conflict resolution section, select Merge the object definition from the archive file with the definition in the repository.

5. Click Next.

   The Import Wizard–Review Conflicts and Actions window displays.

6. Click Next.

   The "Do you wish to proceed?" window displays.

7. Merge any conflicts, if they correspond to your needs.

8. Click Yes.

   The objects from the archive are imported into the Siebel repository.

9. Click Finish to complete the import.

End of procedure

Next Steps

- Import the Campaign.sif archive. See, Procedure: Campaign Synchronization Component: Importing the Campaign.sif Archive File.

Importing the Campaign.sif Archive File

The Campaign.sif archive file contains the modifications for the Siebel Campaign List Contact business component.

Procedure: Campaign Synchronization Component: Importing the Campaign.sif Archive File

Purpose: To import the Campaign.sif archive file.

Start of procedure

1. In Siebel Tools, select Tools > Import from Archive...

2. In the Select Archive to Import window, navigate to the Campaign.sif archive file.

    This file is located at: <target directory>/ objects/<Siebel version>, where <Siebel version> is the subfolder name (/7.7, /8.0, /8.1, /8.1_8.2_OUI or IP2014, depending on the Siebel version being used).

3. Click Open.

    The Import Wizard–Preview window displays.

4. In the Import Wizard–Preview window, in the Conflict resolution section, select Merge the object definition from the archive file with the definition in the repository.

5. Click Next.

    The Import Wizard–Review Conflicts and Actions window displays.

6. In the Conflicting Objects pane, select the Business Component category.

7. In the Object differences pane, select Campaign List Contact. If there are any attributes in the Attribute differences pane, make sure that the Resolution column for each attribute is set to Repository. To do this: right-click on an attribute and select Repository from the drop-down menu.

8. Click Next.

    The "Do you wish to proceed?" window displays.

9. Click Yes.

    The objects from the archive are imported into the Siebel repository.

10. Click Finish to complete the import.

> **Note:** It is possible for siebel_assert_xxx.txt file(s) to be generated by the Siebel environment after importing the archive. This does not affect the functionality of the Component in any way.

End of procedure

Next Steps

*   Import the GenesysCampaignSynchronization.sif archive file. See, Procedure: Campaign Synchronization Component: Importing the GenesysCampaignSynchronization.sif archive file, on page 148.

Importing the GenesysCampaignSynchronization.sif Archive File

During this step, you will import objects into the Siebel repository that are part of the Siebel implementation of the Campaign Synchronization Component. The objects are added to the Genesys Campaign Synchronization2 project in Siebel Tools.

> **Notes:** If the Genesys Campaign Synchronization2 project already exists in your Siebel repository file archive, lock it before importing the GenesysCampaignSynchronization.sif archive file.
>
> If this project does not exist before export, lock it after the export.

Procedure:
Campaign Synchronization Component: Importing the GenesysCampaignSynchronization.sif archive file

Purpose: To import the GenesysCampaignSynchronization.sif archive file for the Campaign Synchronization Component.

Start of procedure

1. In Siebel Tools, select Tools > Import from Archive...

2. In the Select Archive to Import window, navigate to the GenesysCampaignSynchronization.sif archive file.

This file is located at: target directory>/ objects/<Siebel version>, where <Siebel version> is the subfolder name (/7.7, /8.0, /8.1, /8.1_8.2_OUI or IP2014, depending on the Siebel version being used).

3. Click Open.

The Import Wizard–Preview window displays.

4. In the Import Wizard–Preview window, in the Conflict resolution section, select Overwrite the object definition in the repository.

5. Click Next.

The Import Wizard–Review Conflicts and Actions window displays.

6. Click Next.

The "Do you wish to proceed?" window displays.

7. Click Yes.

The objects from the archive are imported into the Siebel repository.

8. Click Finish to complete the import.

Note: It is possible for siebel_assert_xxx.txt files to be generated by the Siebel environment after importing the archive. This does not affect the functionality of the component in any way.

End of procedure

Next Steps

• Select the correct path to the Siebel File System directory. See, Procedure: Campaign Synchronization Component: Selecting the correct path to the Siebel File System directory.

Selecting the Correct Path to the Siebel File System Directory

The proper path to your Siebel File System directory in the FileSystem user property of the Genesys CampSynch Campaign Business Service must be provided. This directory is used by the Campaign Synchronization Component to store temporary files (these files contain xml-based requests for synchronization).

For more information about the Siebel File System directory, see the Siebel System Administration Guide.

Procedure:
Campaign Synchronization Component: Selecting the correct path to the Siebel File System directory

Purpose: To provide the proper path to your Siebel File System directory in the user property named FileSystem of the Genesys Campaign Synchronization Business Service.

Start of procedure

1. In Siebel Tools, in the Project field of the Object Explorer, select Genesys Campaign Synchronization2.

2. Select the Business Service folder.

   The Business Services window displays.

3. Select the Genesys CampSynch Campaign Business Service.

4. Using the Types tab of the Object Explorer, select the Business Service User Prop folder under the Business Service folder.

   The Business Service User Props window displays.

5. Select the FileSystem Business Service user property.

6. In the Value column, enter the correct path to the Siebel File System directory as it is defined in your Siebel environment—for example:
   - For Unix: /siebel/filesystem/
   - For Windows: C:\sea\filesystem\

   If no value is defined, the siebsrvr\bin directory is used for temporary files.

End of procedure

Next Steps

- Select the page size in XML-messages. See, Procedure: Campaign Synchronization Component: Selecting the page size in XML-messages.

Selecting the Page Size in XML-Messages

The user property, ContactPageSize, allows for the customization of the XML-message size for large campaign lists. The Adapter splits the wave records into fixed-sized chunks depending on the default value set for XML-messages that are greater or equal then (<=) the value set for the Page Size user property. The default value is 100 records. Follow the steps in Procedure: Campaign Synchronization Component: Selecting the page size in XML-messages to update the Siebel Server parameter, ContactPageSize.

Procedure:
Campaign Synchronization Component: Selecting the page size in XML-messages

Purpose: To set the page size in XML-messages.

Start of procedure

1. In Siebel Tools, in the Project field of the Object Explorer, select Genesys Campaign Synchronization2.

2. Select the Business Service folder.

   The Business Services window displays.

3. Select the Genesys CampSynch Campaign Business Service.

4. Using the Types tab of the Object Explorer, select the Business Service User Prop folder under the Business Service folder.

   The Business Service User Props window displays.

5. Select the ContactPageSize Business Service user property.

6. In the Value column, enter the required number. The default value is 100 records.

Note: When changing the ContactPageSize user property to a larger value, keep in mind that the HTTPSleepTime Siebel parameter for the Genesys HTTP connection, with a default value of 120000 milliseconds (ms), should be enough for processing any requests on the Genesys side. Practically speaking, if a message contains more then 500 records, it could take a longer time for processing, and in this scenario, the HTTPSleepTime parameter size should be increased as well. See the *Gplus Adapter for Siebel CRM User's Guide* for more details on how to setup the HTTP connections for the Campaign Synchronization Component.

End of procedure

Next Steps

- Select the PhoneField and PhoneType Business Service user properties to export during the wave synchronization. See, Procedure: Campaign Synchronization Component: Selecting the phones and their type to export during the wave synchronization.

Selecting the Phones and Their Types to Export During the Wave Synchronization

You must provide the contact phone list that is to be exported during the wave synchronization. For each phone used in the wave synchronization, the Genesys PhoneField and PhoneType Business Service user properties must be added to the Genesys CampSynch Campaign Business Service user property. These PhoneField Business Service user property names must be taken from the Campaign List Contact business component.

Note: By default, the Calculated Home Phone # and the Calculated Work Phone # fields are listed with the corresponding Type values of 1 and 2.

Procedure:
Campaign Synchronization Component: Selecting the phones and their type to export during the wave synchronization

Purpose: To provide the contact phone list that is to be exported during the wave synchronization.

Start of procedure

1. In Siebel Tools, in the Project field of the Object Explorer, select Genesys Campaign Synchronization2.

2. Select the Business Service folder.

3. The Business Services window displays.

4. Select the Genesys CampSynch Campaign Business Service.

5. Using the Types tab of the Object Explorer, select the Business Service User Prop folder under the Business Service folder.

   The Business Service User Props window displays.

6. Select or create the PhoneField<N> Business Service user property, where <N> is a sequence number.

7. In the Value column, enter the required PhoneField name.

8. Select or create the PhoneType<N> Business Service user property, where <N> is a sequence number.

9. In the Value column, enter the required number that corresponds to the Genesys PhoneType Business Service user property. See the Genesys Outbound Contact Solution documentation for more details about the Genesys PhoneType Business Service user properties.

10. Repeat steps 6-9 for each phone that has to be synchronized.

11. If you use runtime events to track the updated contacts and use the non-default PhoneField Business Service user properties, make the required changes in the Genesys CampSynch Tools and Genesys CampSynch Event Handler business services, according to the steps described in the following section. See, Procedure: Campaign Synchronization Component: Customizing the runtime events, on page 154.

End of procedure

Next Steps

- Select CustomField Business Service user properties to export during the wave synchronization. See, Procedure: Campaign Synchronization Component: Selecting the custom fields to export during the wave synchronization, on page 153.

Selecting the Custom Fields to Export During the Wave Synchronization

You must provide the Campaign List Contact business component fields that are to be exported to the Genesys side as part of the contact record. By default, no custom fields are exported to the Genesys side.

Procedure: Campaign Synchronization Component: Selecting the custom fields to export during the wave synchronization

Purpose: To select the custom fields to export during the wave synchronization.

Start of procedure

1. In Siebel Tools, in the Project field of the Object Explorer, select Genesys Campaign Synchronization2.

2. Select the Business Service folder.

3. The Business Services window displays.

4. Select the Genesys CampSynch Campaign Business Service.

5. Using the Types tab of the Object Explorer, select the Business Service User Prop folder under the Business Service folder.

   The Business Service User Props window displays.

6. Select or create the CustomField<N> Business Service user property, where <N> is a sequence number.

7. In the Value column, enter the required CustomField name.

8. Repeat steps 6-7 for each CustomField that has to be synchronized.

9. If you use runtime events to track the updated contacts and use the non-default CustomField Business Service user property, make the required changes in the Genesys CampSynch Tools and Genesys CampSynch Event Handler Business Services, according to the steps described in the following section. See, Procedure: Campaign Synchronization Component: Customizing the runtime events, on page 154.

End of procedure

Next Steps

• Customize the runtime events. See, Procedure: Campaign Synchronization Component: Customizing the runtime events, on page 154.

Customizing the Runtime Events

You must customize the runtime events, if you are doing the following:

• Using the Siebel runtime events to dynamically update the Genesys-side contact data when it is modified on the Siebel-side.

• Using PhoneField and CustomField Business Service user properties that are not the same as the Business Service user properties that are found in the out-of-the-box solution.

Follow the steps provided in Procedure: Campaign Synchronization Component: Customizing the runtime events to customize these runtime events.

Procedure:
Campaign Synchronization Component: Customizing the runtime events

Purpose: To automatically install or uninstall runtime events and generate requests when the Genesys-side contact data is modified on the Siebel-side.

Start of procedure

1. For each PhoneField and CustomField Business Service user property that you added in Procedure: Campaign Synchronization Component: Selecting the custom fields to export during the wave synchronization, on page 153 and

on , you must determine which fields of which business component object contains the contact data.

These fields are from the Campaign List Contact business component object that contains references from the Contact, Employee, List Mgmt Prospective Contact, and other business component objects. For example—the Calculated Home Phone # field value is actually stored in various locations (as shown in bulleted form below) under the following syntax—
<Business Component>:<Field Name>:

- Contact: Home Phone #
- Consumer: Home Phone #
- Employee: Home Phone #
- Person: Home Phone #
- User: Home Phone #
- List Mgmt Prospective Contact: Home Phone #

2. Add the following code lines into the SetupAllRunTimeEvents method of the Genesys CampSynch Tools Business Service:

- xSetupRunTimeEventWithPA(<BusCompName>, "PreSetFieldValue", <FieldName>, 1, <PrevVarName>, <FieldName>, sMethodName, bDebugLog);
- xSetupRunTimeEventWithPA(<BusCompName>, "SetFieldValue", <FieldName>, 1, <VarName>, <FieldName>, sMethodName, bDebugLog);

where:

- <BusCompName>: is the name of the required business component
- <FieldName>: is the name of the required field
- <PrevVarName> and <VarName>: are the names of the global variables where both the previous and the new field values are stored when they are modified.

3. Add the following code lines into the Contact_WriteRecord method of the Genesys CampSynch Event Handler Business Service:

- var <local_prev_variable_name> = TheApplication().GetProfileAttr(<PrevVarName>);
- var <local_variable_name> = TheApplication().GetProfileAttr(<VarName>);

4. Modify the condition (highlighted below as **Bold**) used to determine if the contact record was changed:

if ((sPrevDNC != sDNC) ||
    (sPrevHome != sHome) ||
    (sPrevWork != sWork) ||
    (<local_prev_variable_name> != <local_variable_name>) ||
    (sPrevTZ   != sTZ))

5. Add the following lines of code to clear the global variables:

- TheApplication().SetProfileAttr(<PrevVarName>,"");

- TheApplication().SetProfileAttr(<VarName>,"");

End of procedure

Next Steps

- Look at an example of how to customize a runtime event for the Campaign List Contact business component field named, Calculated Fax Phone #. See, Procedure: Campaign Synchronization Component: Example of a runtime event customization, on .

Procedure:
Campaign Synchronization Component: Example of a runtime event customization

Purpose: To illustrate an example of a runtime event customization for the Campaign List Contact business component Calculated Fax Phone # field.

Start of procedure

1. In Siebel Tools, in the Project field of the Object Explorer, select All Projects.
2. Select the Business Component folder.

   The Business Component window displays.
3. Select the Campaign List Contact business component.
4. Select the Calculated Fax Phone # field (see Figure 16 on).

   The Calculated Fax Phone # field is calculated from the following fields:
   - Fax Phone #
   - Prospect Fax Phone #
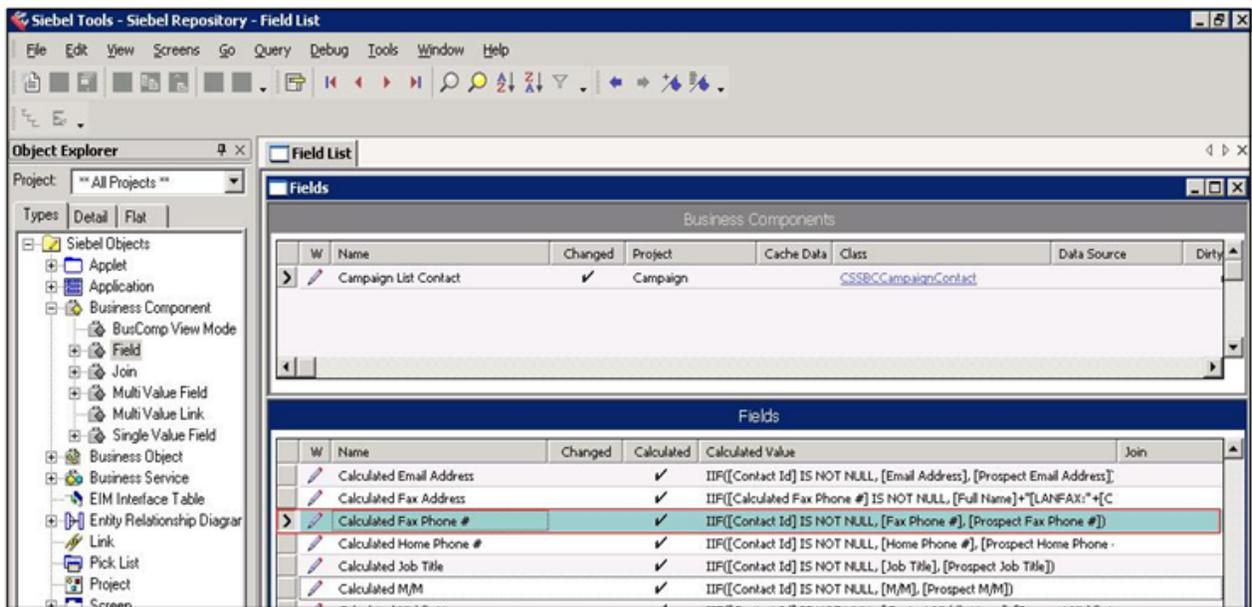
Figure 16: Selecting the Calculated Fax Phone #Field

> **5.** Select the Fax Phone # and Prospect Fax Phone # fields (see Figure 17 and Figure 18).
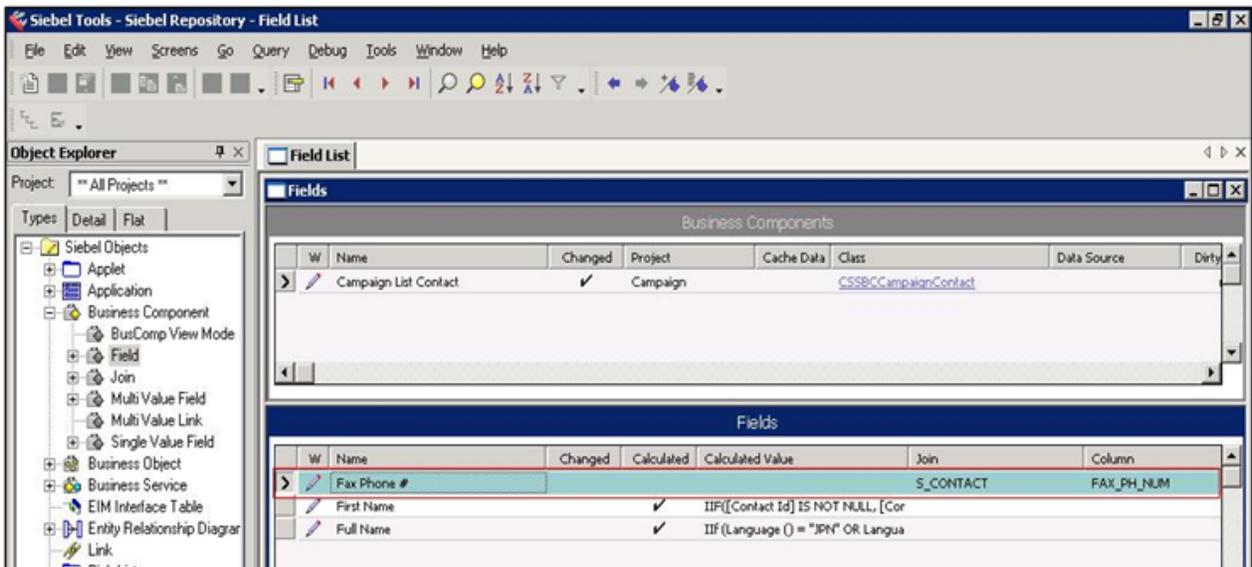
Figure 17: Selecting the Fax Phone # Field
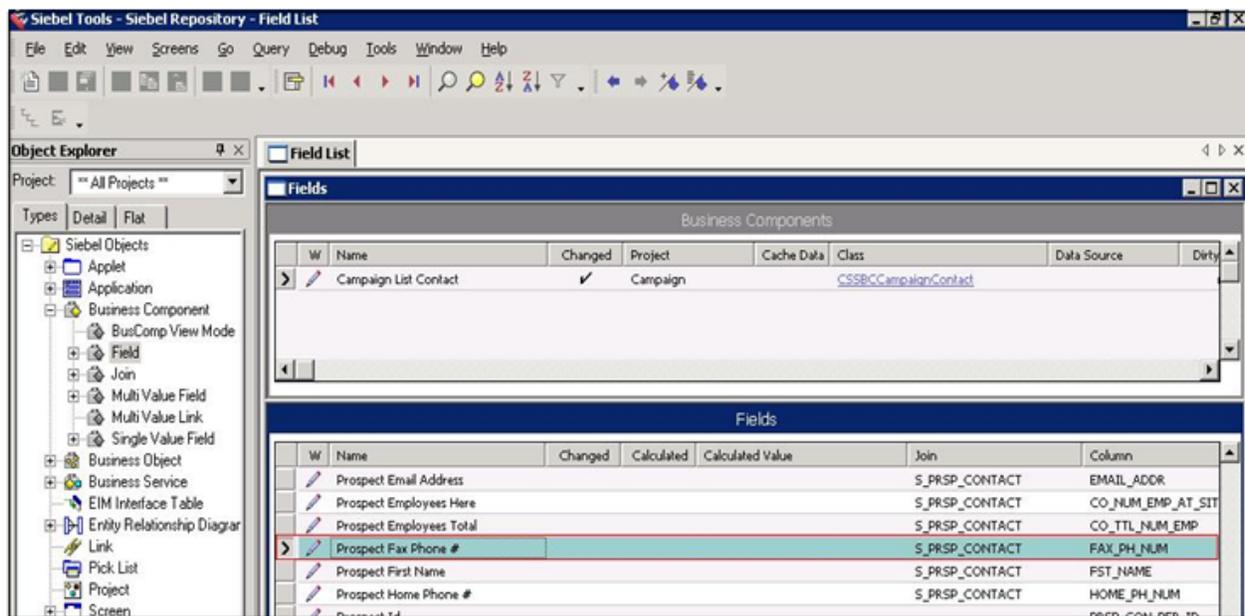
Figure 18: Selecting the Prospect Fax Photo # Field

Both Figure 17 and Figure 18 display the Join column. The Join column displays the foreign key relationship between the joined table and the business component's base table, which means that for every record in the business component that corresponds to a row in the base table there can be a corresponding row in the joined table. The Fax Phone # and Prospect Fax Phone # fields are accessible through this Join column—for example, a join uses the foreign key relationship to make the Fax Phone # data in the Name column available to the S_CONTACT table and to also make the Prospect Fax Phone # data in the Name column and the FAX_PH_NUM data in the Column column available to the S_PRSP_CONTACT table.

6. For the Fax Phone # field name, search through the Contact, Consumer, Employee, Person, and User business components and locate the field names that use the FAX_PH_NUM table column, which is found under the Column column.

For example in Figure 19 on page 159, the corresponding field name for the Contact business component is Fax Phone #. The corresponding field names for the other business component are listed as follows:

- Consumer: Fax Phone #
- Employee: Fax #
- Person: Fax Phone #
- User: Fax #

Note: The field names may vary from one business component to another.
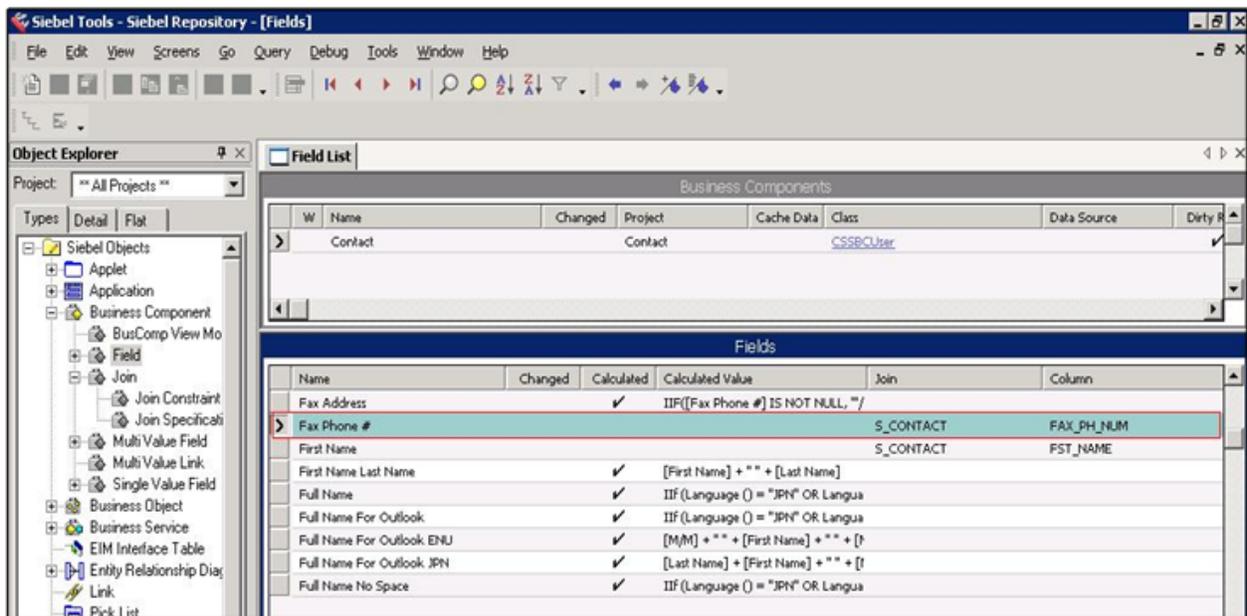
Figure 19: Selecting the FAX_PH_NUM Table Column

7. For the List Mgmt Fax Phone # field name, search through the List Mgmt
Prospective Contact business component and locate the Fax Phone # field
name that uses the FAX_PH_NUM table column, which is found under the
Column column.

8. Edit the code for the SetupAllRunTimeEvents method of the Genesys
CampSynch Tools Business Service. The new lines of code are highlighted
below as **Bold**:

```
function SetupAllRunTimeEvents(sMethodName, bDebugLog)
{
……
    xSetupRunTimeEventWithPA("Contact", "PreSetFieldValue", "Work Phone #",
1, "paContact_PrevWork", "Work Phone #",       sMethodName, bDebugLog);
    xSetupRunTimeEventWithPA("Contact", "SetFieldValue",    "Work Phone #",
1, "paContact_Work",    "Work Phone #",       sMethodName, bDebugLog);
```
   xSetupRunTimeEventWithPA("Contact", "PreSetFieldValue", "Fax Phone
#",   1, "paContact_PrevFax", "Fax Phone #",   sMethodName,
bDebugLog);
   xSetupRunTimeEventWithPA("Contact", "SetFieldValue",   "Fax Phone #",
1, "paContact_Fax",   "Fax Phone #",   sMethodName, bDebugLog);
```
    xSetupRunTimeEventWithPA("Contact", "PreSetFieldValue", "Time Zone Id",
1, "paContact_PrevTZ",  "Time Zone Id",       sMethodName, bDebugLog);
    xSetupRunTimeEventWithPA("Contact", "SetFieldValue",   "Time Zone Id",     1,
"paContact_TZ",      "Time Zone Id",       sMethodName, bDebugLog);
…….
```

```
   xSetupRunTimeEventWithPA("Consumer", "PreSetFieldValue", "Work Phone #",
1, "paContact_PrevWork", "Work Phone #",      sMethodName, bDebugLog);
   xSetupRunTimeEventWithPA("Consumer", "SetFieldValue",   "Work Phone #",
1, "paContact_Work",    "Work Phone #",      sMethodName, bDebugLog);
   xSetupRunTimeEventWithPA("Consumer", "PreSetFieldValue", "Fax Phone
#",     1, "paContact_PrevFax", "Fax Phone #",       sMethodName,
bDebugLog);
   xSetupRunTimeEventWithPA("Consumer", "SetFieldValue",   "Fax Phone
#",     1, "paContact_Fax",    "Fax Phone #",      sMethodName, bDebugLog);
……
   xSetupRunTimeEventWithPA("Employee", "PreSetFieldValue", "Work Phone
Number", 1, "paContact_PrevWork", "Work Phone Number",  sMethodName,
bDebugLog);
   xSetupRunTimeEventWithPA("Employee", "SetFieldValue",   "Work Phone
Number", 1, "paContact_Work",    "Work Phone Number",  sMethodName,
bDebugLog);
   xSetupRunTimeEventWithPA("Employee", "PreSetFieldValue", "Fax #",  1,
"paContact_PrevFax", "Fax #",  sMethodName, bDebugLog);
   xSetupRunTimeEventWithPA("Employee", "SetFieldValue",   "Fax #",  1,
"paContact_Fax",    "Fax #",  sMethodName, bDebugLog);
   xSetupRunTimeEventWithPA("Employee", "PreSetFieldValue", "Time Zone Id",
1, "paContact_PrevTZ",  "Time Zone Id",      sMethodName, bDebugLog);
   xSetupRunTimeEventWithPA("Employee", "SetFieldValue",   "Time Zone Id",
1, "paContact_TZ",    "Time Zone Id",      sMethodName, bDebugLog);
…..
   xSetupRunTimeEventWithPA("Person", "PreSetFieldValue", "Work Phone #",
1, "paContact_PrevWork", "Work Phone #",      sMethodName, bDebugLog);
   xSetupRunTimeEventWithPA("Person", "SetFieldValue",   "Work Phone #",    1,
"paContact_Work",    "Work Phone #",      sMethodName, bDebugLog);
   xSetupRunTimeEventWithPA("Person", "PreSetFieldValue", "Fax Phone #",
1, "paContact_PrevFax", "Fax Phone #",       sMethodName, bDebugLog);
   xSetupRunTimeEventWithPA("Person", "SetFieldValue",   "Fax Phone #",
1, "paContact_Fax",    "Fax Phone #",       sMethodName, bDebugLog);
   xSetupRunTimeEventWithPA("Person", "PreSetFieldValue", "Time Zone Id",
1, "paContact_PrevTZ",  "Time Zone Id",      sMethodName, bDebugLog);
   xSetupRunTimeEventWithPA("Person", "SetFieldValue",   "Time Zone Id",    1,
"paContact_TZ",    "Time Zone Id",      sMethodName, bDebugLog);
……
   xSetupRunTimeEventWithPA("User", "PreSetFieldValue", "Home Phone #",    1,
"paContact_PrevHome", "Home Phone #",      sMethodName, bDebugLog);
   xSetupRunTimeEventWithPA("User", "SetFieldValue",   "Home Phone #",    1,
"paContact_Home",    "Home Phone #",      sMethodName, bDebugLog);
   xSetupRunTimeEventWithPA("User", "PreSetFieldValue", "Fax #",    1,
"paContact_PrevFax", "Fax #",      sMethodName, bDebugLog);
   xSetupRunTimeEventWithPA("User", "SetFieldValue",   "Fax #",    1,
"paContact_Fax",    "Fax #",      sMethodName, bDebugLog);
   xSetupRunTimeEventWithPA("User", "PreSetFieldValue", "Time Zone Id",    1,
"paContact_PrevTZ",  "Time Zone Id",      sMethodName, bDebugLog);
   xSetupRunTimeEventWithPA("User", "SetFieldValue",   "Time Zone Id",    1,
"paContact_TZ",    "Time Zone Id",      sMethodName, bDebugLog);
…….
```

```
   xSetupRunTimeEventWithPA("List Mgmt Prospective Contact",
"PreSetFieldValue", "Work Phone #",    1, "paContact_PrevWork", "Work Phone #",
sMethodName, bDebugLog);
   xSetupRunTimeEventWithPA("List Mgmt Prospective Contact", "SetFieldValue",
"Work Phone #",      1, "paContact_Work",    "Work Phone #",      sMethodName,
bDebugLog);
```

**xSetupRunTimeEventWithPA("List Mgmt Prospective Contact",
"PreSetFieldValue", "Fax Phone #",   1, "paContact_PrevFax", "Fax Phone
#",   sMethodName, bDebugLog);
   xSetupRunTimeEventWithPA("List Mgmt Prospective Contact",
"SetFieldValue", "Fax Phone #",   1, "paContact_Fax",   "Fax Phone #",
sMethodName, bDebugLog);**

```
   xSetupRunTimeEventWithPA("List Mgmt Prospective Contact",
"PreSetFieldValue", "Time Zone",        1, "paContact_PrevTZ",  "Time Zone",
sMethodName, bDebugLog);
   xSetupRunTimeEventWithPA("List Mgmt Prospective Contact", "SetFieldValue",
"Time Zone",         1, "paContact_TZ",      "Time Zone",        sMethodName,
bDebugLog);
……
}
```

9. Edit the code for the Contact_WriteRecord method of the Genesys CampSynch Event Handler Business Service. The new lines of code are highlighted below as **Bold**:

```
function Contact_WriteRecord(bDebugLog)
{
…..
  var sPrevWork = TheApplication().GetProfileAttr("paContact_PrevWork");
  var sWork     = TheApplication().GetProfileAttr("paContact_Work");
```
  **var sPrevFax = TheApplication().GetProfileAttr("paContact_PrevFax");
  var sFax   = TheApplication().GetProfileAttr("paContact_Fax");**
```
  var sPrevTZ   = TheApplication().GetProfileAttr("paContact_PrevTZ");
  var sTZ       = TheApplication().GetProfileAttr("paContact_TZ");
……
    if ( (sPrevDNC  != sDNC) ||
       (sPrevHome != sHome) ||
       (sPrevWork != sWork) ||
```
       (sPrevFax != sFax) ||
```
       (sPrevTZ   != sTZ) )
……
    TheApplication().SetProfileAttr("paContact_PrevWork","");
    TheApplication().SetProfileAttr("paContact_Work","");
```
    **TheApplication().SetProfileAttr("paContact_PrevFax","");
    TheApplication().SetProfileAttr("paContact_Fax","");**
```
    TheApplication().SetProfileAttr("paContact_PrevTZ","");
    TheApplication().SetProfileAttr("paContact_TZ","");
……
```

}

End of procedure

Next Steps

• Add the Genesys Administration screen to the Page tab of the Application object that you are using. See, Procedure: Campaign Synchronization Component: Configuring the Page tab of the Application object.

Configuring the Site Map of the Application Object

The following procedure describes how to add a link to the Administration -Campaign Synchronization screen on the Site Map of the Application object.

Procedure: Campaign Synchronization Component: Configuring the Site Map of the Application object

Purpose: To configure the Site Map of the Application object.

Start of procedure

1. Check-out or lock the project that corresponds to the Application object that you are using. See, Procedure: Campaign Synchronization Component: Checking out existing projects, on page 144

2. In Siebel Tools, in the Project field of the Object Explorer, select the name of the project of the Siebel application that you are using—for example, Siebel Universal Agent.

3. Select the Application folder.

   The Application window displays.

4. Select the Application object that you are using.

5. Using the Types tab of the Object Explorer, select the Screen Menu Item folder under the Application folder.

   The Application Screen Menu Items window displays.

6. Add a new record into the Screen Menu Items table using the following column values listed in Table 11.

Table 11: The Column Values for the Screen Menu Items Table

Column	Value	Comment
Screen	Genesys Administration	
Sequence	<positive number>	Choose an appropriate number that is unique for all of the existing Screen Menu Items tables.
Text-String Reference	X_GEN_GENESYS_ ADMINISTRATION	Choose any string reference—for example, use the default string reference.

End of procedure

Next Steps

• Add the Genesys Administration screen to the Page tab of the Application object that you are using. This procedure is optional. See, Procedure: Campaign Synchronization Component: Configuring the Page tab of the Application object.

Configuring the Page Tab of the Application Object

The following procedure describes how to add the Genesys Administration screen to the Page tab of the Application object.

Procedure:
Campaign Synchronization Component: Configuring the Page tab of the Application object

Purpose: To configure the Page tab of the Application object, if you need a permanent Page tab for the Campaign Synchronization administration. This procedure is optional.

Start of procedure

1. Check-out or lock the project that corresponds to the Application object that you are using. See, Procedure: Campaign Synchronization Component: Checking out existing projects.

2. In Siebel Tools, in the Project field of the Object Explorer, select the name of the project of the Siebel application that you are using—for example: Siebel Universal Agent.

3. Select the Application folder.

    The Application window displays.

4. Select the Application object that you are using.

5. Using the Types tab of the Object Explorer, select the Page Tab folder under the Application folder.

    The Application Page Tabs window displays.

6. Add a new record into the Page Tabs table using the following column values listed in Table 12:

Table 12: The Column Values for the Page Tabs Table

Column	Value	Comments
Screen	Genesys Administration	
Sequence	<positive number>	Choose an appropriate number that is unique for all of the existing Page Tabs tables.
Text—String Reference	X_GEN_GENESYS_ADMINISTRATION	Choose any string reference—for example, use the default string reference.

End of procedure

Next Steps

•   Compile the Siebel repository file. See, Procedure: Campaign Synchronization Component: Compiling the Siebel Repository File, on page 165.

Compiling the Siebel Repository File

You can either compile all of the projects, or you can only compile the locked projects.

Procedure:
Campaign Synchronization Component: Compiling the Siebel Repository File

Purpose: To compile the Siebel repository file.

Start of procedure

1. In Siebel Tools, select Tools > Compile Projects...
2. Select Locked projects.
3. In the Siebel Repository File edit box, enter the name of the repository file.
4. Click Compile.

   The status bar at the bottom of the Object Compiler window indicates when the compilation is finished.
5. When the compilation is finished, close Siebel Tools.

Note: It is possible for siebel_assert_xxx.txt file(s) to be generated by the Siebel environment after compiling the Siebel repository file. This does not affect the functionality of the Component in any way.

End of procedure

Next Steps

- Configure the Siebel Server. See the section, "Configuring the Siebel Server".

Configuring the Siebel Server

Complete the following procedure, described in this section, to configure the Campaign Synchronization Component to work with the Siebel Server.

Procedure:
Campaign Synchronization Component: Configuring the Siebel Server

Purpose: To configure the Siebel Server to work with the Campaign Synchronization Component.

Start of procedure

Deploying the new Genesys Projects

1. Deploy the new Genesys projects to the Siebel Server repository by checking in the following projects:

   a. In Siebel Tools, select Tools > Check In...

      The Check In dialog box displays.

   b. In the Projects list, select the following projects:
      — Campaign
      — Genesys Campaign Synchronization
      — Genesys Symbolic Strings

   c. Click Check In.

Creating Genesys Project Tables on the Siebel Server Database

2. Create the Genesys project tables on the Siebel Server Database by completing the following steps:

   Note: The Campaign Synchronization Component requires its own tables, which do not exist in the original Siebel database.

   ---

   a. In Siebel Tools, in the Table field of the Object Explorer, make a query to select four new Genesys tables: CX_GCAS_ASSIGN, CX_GCAS_CFG_OBJ, CX_GCAS_QUEUE_ITM and CX_GCAS_SUMMARY.

      You can use the first six characters common to all of the table names (CX_GCAS*) to select them all.

   b. With the four tables selected, right-click Add to Archive.

      The Export to Archive File window displays the four tables.

   c. Select the appropriate temporary file name and click Save.

   d. Close the Siebel Tools application, but open it again under the Siebel Server database.

   e. Import the newly created archive file.

   f. Click Next twice.

   g. Click Finish.

   h. In Siebel Tools, in the Project field of the Object Explorer, select Genesys Campaign Synchronization2.

   i. Select the Table folder.

      The Tables window appears, displaying the four newly added tables.

   j. Click Apply.

      A pop-up warning appears.

   k. Click OK to accept the warning.

   l. In the new Apply Schema window, from the Tables drop-down list, select Current Query.

　　　　　　　m. Enter the correct values for the Database user, Database user password and ODBC data source.

　　　　　　　For more information about creating custom tables, see the Siebel documentation.

　　　　　　　n. Click Apply to start the creation process.

　　　　　　　The Changes successfully applied message appears, indicating that the tables were created.

　　　　　　　o. Click Activate to activate these tables.

Deploying the **3.** Deploy the compiled Siebel repository file on your Siebel Server.
Repository File

　　　　　　　For further information about how to deploy an updated Siebel repository file to the Siebel Server, see the Siebel documentation.

Figure 20: Applying New Tables in a Siebel Database

Deploying New Images to the Siebel Server

4. To deploy the new images to the Siebel Server, copy the image files from the following folder, <target>/objects/IMAGES, to the corresponding Siebel Web Server extension folder, (<SWEApp>/public/<language_code>/images).

If your Siebel is configured to use Open UI, copy the images to the folder <SiebSrvr>\WEBMASTER\images\enu, where SiebSrvr is the folder where Siebel server is installed.

End of procedure

Next Steps

- Configure Siebel using the Siebel Web Client. See section, "Configuring Siebel Using the Siebel Web Client" on page 168.

Configuring Siebel Using the Siebel Web Client

You must connect to your Siebel Server using the Siebel Web Client.

Note: After deploying the newly compiled Siebel repository file and restarting the Siebel Server, you must log in as a Siebel Administrator for the Siebel Web Client.

This part of the configuration process includes the following steps:

Importing the Associated List of Values (LOV) and the Application View Data

The following procedure, Procedure: Campaign Synchronization Component: Importing the List of Values and the Application, Views data, describes how to import a List of Values and the Application Views data, that are specific for the Campaign Synchronization Component.

Procedure:
Campaign Synchronization Component: Importing the List of Values and the Application, Views data

Purpose: To import the List of Values and the Application, Views data for the Campaign Synchronization Component.

Start of procedure

1. Log in as a Siebel administrator.

2. Select Site Map > Administration > Business Service, Simulator.

3. Create new record and specify Genesys CampSync Tools as the Service Name.

4. Specify ImportAll as the Method Name.

5. Set Iterations to the value of 1.

6. In the Input Arguments applet, click the Load From File button, browse for the following target directory, <target directory>/objects/<LANG>/GplusCaS_LOV.xml, where <LANG> is the subfolder name (ENU, by default) depending on the Siebel locale you use.

7. Load the GplusCaS_LOV.xml file.

8. Click the Run on the Service Methods applet.

9. Check the results of the import in the Output Arguments applet.

End of procedure

Next Steps

*   Deploy the Workflow Processes on the Siebel Server. See, Procedure: Campaign Synchronization Component: Deploying workflow processes on the Siebel Server, on page 170.

Deploying the Workflow Processes on the Siebel Server

The workflow processes help automate the business processes and they are defined within Siebel to emit or to receive Siebel XML-messages. The Campaign Synchronization Component requires that two workflow processes be deployed on the Siebel Server.

Procedure:
Campaign Synchronization Component: Deploying workflow processes on the Siebel Server

Purpose: To deploy the workflow processes on the Siebel Server.

Start of procedure

1. Select Site Map > Administration > Business Process, Workflow Deployment.

2. In the Active Workflow Processes applet, choose the Import Processes menu option.

   The Workflow Process Import pop-up window displays.

3. In the Workflow Process Import window, select the Genesys_CaS_Campaign_WF.xml file.

   This file is located at: <target directory>/ objects/<Siebel version>, where <Siebel version> is the subfolder name (7.7, 8.0, 8.1, 8.1_8.2_OUI or IP2014, depending on the Siebel version being used).

4. Click Import.

   The Genesys CaS Campaign WF process appears in the applet when the import is finished.

5. Repeat steps 3 and 4 for the Genesys_CaS_CR_WF.xml file. The Genesys CaS CR WF process appears when the import is finished.

End of procedure

Next Steps

- Create the runtime events that are required by the Campaign Synchronization Component. See the section, "Creating Runtime Events".

Creating Runtime Events

To create the runtime events required by the Campaign Synchronization Component, you have to run the InstallRunTimeEvents method of the Genesys CampSynch Tools Business Service once using the Siebel Business Service Simulator applet.

Note: The runtime events used by the Campaign Synchronization Component replace functions that are performed by scripts in the older versions of the component. Make sure that the runtime events from the previous version of the Campaign Synchronization Component are deleted.

Procedure:
Campaign Synchronization Component: Creating the runtime events

Purpose: To create the runtime events that are required by the Campaign Synchronization Component.

Start of procedure

1. Select Site Map > Administration > Business Service, Simulator.

2. In the Service Methods applet, create a new record.

3. Specify the following parameters for the new record:
   - Service Name: Genesys CampSynch Tools
   - Method Name: InstallRunTimeEvents
   - Iterations: 1

4. Click Run.

Note: No results are propagated through the Output Arguments applet.

End of procedure

Next Steps

- Assign the correct responsibility to an agent. See, Procedure: Campaign Synchronization Component: Assigning the correct responsibility to an agent.

Assigning the Correct Responsibility to an Agent

The Campaign Synchronization Component contains an administration GUI that is provided within a special screen and views. To access these views, a special Genesys CampSynch Administrator Siebel responsibility is created during the import of the List of Values (LOV), the Application, Views data and the workflow processes.

Procedure:
Campaign Synchronization Component: Assigning the correct responsibility to an agent

Purpose: To assign the correct responsibility to an agent that is dedicated to managing the campaign synchronization process.

Start of procedure

1. Select Site Map > Administration > Application, Responsibilities.

2. On the Responsibilities applet, select the Genesys CampSynch Administrator responsibility.

3. Add the required agents by using the Users applet on the Responsibilities tab.

Note: The assigned agents should have permission to access and create Profile Configurations (Site Map > Administration > Server Configuration, Profile Configuration). These procedures are required when setting up a campaign synchronization. They are described in the *Gplus Adapter for Siebel CRM User's Guide* as part of the campaign synchronization configuration procedure. If the agents, who are assigned to manage the campaign synchronization, do not have the ability to access and create new Profile Configurations then perform these procedures yourself while logged in as a Siebel administrator.

End of procedure

Next Steps

• Restart the Siebel Server. See the section, "Restarting the Siebel Server".

Restarting the Siebel Server

Restart the Siebel Server that you are using to make the List of Values (LOV) and the other data available to use.

![Genesys logo]

6 Deploying the Voice Component

This chapter describes how to configure and install the G*plus* Adapter for Siebel CRM Voice Component, which is a driver-based component and contains the following sections:

The Voice Component functionality can be used in any or all of the following configurations:

- Voice-only call centers (inbound calls)
- Expert Contact functionality (CTI-Less T-Server support)
- Outbound Campaigns (outbound campaign calls)

- Universal callback functionality (voice callback requests)

See the G*plus Adapter 8.0 for Siebel CRM User's Guide* for additional information about using this component.

Before you add any other driver-based Adapter Components, you must configure the Voice Component, including all of the Voice features that you intend to use.

> **Notes:** Server-based components, such as Campaign Synchronization and Configuration Synchronization, must be configured before the Voice Component.
>
> The Genesys G*plus* Communication Server is a prerequisite for the Voice Component.

This release of the G*plus* Voice Component supports several major features, including the following:

- Basic Voice
- Expert Contact
- Outbound Campaign
- Universal Callback (Voice features)

Overview

The G*plus* Adapter 8.0 for Siebel CRM Voice Component provides a single, consistent user interface for customer relationship management, telephony control, outbound-campaign calls, expert contact, and voice callback functionality.

Features and Enhancements

The Voice Component supports the following features, described below, any or all of which can be implemented based on the needs of your contact center:

- Basic Voice (inbound and outbound call functions)
- Expert Contact with CTI-Less T-Server support
- Outbound Campaign (outbound campaign calls)
- Universal Callback (voice callback requests)

Basic Voice

The Basic Voice feature is intended for call centers that work with inbound calls. This feature must be imported first, because it defines the Genesys Voice Project in Siebel.

Expert Contact

The Expert Contact feature serves to establish connections with associates, known as experts or knowledge workers, who are not regular call center agents, and whose phones are not directly monitored by T-Server. Usually this expert is a person who has the advanced skills required to handle specific customer requests, or to solve customer problems. The expert has the option to preview data about an incoming call in Preview Interaction mode, and to manually notify the CTI-less T-Server about an incoming call and the progress of that call. Because the T-Server has no CTI-link to the switch, it does not necessarily receive notification of the expert's phone interactions, so the expert must use the Agent Desktop Toolbar to manually update the status of the interaction.

For additional details about Expert Contact, refer to the *Genesys Expert Contact 7 Getting Started Guide*, or the *Genesys Expert Contact 7 Deployment Guide.*

Outbound Campaign

The Outbound Campaign feature is intended for call centers that primarily work with outbound campaign calls. It supports Genesys Outbound Desktop Protocol version 7.0. The Outbound Campaign feature provides integration of Siebel CRM with the Genesys Outbound Contact Server capabilities. The Outbound Campaign feature operates with the Voice Component and implements full functionality of the Outbound Contact desktop on the Siebel CRM Communications Toolbar for outbound campaign functionality. For more information about the Siebel CRM Communications Toolbar, see the *Gplus Adapter 8.0 for Siebel CRM User's Guide*.

Universal Callback

The Universal Callback feature is used with the Voice Component to provide voice and web callback functionality. For example—using voice callback functionality in your company's Call Center IVR, a customer can request that a representative from your company call the customer back. The customer simply selects the "Call Back" option in the IVR menu and then enters a telephone number with an approximate callback time. In the section titled "Checking the Voice Callback Installation" on page 273, Voice Callback scenarios provide additional details about using this feature.

Each incoming voice callback interaction creates an activity record in the Siebel application and a new work item on the agent Communications Toolbar.

The Description field from the activity record is used for storing user data fields passed from the Universal Callback Server. The set of user data fields that are stored can be modified to meet contact center requirements.

New in This Release

This section provides information about new features or functionality in the Voice Component.

Release 8.0.210 No new features were introduced in this release.

Release 8.0.2 The following new features or functionality are included in this release:

- Support for Siebel versions 8.1.1.14/8.2.2.14 (IP2014) for both HI and Open UI modes.
- Support for SIP Business Continuity.
- Support for Genesys Framework 8.5.
- Support for Hoteling (Free Seating).
- Support for Red Hat Enterprise Linux 6.
- Support for Red Hat Enterprise Linux 7.
- Support for AIX 7.1.
- Support for Solaris 11.

Release 8.0.110 The following new features or functionality are included in this release:

- Support for Siebel versions 8.1.1.11/8.2.2.4 (IP2013) for both HI and Open UI modes.

Release 8.0.1 No new features were introduced in this release.

Release 8.0.0 The following new features or functionality are included in this release:

- Support for the Push Preview dialing mode for Outbound Campaigns.
- Support for the Red Hat Enterprise Linux (RHEL) operating system.

Installation

This section describes the installation process for the G*plus* Adapter 8.0 for Siebel CRM Voice Component. The G*plus* Adapter can be installed on Windows or UNIX platforms. On UNIX, you can use either Solaris or AIX.

Select one of the following sections depending on your environment:

Overview of Voice Component Installation

The procedure involved in installing the Voice Component and setting up its standard features are summarized below:

Procedure:
Voice Component: Installing the Voice Component

Purpose: To install the Voice Component.

Start of procedure

1. Install the G*plus* Adapter from the installation package.

2. Using Genesys Configuration Manager, update the Genesys configuration and collect the required information.

3. Using Siebel Tools, import the Siebel repository file (*.srf) archive files from the installation package.

4. Customize the Siebel Communications Toolbar.

5. Recompile the Siebel *.srf file.

6. Deploy the Siebel *.srf file on the Siebel Server.

7. Configure the Siebel Call Center application.

8. Synchronize the time zone names in Siebel and Genesys.

9. Customize and import the Communications configuration file, GenComm_universal.def.

10. Create and administer the call center agents.

11. Start a test session for each feature, based on instructions in the section titled, "Checking Installations" on .

End of procedure

Each of the processes listed above are explained in the sections below, and explanations are accompanied with step-by-step instructions for completing each task in the process.

Note: Customization of the Siebel application and the G*plus* Adapter is very common. No overview, including the one provided above, could encompass all of the possible customizations of the G*plus* Adapter. However, some of the more common G*plus* Adapter customizations, and some general information related to the G*plus* Adapter customization, is provided in the section titled "Advanced Customization of Voice Component Configurations" on .

Next Steps

*   Install the Gplus Adapter 8.0 for Siebel CRM Voice Component. Select one of the following sections, depending on your environment:
    *   "Installing the Voice Component in a Windows Environment" on

Setup Types

This version of the Adapter is compatible with the following versions of Siebel CRM Server: 7.7, 7.8, 8.0, and 8.1.

For information about the files that you install and their location, see the section "Target Directory Structure and File Locations".

Target Directory Structure and File Locations

The following generic directory and host names are used in the description of the directories structure:

• <Gplus Communication Server directory> is the G*plus* Communication Server installation directory.

• <Destination Directory> is the destination directory for installation used by the installation script to copy the Adapter for Siebel CRM Voice files.

• Files in subdirectory /7.7 <Destination Directory>/7.7 are related to Siebel versions 7.7 /7.8.

• Files in subdirectory /8.0 <Destination Directory>/8.0 are related to Siebel version 8.0.

• Files in subdirectory /8.1 <Destination Directory>/8.1 are related to Siebel version 8.1.

• Files in subdirectory /8.1_8.2_OUI <Destination Directory>/8.1_8.2_OUI are related to Siebel 8.1.1.11/8.2.2.4 (IP2013).

• Files in subdirectory /IP2014 <Destination Directory>/IP2014 are related to Siebel 8.1.1.14/8.2.2.14 (IP2014).

• <Web Server Host> is the Web Server host name where the Siebel Web Server extension was installed.

• <SWEIconImages directory> is the Siebel Web Server extension icon images directory.

Installing the Voice Component in a Windows Environment

This section describes how to install the Voice Component on a Windows system.

Note: Before you install the Voice Component, stop the G*plus* Communication Server.

Procedure:
Voice Component: Installing the Voice Component in a Windows environment

Purpose: To install the Voice Component in a Windows environment.

Start of procedure

1. Run the setup.exe file from the installation package (windows\setup.exe).

2. When prompted with the following message, Genesys Installation Wizard will install the Gplus Adapter for Siebel CRM Voice in the following Destination Folder, choose the location of this destination folder.

   For example:

   C:\Program Files\GCTI\Gplus Adapter for Siebel CRM Voice

3. When prompted with the following message, Genesys Installation Wizard has collected all required information and is ready to install Gplus Adapter for Siebel CRM Voice on your computer, click Install.

4. When the Genesys Installation Wizard completes its execution, you must *manually* copy the following files to their proper locations:

   Copy the files representing the Siebel icon images from the destination directory, <Destination Directory>/<Siebel Version>/images/, where <Siebel Version> is 7.7/8.0/8.1/8.1_8.2_OUI/IP2014, depending on your version of Siebel Server, to the following icon images directory: < Web Server Host>/<SWEIconImages directory>.

   If your Siebel is configured to use Open UI, copy the images to the folder <SiebSrvr>\WEBMASTER\images\<LNG>, where SiebSrvr is the folder where Siebel server is installed and LNG is the language code.

   For Siebel version 7.8, use the 7.7 directory.

Note: The installation package contains new complementary pairs of icons for Ready/Not Ready buttons. During the installation process, standard Siebel icons for the Not Ready button are overwritten with new ones from the installation package. To preserve the standard Siebel icons, Genesys recommends that you back up image files icon_notready_enabled.gif and icon_notready_disabled.gif in a safe location. These files should be restored manually if the G*plus* Adapter is uninstalled.

End of procedure

Next Steps

- Configure the Genesys part of the Voice Component. See the section, "Configuring Genesys" on .

Installing the Voice Component in a UNIX Environment

This section describes how to install the Voice Component on the Solaris and AIX operating systems.

The user, running the installation script, must have proper permissions to create and write to the directories where the G*plus* Communication Server (GenCommServer) is installed.

Note: Do not use special symbols in any destination directory name when installing in a UNIX environment.

Select one of the following sections depending on your environment:

- Prerequisites for Solaris
- Prerequisites for AIX
- Prerequisites for Linux

Prerequisites for Solaris

The Voice Component must be installed on a computer where the G*plus* Communication Server (GenCommServer) is installed on Solaris. Before starting the installation of the Voice Component on Solaris, you must have the following dynamically linked runtime libraries installed on the computer where Siebel CRM is installed:

- libsocket.so
- libnsl.so
- libdl.so
- librt.so
- libpthread.so
- libm.so
- libc.so
- libmp.so
- libaio.so
- libthread.so
- libc_psr.so

Follow the instructions under the heading, "Running the Installation Shell Script" on page 183 to continue with the installation process.

Prerequisites for AIX

The Voice Component must be installed on a computer where the G*plus* Communication Server (GenCommServer) is installed on an AIX platform. Before starting the installation of the Voice Component on AIX, you must have

the following dynamically linked runtime libraries installed on the computer where Siebel CRM is installed:

- libpthreads.a

- libC.a

- libc.a

- librtl.a

- libcrypt.a

Follow the instructions under the heading, "Running the Installation Shell Script" on to continue with the installation process.

Prerequisites for Linux

The Voice Component must be installed on a computer where the Gplus Communication Server (GenCommServer) is installed on an Linux platform. Before starting the installation of the Voice Component on Linux, you must have the following dynamically linked runtime libraries installed on the computer where Siebel CRM is installed:

- libdl.so

- libm.so

- libgcc_s.so

- libc.so

- libpthread.so

- libstdc++.so

- ld-linux.so

Follow the instructions under the heading, "Running the Installation Shell Script" to continue with the installation process.

Running the Installation Shell Script

The next step in the installation process is to install the shell script.

Procedure:
Voice Component: Running the installation shell script

Purpose: To install the Voice Component in Solaris, AIX, or Linux. environments.

Start of procedure

1. Stop the G*plus* Communication Server.

2. At the Solaris, AIX, or Linux prompt, execute the install.sh file.

3. When prompted with the message, Please enter the full path to your Gplus Communication Server for Siebel CRM directory =>, enter the full path to the G*plus* Communication Server for Siebel CRM installation directory—for example:

/data/GplusCommunicationServerFolder

4. When the following message appears:

Note that path to "Gplus Communication Server for Siebel CRM directory" directory can't be used as input to the "Destination directory for installation" prompt.

Please enter full path of the destination directory for installation

=>

enter the full path of the destination directory for installation—for example:

/data/inst_dest

Files are extracted from the installation package to the destination directory for installation, and then are copied into the correct folders.

5. When the installation script completes its execution, you must *manually* copy the following files to their proper locations.

Depending on your version of the Siebel Server, copy the files representing icon images from <Destination Directory>/<Siebel Version>/images/ to the following directories:

< Web Server Host>/< SWEIconImages directory >

If your Siebel is configured to use Open UI, copy the images to the folder <SiebSrvr>\WEBMASTER\images\enu, where SiebSrvr is the folder where Siebel server is installed.

For Siebel version 7.8, use the 7.7 directory.

Note: The installation package contains new complementary pairs of icons for the Ready/Not Ready buttons. During the installation process, the standard Siebel icons for the Not Ready button are overwritten with new ones from the installation package. To preserve the standard Siebel icons, Genesys recommends that you back-up the following image files, icon_notready_enabled.gif and icon_notready_disabled.gif, in a safe location. These files should be restored manually, if the G*plus* Adapter is uninstalled.

End of procedure

Next Steps

- Configure the Genesys part of the Voice Component. See the section, "Configuring Genesys" on .

Uninstalling the Voice Component

Complete information about uninstalling the G*plus* Adapter and its components is provided in Chapter 14, "Uninstallation Instructions," on page 565.

Configuring Genesys

This section describes the configuration of the Genesys requirements of the Voice Component. It includes information and instructions for configuring the Voice Component functionality.

Configuring the Genesys section of the Voice Component consists of the following sections:

Prestart Information

Before starting the configuration process for each G*plus* Adapter component or feature listed below, you must have the listed applications installed and running:

Required for All Components

Genesys Framework

• Configuration Database
• Configuration Server
• Configuration Manager
• T-Servers

Expert Contact Feature

Genesys Expert Contact

• CTI-Less T-Server

Outbound Campaign Feature

Genesys Outbound Campaign

• Outbound Contact Server
• Outbound Contact Manager

Universal Callback Feature

Genesys Framework

- Universal Router
- Universal Callback Server

Modifying Genesys Settings

Genesys Configuration Manager provides access to most of the following information mentioned in this section:

- Basic Voice Feature
- Expert Contact Feature
- Outbound Campaign Feature

Basic Voice Feature

The Basic Voice feature does not require any Genesys settings update. No mandatory modifications are needed.

Expert Contact Feature

You may need to change configuration settings when a Network T-Server is used with respect to CTI-Less T-Server. Most configuration changes must be done in the Network T-Server configuration object. Refer to the Genesys *Expert Contact 7 Deployment Guide* for specific information and instructions.

Outbound Campaign Feature

Several Outbound Contact Server (OCS) installation settings must be changed after the default installation of the Outbound Campaign feature. You can make these changes in Configuration Manager using the Options tab of the OCS application properties, as described in the following procedure, Procedure: Voice Component: Modifying Genesys Outbound Contact Server settings.

Procedure:
Voice Component: Modifying Genesys Outbound Contact Server settings

Purpose: To modify the Genesys Outbound Contact Server settings.

Start of procedure

1. In Configuration Manager, select Environment > Applications and right-click the Outbound Contact Server Application.

2. Select Properties.
   The Properties dialog box displays.

3. Select Options > OCServer (see Figure 21).

   To change the default value of the key, double-click the line you want to change:
   outbound_release_action > hard_not_ready
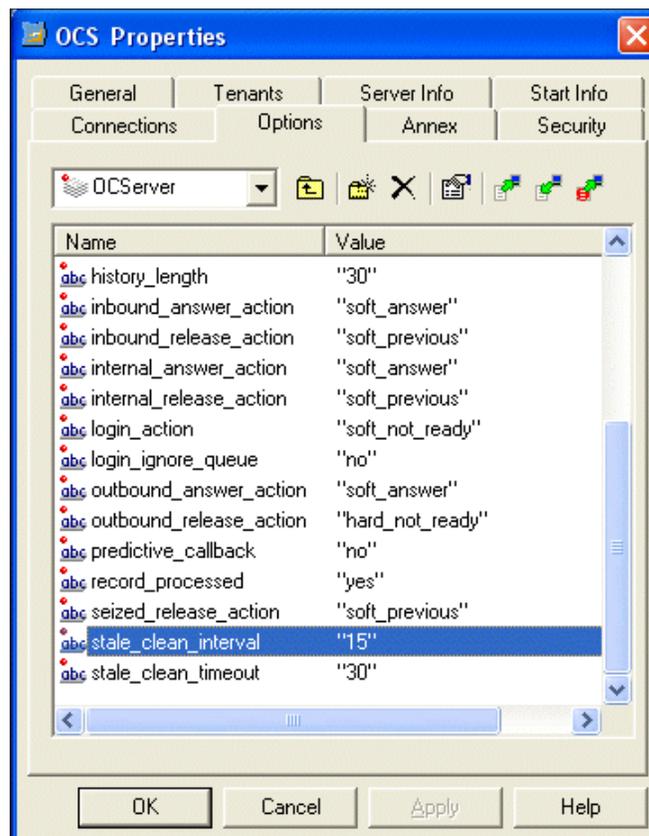
Figure 21: Modifying Outbound Contact Server Settings

4. Check to make sure the configuration option, record_processed, is set to yes. If not, change the value to yes. For more information about the record_processed option, see the *Genesys Outbound Contact 7.0 Reference Manual.*

5. If you are using Outbound Contact release 7.0 or later, you can add the configuration option customer_id, value = crm_contact_id (for more information, see the topics "The customer_id Configuration Option" on page 190 and "Siebel CRM Applications" on page 196).

    If you are using an earlier version of Outbound Contact, continue to Step 6.

6. Click Apply to submit changed values.

7. Click OK.

End of procedure

Next Steps

- Set the GSW_ATTEMPTS attribute. See, Procedure: Voice Component: Setting the GSW_ATTEMPTS attribute, on page 189.

Modifying the Outbound Contact Manager Component Settings

After default installation, no further changes are required for Outbound Contact Manager.

For further information on how to install Outbound Contact Manager, see the *Genesys Outbound Contact Getting Started Guide*.

Setting the Send Attribute Values in the Calling List Format

The send_attribute = GSW_ATTEMPTS key-value pair must be specified in the Attempt field in the calling list format. You must specify this value because after the default installation, Outbound Contact Server does not send the number of attempts to the client, and vice versa. Figure 22 displays the attempt Properties dialog box where you specify the key-value pair.
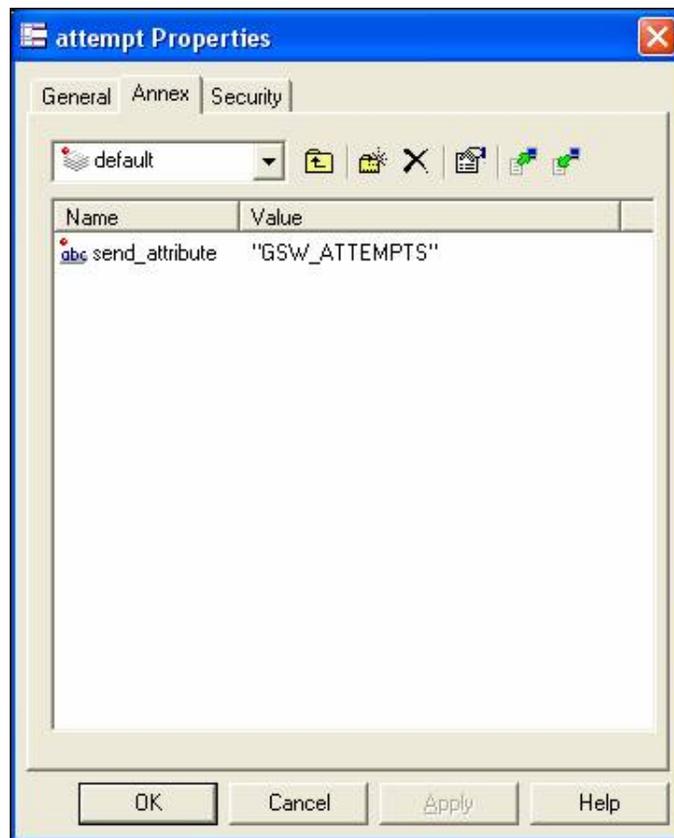
Figure 22: Setting the Send Attribute

For more information on how to set the send_attribute = GSW_ATTEMPTS key-value pair, see Procedure: Voice Component: Setting the GSW_ATTEMPTS attribute, on page 189.

Procedure:
Voice Component: Setting the GSW_ATTEMPTS attribute

Purpose: To set the GSW_ATTEMPTS attribute.

Start of procedure

1. In Configuration Manager, select the section Formats, used by your campaign.

2. Browse to the appropriate Format record.

3. Right-click the Attempt field and select Properties from the shortcut menu.

4. Select the Annex tab.

5. Create the section default and under this section, create a new record called send_attribute = GSW_ATTEMPTS (see "Setting the Send Attribute Values in the Calling List Format" on page 188).

6. Click OK to save the record.

End of procedure

Next Steps

• Gather the Genesys settings. See the section, "Gathering the Genesys Object Settings" on page 191.

DoNotCall Requests Based on Siebel Customer ID

The Do Not Call (DNC) function prevents a particular telephone number or customer ID from being dialed. A Do Not Call request can be handled during either an outbound call or an inbound call. The Campaign Synchronization Component supports Do Not Call requests based on the Siebel customer ID (this requires Genesys Outbound Contact Server release 7.0, or later). The Do Not Call request is an extension of the Outbound Contact Server (OCS) desktop protocol. You can also use the Do Not Call request based on the record handle and phone number. Setting the Do Not Call request based on the customer's ID resolves the problem of more than one customer sharing the same phone number. For more information about the Do Not Call request, see the *Outbound Contact Deployment Guide.*

The customer_id Configuration Option

To support the Do Not Call request based on the Siebel customer ID, the Outbound Contact Server associates a Calling List table's field with their customer ID. The name of the Calling List table's fields that are used for the customer ID is defined by the configuration option, customer_id (see the section, "Specifying the Do Not Call Command Attribute Keys" on page 259). If this option is not present in the calling list or the Application object, or if it contains a field name that does not exist in the Calling List table, by default, Outbound Contact Server uses the phone number to determine the records that should not be dialed.

Gathering the Genesys Object Settings

You must find and record several Genesys object settings in Genesys Configuration Manager before you proceed any further with the Voice Component installation.

Warning!   The data identified in this section is required to complete the installation.

The specific Genesys data that you require depends on the G*plus* Adapter components and features that you plan to implement for your Contact Center. The data requirements for each configuration are listed as follows under the corresponding headings below:

- Data Required for All Voice Configurations, page 191
- Basic Voice Feature Data, page 192
- Expert Contact Feature Data, page 192
- Outbound Campaign Feature Data, page 193
- Outbound Contact Server Configuration Parameters, page 193
- Universal Callback Feature Data, page 195

Data Required for All Voice Configurations

For all Voice configurations, complete the following:

- Find and record the T-Server connection information. See, Procedure: Voice Component: Finding the T-Server connection information, on page 191 for the steps on how to gather this connection information.

  The required information includes:

  - The T-Server application name.

  For the pre-7.5 description schema of connecting to the T-Server, the required information includes:

  - The T-Server host name
  - The T-Server connection port
  - The switch type information
  - The backup T-Server host name (if applicable)
  - The backup T-Server connection port (if applicable).

Procedure:
Voice Component: Finding the T-Server connection information

Purpose: To find the T-Server connection information.

Start of procedure

1. In Configuration Manager, open the Environment view and then select the Applications view.

2. From the toolbar, click New.

3. In the Browse dialog box, select the Application Template for this application (navigating to the appropriate folder, if necessary), and click OK.

4. On the Configuration tab, enter the following information for the T-Server application configured for use with Outbound Contact Server:
   - Name: Enter the name of the T-Server application.
   - Connections: Click Add, and then add the T-Server application.

5. Open the T-Server application's Properties dialog box.

6. Click the Server Info tab.

7. In the Server Info section, enter the following T-Server connection information (which you will need later for the G*plus* Outbound Campaign feature and for configuring the Siebel Call Center application):
   - Host: Host name
   - Communication port: Communication port number

End of procedure

Next Steps

- No further steps are required.

Basic Voice Feature Data

If your application is configured to receive network calls and to use the Network Attended Transfer/Conference (NAT/C) feature, then the following information is required:

- The premise switches' names as they are defined in Genesys Configuration Manager

Expert Contact Feature Data

For the Expert Contact feature, find and record the following CTI-Less T-Server connection information:

- The T-Server host name
- The T-Server connection port
- The switch type information
- The backup T-Server host name
- The backup T-Server connection port

Based on the actual configuration of Genesys Expert Contact, find the CTI-Less T-Server, and note the corresponding virtual switch name.

Outbound Campaign Feature Data

For the Outbound Campaign feature, find and record the following:

- Gather the T-Server connection information
- Gather the Outbound Contact Server configuration information
- Check the T-Server settings

    The required information includes:

    - The T-Server host name
    - The T-Server connection port
    - The switch type information
    - The backup T-Server host name
    - The backup T-Server connection port
    - The name of the voice queue on which the Outbound Campaign is running
    - The version of the Outbound Contact Server you use. See"Voice Component: Finding the current version of Outbound Contact Server (OCS)" on page 195.

Outbound Contact Server Configuration Parameters

This section describes how to configure the Outbound Contact Server parameters.

Procedure:
Voice Component: Finding the ACD Queue/PBX data information

Purpose: To find the Outbound Contact Server (OCS) ACD (automatic call distribution) Queue/PBX (switch) information.

Start of procedure

1. In Configuration Manager, select Tenant > Agent Groups or Tenant > Place Groups.

2. Select a particular Group.

3. Select Properties.
   The Properties dialog box displays.

4. Select the Advanced tab (see Figure 23).
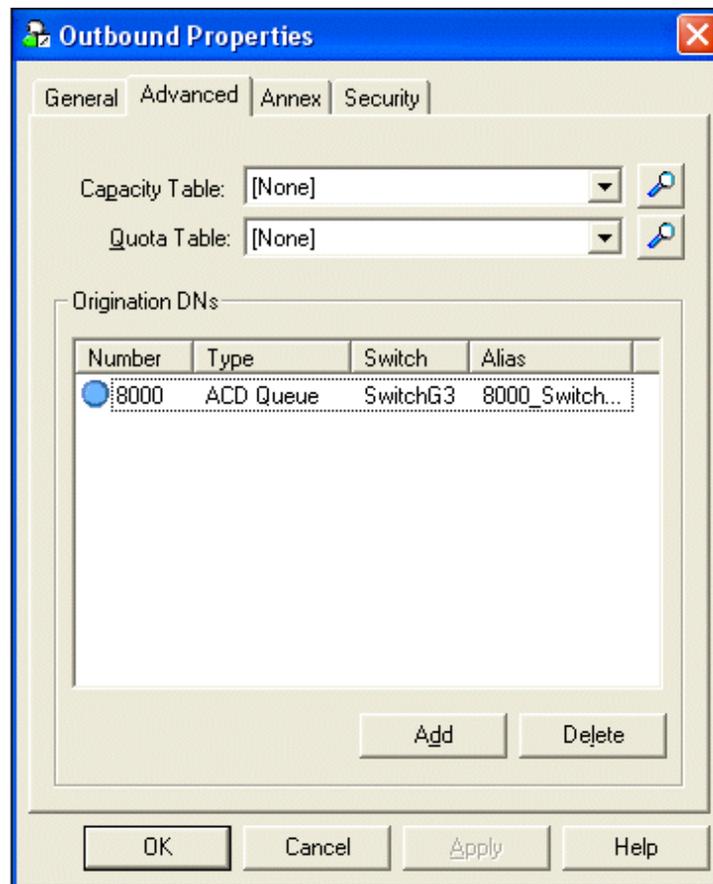
Figure 23: Outbound Properties Dialog Box

5. From the Origination DNs pane, make note of the ACD Queue/PBX
information, which is used later for the G*plus* Voice Component
configuration.

For example, in Figure 23 on page 194, the Origination DNs information
required includes: DN Number (8000), Type (ACD Queue), Switch
(SwitchG3). For further information, see "Configuring Siebel Using the
Siebel Web Client" on page 212 and Procedure: Voice Component:
Associating agents with an existing Communications profile and Genesys
settings, on page 226.

End of procedure

Next Steps

• Find and record the current version of Outbound Contact Server (OCS).
See, Procedure: Voice Component: Finding the current version of
Outbound Contact Server (OCS), on page 195.

Procedure:
Voice Component: Finding the current version of Outbound Contact Server (OCS)

To find and record the current version of Outbound Contact Server (OCS) that you use:

Start of procedure

1. Start at the command prompt window from Windows, or from a terminal window in UNIX.

2. Navigate to the OCS installation home folder (for example—C:\GCTI\OCS on the Windows platform).

3. Run the Outbound Contact Server executable with the -V switch:
   > ocs_server -V

4. Write down the Outbound Contact Server version.

End of procedure

Next Steps

- Configure the Siebel side of the Voice Component. See the section, "Configuring Siebel" on .

Universal Callback Feature Data

The Voice and Web Callback Configuration does not require any Genesys settings update. The time zone synchronization is a mandatory modification for this configuration.

For this feature:

- Gather the T-Server connection information.
- Gather the Universal Callback server information.

This required information includes:

For T-Server:

- The T-Server host name
- The T-Server connection port
- The switch type information
- The backup T-Sever host name
- The backup T-Server connection port

For Universal Callback Server:

- The switch name of Voice T-Server
- The Routing Points list on this switch

Configuring Siebel

This section describes how to configure the Siebel part of the Voice Component for Siebel CRM.

Configuring the Siebel section of the Voice Component consists of the following steps:

Prestart Information

Before starting this part of the configuration process, you should have the following Siebel CRM and Genesys G*plus* applications installed and running:

Siebel CRM Applications

- You must have the Siebel Tools application running.
- You should be connected to the local copy of the Siebel Server database.
- You should have sufficient privileges to check-in and check-out projects from the Siebel Server repository.
- To use the Outbound Campaign feature and the Outbound Contact Server product, you must have the Siebel eMarketing application installed.
- You should ensure that the required steps in "Patching and Configuring Siebel CRM" on page 41 are completed.

Genesys G*plus* Applications

- The Configuration Synchronization Component for Siebel CRM application is optional and may be used with any Voice Component feature.
- The G*plus* Campaign Synchronization Component for Siebel CRM application is optional and it is designed to be used with the Outbound Campaign feature.

Configuring Siebel Using Siebel Tools

Use the Siebel Tools to compile an updated version of the Siebel repository file (SRF or *.srf file), which serves to define the Siebel Contact Center application on the Siebel Server.

For more information about using Siebel Tools, see the Siebel documentation.

Configuring Siebel using the Siebel Tools consists of the following sub-sections:

Overview of the Siebel Repository File Preparation

When compiled, the Siebel repository file integrates information from the various G*plus* Adapter Voice Component archives files (the *.sif files) that you import.

For each Voice Component feature that you wish to implement, you must import a Voice Component *.sif file and resolve any data conflicts introduced by the imported file.

The Voice Component features and their associated *.sif files are listed in Table 13 on page 198. The *.sif files are provided with the G*plus* Adapter installation disk.

Note: Depending on the Siebel Server version, the appropriate *.sif files should be imported from the following destination folders:
- For Siebel 7.7/7.8:<destination folder>/7.7/
- For Siebel 8.0: <destination folder>/8.0/
- For Siebel 8.1: <destination folder>/8.1/
- For Siebel 8.1.1.11/8.2.2.4 (IP2013): <destination folder>/8.1_8.2_OUI/
- For Siebel 8.1.1.14/8.2.2.14 (IP2014): <destination folder>/IP2014

See the following installation sections for more information:
- Procedure: Voice Component: Importing the GenSymbolicStrings.sif archive file, on page 201
- Procedure: Voice Component: Importing the GenComm.sif archive file, on page 203
- Procedure: Voice Component: Importing the GenesysTools.sif archive file, on page 204

Table 13: Voice Component Features and Associated *.sif Files

Feature	Voice Component archive (*.sif) file
Basic Voice	GenSymbolicStrings.sif and GenComm.sif
Genesys Tools	GenesysTools.sif
Expert Contact	GenComm_ECS.sif
Outbound Campaign	GenComm_OCS.sif *
Universal Callback	GenComm_VCB.sif

* In some circumstances, you may have to manually update certain objects to resolve any data conflicts.

Checking Out Existing Projects from the Siebel Repository for the Voice Component

The Siebel implementation of the Voice Component makes use of a number of objects provided by Siebel. use the procedure to enable the modifications of these objects. To do so, you will need to check-out the corresponding projects from the Siebel repository.

Procedure:
Voice Component: Checking out existing projects

Purpose: To check-out existing projects from the Siebel repository.

Start of procedure

1. In Object Explorer, start Siebel Tools against the local database.

2. In Siebel Tools, select Tools > Check Out...

   The Check Out dialog box displays.

3. Select Project category.

4. In the Projects list, select the appropriate projects listed below in Table 14, "Projects for All Supported Siebel Versions - Common for all Components and Feature-Specific," on page 200.
   - Select the projects to be locked based on the features you want to install:
     i. Always lock the projects that are common to all components.
     ii. In addition, lock any other projects that are specific to the features you are installing.

Note: When you first open the Projects list, the Genesys Voice, Genesys Tools, and Genesys Symbolic Strings projects do not yet exist, so are not available to be checked-out and locked.
   - The project named Genesys Symbolic Strings is created when you import the GenSymbolicStrings.sif file, as explained in Procedure: Voice Component: Importing the GenSymbolicStrings.sif archive file, on page 201. Lock the project after importing the GenSymbolicStrings.sif file.
   - The project named Genesys Voice is created when you import the GenComm.sif file, as explained in Procedure: Voice Component: Importing the GenComm.sif archive file, on page 203 below. Lock the project after importing the GenComm.sif file.
   - The project named Genesys Tools is created when you import the GenesysTools.sif file, as explained in Procedure: Voice Component: Importing the GenesysTools.sif archive file, on page 204. Lock the project after importing the GenesysTools.sif file.

Table 14: Projects for All Supported Siebel Versions - Common for all Components and Feature-Specific

Feature	Archive File	Common Projects (for all components)	Feature-Specific Projects
Basic Voice	GenSymbolicStrings.sif GenComm.sif GenesysTools.sif	• Command • Communication • Communication Administration • Persistent Customer Dashboard • Genesys Symbolic Strings	• Genesys Voice* • Genesys Tools* • Employee*
Expert Contact	GenComm_ECS.sif		None
Outbound Campaign	GenComm_OCS.sif		• Campaign
Universal Callback	GenComm_VCB.sif		• Activity • Contact • Contact (SSE) • Table Activity • <Project related to an application on which the Adapter is deployed with the Universal Callback feature >**

* Lock the project after import of the corresponding *.sif file.

** This project is locked when deploying the Universal Callback feature as described in Procedure: Voice Component: Resolving the Siebel application data conflicts, on page 206. This is the project related to an application on which the Adapter is deployed—for example: Siebel Universal Agent (for Siebel Horizontal version) or Siebel Financial Services (for Siebel Vertical versions).

End of procedure

Next Steps

• Import the Voice Component *.sif archive files. See the section, "Importing the Voice Component (*.sif) Archive Files".

Importing the Voice Component (∗.sif) Archive Files

By importing the Voice Component archives, you import the fundamental data required to update the Siebel repository file for the Voice Component.

Deployment Order   There is one limitation on the order in which you deploy the *.sif files—you must first install the GenSymbolicStrings.sif file, and then install the GenComm.sif file. These two files provide the Basic Voice functionality, and are both mandatory for any and all configurations.

The deployment of the GenesysTools.sif file is mandatory as it used during the configuration of the Siebel Call Center application.

There are no restrictions on the order in which you deploy the rest of the archive files. These other files are technically optional, so there is no restriction on the order of their deployment. Each archive file is responsible for a particular feature's functionality.

Before importing the mandatory *.sif files, you must check-out and lock the existing projects from the Siebel repository. See the section, "Checking Out Existing Projects from the Siebel Repository for the Voice Component".

Procedure:
Voice Component: Importing the GenSymbolicStrings.sif archive file

Purpose: To import the mandatory archive file with symbolic strings, GenSymbolicStrings.sif, that is used across the Siebel archive files provider within the Adapter.

Notes: • The GenSymbolicStrings.sif file must be imported first.
• If the Genesys Voice Component project already exists in your Siebel repository file archive, lock it before importing the GenSymbolicStrings.sif archive file. If this project does not exist before export, lock it after the export.

Start of procedure

1. In Siebel Tools, select Tools > Import from Archive...

2. In the Select Archive to Import window, navigate to the GenSymbolicStrings.sif archive file.

    The Voice Component installation program created this file in its destination directory. Navigate to the appropriate directory in your environment, as follows:
    • For Windows: <Destination directory>/<Siebel Version>.
    • For UNIX: <Destination directory>/<Siebel Version>.

3. Click Open.

    The Import Wizard–Preview window displays (see Figure 24 on page 202).

Figure 24: Import Wizard–Preview Window

> **Note:** In the Conflict resolution area, the Merge the object definition from the archive file with the definition in the repository button is selected by default (see Figure 24).

4. In the Import Wizard–Preview window, in the Conflict resolution section, select Overwrite the object definition in the repository.

5. Click Next.

   The Import Wizard–Review Conflicts and Actions window displays.

6. Merge any conflicts, if they correspond to your needs.

7. Click Next.

   The "Do you wish to proceed?" window displays.

8. Click Yes.

   The objects from the archive are imported into the Siebel repository.

9. Click Finish to complete the import.

End of procedure

Next Steps

- Import the GenComm.sif archive file. See, .

Procedure:
Voice Component: Importing the GenComm.sif archive file

Purpose: To import the mandatory GenComm.sif archive file.

Note: If the Genesys Tools project already exists in your Siebel repository file archive, lock it before importing the GenComm.sif archive file. If this project does not exist before export, lock it after the export.

Start of procedure

1. In Siebel Tools, select Tools > Import from Archive.

2. In the Select Archive to Import window, select the Voice Component archive file, GenComm.sif.

The Voice Component installation program created this file in its Destination directory. Navigate to the appropriate directory for your environment, as follows:
- For Windows:

    <Destination directory>/<Siebel Version>
- For UNIX:

    <Destination directory>/<Siebel Version>

3. Click Open.
The Import Wizard–Preview window appears (see).

4. Resolve any Siebel application data conflicts. Refer to Table 15 on to see which Voice Component features have conflicts that need to be resolved. See, .

End of procedure

Next Steps

- Import the GenesysTool.sif archive file. See, .

Procedure:
Voice Component: Importing the GenesysTools.sif archive file

Purpose: To import the GenesysTool.sif archive file.

Note: If the Genesys Tools project already exists in your Siebel repository file archive, lock it before importing the GenesysTools.sif archive file. If this project does not exist before export, lock it after the export.

Start of procedure

1. In Siebel Tools, select Tools > Import from Archive...

2. In the Select Archive to Import window, navigate to the GenesysTools.sif archive file.

   The Voice Component installation program created this file in its destination directory. Navigate to the appropriate directory in your environment, as follows:
   - For Windows: <Destination directory>/<Siebel Version>.
   - For UNIX: <Destination directory>/<Siebel Version>.

3. Click Open.

   The Import Wizard–Preview window displays (see Figure 24 on page 202).

4. Resolve any Siebel application data conflicts. Refer to Table 15 on page 206 to see which Voice Component features have conflicts that need to be resolved. See, Procedure: Voice Component: Resolving the Siebel application data conflicts, on page 206.

End of procedure

Next Steps

- If you are importing the *.sif files for the selected optional Voice Component features, see, Procedure: Voice Component: Importing the *.sif files for the selected optional Voice Component features, on page 205.

- If you are not importing anymore optional *.sif files, you must resolve any outstanding Siebel application data conflicts. See, Procedure: Voice Component: Resolving the Siebel application data conflicts, on page 206.

Procedure:
Voice Component: Importing the *.sif files for the selected optional Voice Component features

Purpose: To import the *.sif files for the selected optional Voice Component features.

Note: If you are deploying the Universal (Voice) Callback feature, see for more information on how to resolve the Siebel application data conflicts.

Start of procedure

1. In Siebel Tools, select Tools > Import from Archive...

2. For every selected optional feature, do the following:

3. In the Select Archive to Import window, navigate to the corresponding *.sif file archive file.

   The Voice Component installation program created this file in its destination directory. Navigate to the appropriate directory in your environment, as follows:

   • For Windows: <Destination directory>/<Siebel Version>.

   • For UNIX: <Destination directory>/<Siebel Version>.

4. Click Open.

   The Import Wizard–Preview window displays. (see Figure 24 on page 202).

5. Resolve any Siebel application data conflicts. Refer to Table 15 on page 206 to see which Voice Component features have conflicts that need to be resolved. See, Procedure: Voice Component: Resolving the Siebel application data conflicts, on page 206.

End of procedure

Next Steps

• Resolve the conflicts between the Siebel application for which the *.sif file was originally prepared and the Siebel application on which the *.sif file is deployed. See the section, "Resolving Siebel Application Data Conflicts".

Resolving Siebel Application Data Conflicts

Some of the archive data that you import can conflict with the current Siebel application data.

Conflicts between the Siebel application for which the *.sif file was originally prepared and the Siebel application on which the *.sif file is deployed may

require resolution. Table 15 displays the Voice Component features with the conflicts that need to be resolved.

Table 15: Features with Conflicts to be Resolved

Feature	Voice Component archive (.sif) file	Conflict with *Gplus* Components	Conflict with Siebel Applications	New Table Fields
Basic Voice	GenComm.sif	No	Yes	No
	GenesysTools.sif	No	No	No
	GenSymbolicStrings.sif	No	No	No
Expert Contact	GenComm_ECS.sif	No	No	No
Outbound Campaign	GenComm_OCS.sif	No	Yes	No
Universal Callback	GenComm_VCB.sif	No	Yes	Yes

Refer to the following Procedure: Voice Component: Resolving the Siebel application data conflicts to help you resolve these data conflicts.

Procedure: Voice Component: Resolving the Siebel application data conflicts

Purpose: To resolve the Siebel application data conflicts that are created during *.sif file import process.

Start of procedure

1. Start first with the GenCom.sif file and then each additional Voice Component feature that you wish to implement. Repeat the conflict resolution steps below, substituting for the GenComm.sif file, the *.sif file associated with that optional Voice Component feature that you implemented. Refer to Table 15 on page 206 to check which Voice Component features have conflicts.

2. Click Next on the Import Wizard–Preview window (see Figure 24 on page 202).
   The Review Conflicts and Actions screen displays.

3. In the Review Conflicts and Actions screen, from the Conflicting Objects panel, navigate to Business Component, and select it so that the list of business components appears.
   - If there are no business components, skip to Step 5.

4. For each business component in the list, select it in the Conflicting Objects panel.

- In the Attribute Differences panel, right-click on each of the following attributes (if they exist) and re-set the resolution to Repository:
    - Class
    - Search Specification
    - Sort Order
    - Sort Specification

Note: A business component does not necessarily have any of these attributes: Class, Search Specification, Sort Order, and Sort Specification.

---

5. In the Review Conflicts and Actions screen, from the Conflicting Objects panel, navigate to Applet, and select it so that the list of applets appears.

- If no applets are listed, skip to Step 7.

6. For each applet in the list with Siebel application data conflicts, do the following:

- In the Attribute Differences panel, right-click on each of the following attributes (if they exist) and re-set the Resolution to Repository:
    - Search Specification
    - Sort Order

7. After importing the GenComm.sif file, lock the Genesys Voice project.

8. After importing the GenesysTools.sif file, lock the Genesys Tools project.

No other projects needs to be locked after importing any of the other *.sif archive files.

9. For each additional Voice Component feature that you wish to implement, repeat the import and conflict resolution steps above; but note the special requirement for the Universal Callback feature below in Step 10.

Implementing the Universal (Voice) Callback Feature

10. If you are implementing the Universal (Voice) Callback feature:

a. Import the corresponding *.sif file

b. Resolve conflicts between Siebel Server applications, as necessary. Use Table 15, "Features with Conflicts to be Resolved," on page 206 to check for potential conflicts.

c. Manually copy the script as follows:

i. From the Object Explorer, go to the Application objects.

ii. From the Applications window, select the application on which the Adapter is deployed—for example, Siebel Universal Agent for Siebel Horizontal versions, or Siebel Financial Services for Siebel Vertical versions.

iii. Right-click the application record and select Edit Server Script from the drop-down menu.

> **iv.** Edit the Application_PreNavigate method for the selected Server
> Script with the code provided in the file <InsDir>/<Siebel
> Version>/Application_PreNavigate_VCB.es. If there are no server
> scripts associated with the Application object, you will be asked to
> select the scripting language. Select eScript in this case. Refer to
> "Updating the Application's Application_PreNavigate Event
> Server Script" on in the Appendix.

11. Compile the updated *.srf file. See the section, "Compiling the Siebel
Repository File" on .

12. Deploy the new *.srf file. See the section, "Deploying the Repository File"
on .

End of procedure

Next Steps

• Compile the Siebel repository file. See the section, "Compiling the Siebel
Repository File" on .

Compiling the Siebel Repository File

Importing a Voice Component archive file (*.sif) effectively modifies the
Siebel Communications Toolbar, so that it provides the standard toolbar
buttons for the Voice Component feature associated with that *.sif file. The
Siebel Communications Toolbar, including these modifications and any
customizations, is not created until the Siebel repository file is compiled.

Note: Each communications toolbar item has its own position specified by
an appropriate field. You must ensure that after importing new toolbar
items there are no conflicts for the position field that appears. Siebel
does not check this, and conflicts may cause issues with the
communications toolbar appearance.

Importing the primary Voice Component file, GenComm.sif, creates the
fundamental Siebel Communications Toolbar. Each of the other Voice
Component archive files that you import modifies the toolbar, as applicable,
for the corresponding Voice Component features.

Note: As with the Voice Component archive files, each of the other G*plus*
Adapter archive (*.sif) files modifies the Siebel Communications
Toolbar, as applicable. That is, each of the archive files associated
with the other G*plus* Adapter components also includes code that
modifies the Siebel Communications Toolbar to provide the buttons
and controls necessary to work with the installed features.

Procedure:
Voice Component: Compiling the Siebel repository file

Purpose: To compile the Siebel repository file.

Start of procedure

1. In Siebel Tools, select Tools > Compile Projects.

2. Select Locked projects.

3. In the Siebel Repository File edit box, select the name of the repository file.

4. Click Compile.
   The status bar at the bottom of the Object Compiler window indicates when the compilation is finished.

5. When the compilation is finished, close Siebel Tools.

Note: A siebel_assert_xxx.txt file may be generated by the Siebel environment after compiling the Siebel repository file. This does not affect the functionality of the component in any way.

End of procedure

Next Steps

- If you are deploying the Universal Callback feature, update the new tables in the Genesys project server database. See the section, "Updating the S_EVT_ACT_X Table in the Genesys Project Server Database" on .

- If you are not deploying the Universal Callback feature, deploy the Siebel repository file. See, "Deploying the Repository File" on .

Updating the S_EVT_ACT_X Table in the Genesys Project Server Database

This section is required only if you are deploying the Universal Callback feature.

Procedure:
Voice Component: Checking-in the Table Activity project changes to the Siebel Server repository

Purpose: To check-in the changes made to the Table Activity project.

Start of procedure

1. In Siebel Tools, select Tools > Check In.

   The Check In dialog box displays.

2. In the Projects list, select the following project:

   Table Activity

3. Click Check In.

End of procedure

Next Steps

- Apply the changes to the Siebel database tables. See, Procedure: Voice Component: Applying changes to the Siebel database tables.

Procedure:
Voice Component: Applying changes to the Siebel database tables

Purpose: To update the database table, S_EVT_ACT_X, on the local Siebel database (or on a Siebel Server database).

Start of procedure

1. Connect Siebel Tools to the server database.

2. In Siebel Tools, in the Object Explorer, navigate to the Table object.

3. Navigate to the List of Tables pane.

4. Locate the S_EVT_ACT_X table.

5. Click Apply. A warning window is displayed.

6. Click OK to accept the warning.

7. In the Apply Schema window, from the Tables drop down list, select Current Row.

8. Enter the correct values for the Database user, Database user password, and ODBC data source fields.

   You must enter the user name and password for a Database user who has administrator's privileges in the Siebel environment.

   See the Siebel documentation for more information about creating custom tables.

9. Click Apply.

   The message Changes successfully applied appears, indicating that the tables were created.

10. Click the Activate button to propagate the database changes and make them available to all users.

End of procedure

Next Steps

• Deploy the Siebel repository file. See the section, "Deploying the Repository File".

Deploying the Repository File

Follow the steps in Procedure: Voice Component: Deploying the repository file to deploy the repository file.

Procedure:
Voice Component: Deploying the repository file

Purpose: To deploy the updated *.srf file.

Start of procedure

1. To deploy the *.srf file, stop the Siebel Server.

2. Copy the new repository file to a specific location within the server installation.

3. Restart the server.

4. Additionally, you may have to generate and deploy browser scripts for the new repository file (see the section, "Scripts" on page 623).

For further information on deploying an updated repository file to the Siebel Server, refer to Siebel documentation.

End of procedure

Next Steps

• Configure the Siebel Call Center application. See the section, "Configuring Siebel Using the Siebel Web Client".

Configuring Siebel Using the Siebel Web Client

This section describes the specific processes required to complete the configuration for each Voice component feature. You must connect to your Siebel Server using the Siebel Web Client.

Note: After deploying the newly compiled Siebel repository file and restarting the Siebel Server, you must log in as a Siebel Administrator for the Siebel Web Client.

Complete the following procedures, described in this section, to configure the Siebel Call Center application on the Siebel Server.

- Update the associated List of Values table.
- Create a customized Communications configuration with the G*plus* Voice driver and profiles.
- Create agents with appropriate rights and permissions.
- For Outbound Campaign, associate campaigns with the proper employees.
- For Outbound Campaign or Voice Callback, you must manually synchronize the Time Zones names in Siebel and Genesys.

Configuring the Siebel Call Center application consists of the following sections:

Updating the Associated List of Values (LOV) Table(s)

This section describes how to import a List of Values table, that is specific for the Voice Component and how to add a new list of reason code values and how to create a new List of Value - Type records for the Voice Component features.

Procedure:
Voice Component: Importing the List of Values table

Purpose: To import the List of Values table for the Voice Component.

Start of procedure

1. Log in as a Siebel administrator.

2. Open the Site Map, then navigate to Administration - Business Service, Simulator.

3. Specify Genesys Tools as the Service Name.

4. Specify ImportAll as the Method Name.

5. Set Iterations to the value of 1.

6. In the Input Arguments applet, click the Load From File button, browse for <InsDir>/<Siebel Version>/GenComm_LOV.xml and load this file as shown in Figure 25.
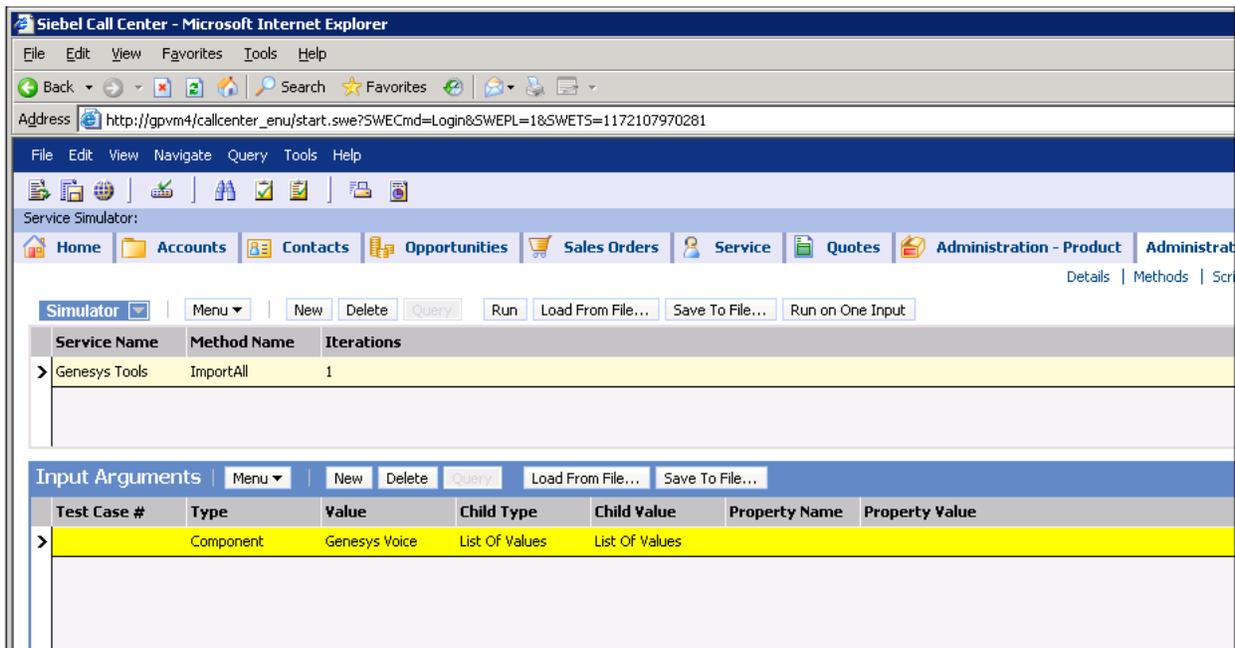
Figure 25: Importing the Lists of Values Table

7. Click Run on the Service Methods applet. The import results can be checked in the Output Arguments applet.

If only the Voice Component is Deployed

8. If only the Voice Component is deployed, but not the Multimedia Component, then create a new List of Values (LOV) and *manually* populate it with the values, as described below:

   - In Administration Data > List of Values, create a new LOV with the following parameters:
     - Display Value Gplus Universal
     - Lang-Indep Code Gplus Universal
     - Type COMM_MEDIA_TYPE
     - Lang Name English-American
     - Parent LIC COMMON

- Order N/A (leave empty)
- Active On
- Translate On
- Replic Level All

End of procedure

Next Steps

- Create a list of the reason code values for the Basic Voice feature. See, Procedure: Voice Component: Creating a list of the reason code values for the Basic Voice feature.

Procedure:
Voice Component: Creating a list of the reason code values for the Basic Voice feature

Purpose: To create a list of the reason code values for the Basic Voice feature.

Start of procedure

Creating a New List of Values - Type Record

1. Navigate to the List of Values - Type table, using the Siebel Tools, choose the Screens application menu, choose System Administration, and then the List of Values menu item.

2. Create a new record in the List of Values - Type applet.

3. Specify the following parameters for the new record:
   - Type: REASON_CODE
   - Translate: Selected
   - Leave the default values for all other fields.

4. Save the new record.

Adding Reason Code Values

5. Add a new item to the List of Values - Type table for each reason code in your contact center.

6. Select the Item Type: REASON_CODE.

7. Provide the following information for this item:
   a. For the Display Value field, provide the string representing the reason code—for example, Reason: Rest Room; Reason: Lunch; Reason: Gone for break, and so on. See the note below.
   b. For the Language Independent field, provide the positive integer value for the reason code—for example, 10, 20, 30, and so on.
   c. For the Language field, select English-American—for other languages, see the Siebel documentation.

    d. Leave the default values for all other fields.

Note: Genesys recommends that in the Display Value field, you use the value Reason: prefix. This value gives an agent a hint to make a choice of the reason code in the drop-down menu that is displayed in the Transfer Multiple LOV Popup Applet applet.

End of procedure

Next Steps

• To create a list of the Genesys call status values for the Expert Contact feature. See the section, Procedure: Voice Component: Creating a list of the Genesys call status values for the Expert Contact feature, on page 215.

Procedure:
Voice Component: Creating a list of the Genesys call status values for the Expert Contact feature

Purpose: To create a list of Genesys call status values for the Expert Contact feature.

Start of procedure

Creating the New List of Values - Type Records

1. Navigate to the List of Values - Type table, using the Siebel Tools, choose the Screens application menu, choose System Administration, and then the List of Values menu item.

2. Create a new record in the List of Values - Type applet.

3. Specify the following parameters for the new record:
   • Type: GENESYS_CALL_STATUS
   • Translate: Selected
   • Leave the default values for all other fields.

4. Save the new record.

5. Create a new record in the List of Values - Type applet.

6. Specify the following parameters for the new record:
   • Type: GENESYS_LOCATION
   • Translate: Selected
   • Leave default values for all other fields.

7. Save the new record.

Specifying the Genesys Call Status Types

8. Create a new record in the List of Values - Type applet.

9. Specify the following parameters for the new record:
   • Type: GENESYS_CALL_STATUS

- Display Value: Active
- Language Independent Code: Active
- Language Name: English-American
- Active: Selected
- Translate: Selected
- Leave the default values for all other fields.

10. Save the new record.

11. Complete steps 8 through 10 for all of the values provided below that are used as the Display Value and the Language Independent Code:
    - Active
    - Dialing
    - Established
    - Held
    - Released
    - Ringing

Specifying Genesys Locations

12. Create a new record in the List of Values - Type applet.

13. Specify the following parameters for the new record:
    - Type: GENESYS_LOCATION
    - Display Value: [Network Switch Name]
    - Language Independent Code: [Network Switch Name]
    - Language Name: English-American
    - Active: Selected
    - Translate: Selected
    - Leave default values for all other fields.

    Note: The values in this procedure are shown only as an example.

14. Save the new record.

End of procedure

Next Steps

- Create a list of the Genesys Routing Point values for the Voice Callback feature. See, Procedure: Voice Component: Creating a list of the Genesys Routing Point values for the Voice Callback feature.

Procedure:
Voice Component: Creating a list of the Genesys Routing Point values for the Voice Callback feature

Purpose: To create a list of the Genesys Routing Point values for the Voice Callback feature.

Start of procedure

Creating the New List of Values - Type Records

1. Navigate to the List of Values - Type table, using the Siebel Tools, choose the Screens application menu, choose System Administration, and then the List of Values menu item.

2. Create a new record in the List of Values - Type applet.

3. Specify the following parameters for the new record:
   - Type: GENESYS_ROUTING_POINT
   - Translate: Selected
   - Leave the default values for all other fields.

4. Save the new record.

Adding the Routing Point Values

5. Create a new record in the List of Values - Type applet.

6. Specify the following parameters for the new record:
   - Type: GENESY_ROUTING_POINT
   - Display Value: [Routing Point Name]
   - Language Independent Code: [Routing Point]
   - Language Name: English-American
   - Active: Selected
   - Translate: Selected
   - Leave the default values for all other fields.

7. Save the new record.

8. Complete steps 5 through 7 for all of the Routing Points in the list of Routing Points that you gathered according to the instructions in the section, "Gathering the Genesys Object Settings" on .

End of procedure

Next Steps

- Create a customized Communication configuration, driver, and profile. See the section, "Creating a Customized Communications Configuration, Drivers, and Profiles".

Creating a Customized Communications Configuration, Drivers, and Profiles

The Communications configuration, drivers, and profiles are all created by importing data from the configuration definition file—GenComm_universal.def. The G*plus* Adapter supplies this file, which is also called the (*.def) file. It includes the profiles, the drivers, and the command and event definitions for the following Gplus Adapter Components:

- G*plus* Voice
- G*plus* Open Media.

Before importing the data from the configuration definition file, the sample configuration parameters defined in this file must be customized to reflect your actual contact center configuration. Then you can import the edited configuration definition file. You can selectively import the following data from the configuration parameters file:

- profiles
- drivers
- commands
- events.

The GenComm_universal.def file comes with the G*plus* Adapter, and it is typically worthwhile to create a copy and familiarize yourself with it.

Overview

Three procedures are involved in creating a Communication configuration:

1. Editing the configuration definition file. This may include editing to create additional profiles for the same media type. See, Procedure: Voice Component: Modifying and preparing the configuration definition file for import, on page 219.

2. Creating the configuration record in Siebel. See, Procedure: Voice Component: Creating a new configuration record in Siebel, on page 221.

3. Importing the configuration. See, Procedure: Voice Component: Importing the new configuration record into Siebel, on page 222.

Note: When the import task is completed, remove any unused profiles or profiles. It is important to specify particular media types to work with if you do not plan to use the full line of G*plus* Components.

Editing the Configuration Definition File

The configuration definition (*.def) file must be edited to:

- Supply the correct values for the required driver's fields.

- Customize the commands and events definitions.
- Make it version-specific for the Outbound Campaign feature.
- Create additional profiles for the same Media Type—for example, a second Voice profile.

Use the appropriate path to access the *.def file:

For Windows   <Voice Component folder>\<Siebel Version>\GenComm_universal.def

For UNIX   <Voice Component folder>/<Siebel Version>/GenComm_universal.def

Procedure:
Voice Component: Modifying and preparing the configuration definition file for import

Purpose: To modify and prepare the configuration definition file for import.

Start of procedure

1. Locate the appropriate definition file and open it for editing.
2. Make a copy of the definition file.

> **Note:** Always make a copy of the definitions file in case you need to return to the original settings. Use the Notepad application for Windows, or a visual (vi) editor for UNIX.

3. You must define the location parameters for each configuration:
   - In the [Driver:] section, modify the ChannelString parameter to specify the media types to be used.
     - If you plan to use only the voice functionality without the Push Preview dialing mode of the Outbound Campaign feature, use "voice" as the only value—for example:

     Channel String = "voice"
     - If you plan to use the Push Preview dialing mode of the Outbound Campaign feature, use "outboundpreview" as the value—for example:

     Channel String = "voice,outboundpreview"

   > **Note:** You must install the G*plus* Multimedia component to use the Push Preview dialing mode of the Outbound Campaign feature.

   - In the [Configuration Parameters] section, define the ConnectString parameter as a premise switch name *exactly* as it is defined in Genesys Configuration Manager for T-Server—for example:

ConnectString = "Prem-1-G3_72"

- In the [Driver Parameters: …] section, define the LocalConnString parameter as a premise switch name *exactly* as it is defined in Genesys Configuration Manager for T-Server—for example:

    Driver:LocalConnString = "Prem-1-G3_72"

4. Locate and modify the following parameters for each profile:

- Driver:TServerAppName—the T-Server application name, as it appears in Genesys Configuration Manager. The corresponding T-Server should be included in the Connections section of the GenCommServer application in Configuration Manager. Parameters of the connection, such as secure connection, should be configured in the GenCommServer application in Configuration Manager, according to the *Genesys 8.0 Security Deployment Guide*.

Note: The Driver:BCTServerAppName parameter must be added in addition to the Driver:TServerAppName parameter, if the current G*plus* Voice Component is deployed on Genesys alternative DR Peer Site in SIP Business Continuity topology (see option 2a under "Possible Deployment Scenarios" on page 48).

This scheme of connection to T-Server was introduced in release version 7.5. To keep backward compatibility with previous releases, it is allowed to leave this parameter unchanged, as in Driver:TServerAppName = "CHANGE_ME", or to remove it from the configuration. In this case, use the previously-allowed set of parameters shown below:

- Driver:ServerHost—the host name of the machine, where T-Server is running (for example, myhost1). You must provide a value for this parameter.
- Driver:ServerPort—the port number for T-Server in decimal form, for example: 5443. You must provide a value for this parameter.
- Driver:BackupServerHost—the host name of the machine where the backup T-Server is running (for example, myhost1bk). This parameter is optional.
- Driver:BackupServerPort—the port number for the backup T-Server in decimal form (for example: 5445). This parameter is optional.
- Driver:SwitchType—numeric value, which defines the type of the switch. For the list of available (supported) values, see "The Voice Component Driver Parameters" on page 278.

That approach, however, does not allow usage of the new security features in Genesys 7.5.

5. For the Expert Contact feature, specify the following parameters:

- Service:HasKWProtocol = TRUE.
    This enables the protocol support.

6. For the Outbound Campaign feature (with its OSC functionality), specify the following parameters:

- Service:HasOCSProtocol = TRUE
  This enables the protocol support.

> **Note:** See, Table 21, "The Voice Component Driver Parameters," on page 279, for more information about other OCS driver parameters.

7. For the Universal Callback feature, specify the following parameters:
   - Service:HasVCBProtocol = TRUE
     This enables the protocol support.
   - Service:SwitchName = <Voice Switch Name>
   - Service:VCBDefaultRoutingPoint = <Routing Point>
   - The Service:VCBProcessedOnCallRelease parameter must be synchronized with the Genesys VCB Server callback_processed configuration option on the appropriate Routing Point or VCB Server instance, if not specified on the Routing Point.
     - If you want the value of the callback_processed configuration option to equal True, then set the value of the VCBProcessedOnCallRelease parameter to equal False.
     - If you want the value of the callback_processed configuration option to equal false, then set the value of the VCBProcessedOnCallRelease parameter to True.
8. Save the definition file.

End of procedure

Next Steps

- To create a new configuration record in Siebel. See Procedure: Voice Component: Creating a new configuration record in Siebel.

Procedure:
Voice Component: Creating a new configuration record in Siebel

Purpose: To create a new configuration record in Siebel.

Start of procedure

1. Select Site Map > Administration - Communications > All Configurations.
   A list of the existing configurations displays.
2. Click New to create a new configuration.

3. Enter a name for the new configuration—for example, Genesys
    Configuration.

4. Click Save.

End of procedure

Next Steps

*   To import the new configuration record into Siebel. See, Procedure: Voice
    Component: Importing the new configuration record into Siebel, on
    page 222.

Procedure:
Voice Component: Importing the new configuration
record into Siebel

Note: Before importing the new configuration record, see the section
"Advanced Customization of Voice Component Configurations" on
page 233 for additional ways to configure this functionality.

Purpose: To import the new configuration record into Siebel.

Start of procedure

1. Select a configuration record you have created—for example, Genesys
    Configuration.

2. Click the Import Configuration button.

3. Click Next.

4. In the Next window, define the configuration file that you want to import,
    and select one, or more, of the four data parts. Your selection depends on
    your requirements. Select all of the following data parts for the initial
    import:
    *   drivers and profiles
    *   commands
    *   events
    *   configuration parameters

5. Use the Browse button to select the G*plus* Adapter configuration file:
    GenComm_universal.def, or a customized version of it.

6. Click OK to start importing.

End of procedure

Next Steps

- Remove the unused profiles and drivers that contain insufficient connection information. See the section, Procedure: Voice Component: Removing Unused Profiles in the Configuration, on page 223.

Procedure:
Voice Component: Removing Unused Profiles in the Configuration

Purpose: To remove the unused profiles and drivers containing insufficient connection information. It is important to remove the unused profiles and drivers, if you do not plan to use the full line of the G*plus* Components.

Note: Before changing the *.def file, always create and save a backup copy.

Start of procedure

1. Select Site Map > Administration - Communication > All Configurations.

   A list of the existing configurations displays.

2. Select a record—for example, Genesys Configuration.

3. Select the Profiles tab.

4. Remove the unused profiles and drivers.

End of procedure

- Connect to the T-Server Application object. See the section, "T-Server Connections".

T-Server Connections

A T-Server connection can be made in two different ways:

- A T-Server connection using the Connections tab of the Gplus Communication Server: In this scenario, the Siebel configuration uses the Driver:TServerAppName parameter to refer to the T-server connection. It is also possible to define advanced connection parameters—for example, security settings.

- A T-Server connection using the Siebel configuration from pre-7.5 versions of the Adapter: If the Driver:TServerAppName parameter was not defined in the Siebel configuration, or if its value is CHANGE_ME, then all of the connection parameters are taken from the Siebel configuration as they existed in the pre-7.5 versions of the Adapter. In this scenario, the security settings cannot be defined.

Defining the T-Server Connection in Genesys Configuration Manager

In this type of configuration, the connection to the T-Server Application object should be added to the list of G*plus* Communication Server connections, and in the Siebel configuration, the Driver:TServerAppName parameter should be configured to specify the T-Server Application object name of the primary T-Server, exactly as it is configured in Configuration Manager.

> **Note:** If the given G*plus* Voice component is a part of SIP Business Continuity deployed at the alternative DR Peer Site, the name of the T-Server Application object should be specified in Driver:BCTServerAppName during Siebel Configuration (see "Configuring Siebel" on).

If the value of the TServerAppName parameter is defined in the Siebel configuration, the Adapter searches the connection parameters in Configuration Manager and uses these parameters to establish a connection with T-Server. If this connection is found, then the T-Server connection parameters (including the advanced parameters) are taken from Configuration Manager, and the AdapterAppName parameter is ignored (the G*plus* Communication Server application name is used instead).

Also, the switch type is taken from the T-Server Application object. If the connection to the related T-Server is not found under the Connections tab of the G*plus* Communication Server Application object, the Adapter will not open a connection to T-Server. The configuration of the G*plus* Communication Server Application object must be updated in Configuration Manager by defining the connection to the related T-Server.

Using this method, it is possible to define a secure connection between the Adapter and T-Server. If secure connection is defined, the Adapter and T-Server will use Transport Layer Security (TLS) when exchanging data. For details on configuration of security in Genesys, please refer to the *Genesys Security Deployment Guide*.

Defining the T-Server Connection in the Siebel Configuration

In this type of configuration, the Driver:TServerAppName parameter is:

• not defined in the Siebel configuration

• or, has the value of CHANGE_ME.

All connection parameters are taken from the Siebel configuration as they existed in the pre-7.5 versions of the Adapter. In this scenario, the security settings cannot be defined.

Agent Administration

This section explains the processes involved in creating agents and agent-related records.

Before using the Voice Component, you must first create the following:

- Agent (see the section, "Creating Agents" on page 225.)
- Positions (see the section, "Creating Positions, Telesets, and ACD Queues" on page 228.)
- Telesets (see the section, "Creating Positions, Telesets, and ACD Queues" on page 228.)
- ACD Queues (see the section, "Creating Positions, Telesets, and ACD Queues" on page 228.)
- Agent Group (see the section, "Outbound Campaign Agent Configuration" on page 231.)

Creating Agents

To create an agent, you must re-define an existing employee (as defined in Siebel) to function as an agent with the G*plus* Adapter.

To be an agent, the employee must:

- Have the correct responsibility and position in regards to the Adapter.
- Be associated with a relevant profile.
- For Outbound Voice and Voice Callback configurations, it is essential to specify the proper Time Zone.

Procedure:
Voice Component: Assigning the correct responsibility and position to an existing employee

Purpose: To assign the correct responsibility and position to an employee in regards to the Adapter.

Start of procedure

1. Select Site Map > Users Administration > Employees.

2. Select an employee who is supposed to function as an agent in regards to the Adapter.

3. Specify the following parameters for the employee record:
   - Responsibility:
     - Click on the field.
     - Click Add new from the drop-down menu.

- Select Universal Agent.
- Click OK.
- Click OK to return to the main screen.
- Position:
  - Click the field.
  - Click New from the drop-down menu.
  - Browse to Telemarketing position.
  - Click OK.
  - Mark the position as Primary.
  - Click OK to return to main screen.
- Time Zone:
  - Click the field.
  - Browse to desired time zone.
  - Click OK.

4. Save the record.

End of procedure

Next Steps

- Associate the agents with an existing Communications profile. See,
  Procedure: Voice Component: Associating agents with an existing
  Communications profile and Genesys settings.

Procedure:
Voice Component: Associating agents with an existing Communications profile and Genesys settings

Purpose: To associate agents with an existing Communications profile and Genesys settings.

Start of procedure

1. In the Siebel Site Map, navigate to Administration - Communications> All Configurations.

2. On the Configurations applet, locate the G*plus* Outbound Contact configuration.

3. Click the applet to make it active.

4. Select the Agents tab.

5. Click New.

6. On the Add Agents screen, browse to and select an agent.

7. Click OK to add the agent into the profile.

8. Save the agent's record.

9. Select Administration - Communications > Agent General Profile.

10. Specify the following values for the record:

    Agent Login: Agent login ID into ACD/PBX from Genesys

    Password: Agent login password into ACD/PBX

11. Save the record.

12. Select the Telesets tab.

13. Click New.

14. Navigate to a particular Teleset on the Add Teleset drop-down menu. (The Teleset value should match the one specified in Genesys.)

15. Select the Teleset to make it active and click OK.

16. Optional step: To select ACD Queues for the agent:

    a. Select the ACD Queues tab.

    b. For each agent, add 0 (zero) or more ACD Queues for the agent to log into.

    c. Select the ACD Queue configured with the Genesys settings, and set the ACD Queue as Primary.

17. Save the record.

End of procedure

Next Steps

*   Associate an agent to a campaign. See, Procedure: Voice Component: Associating an agent to a campaign

Procedure:
Voice Component: Associating an agent to a campaign

Purpose: To associate an agent to a campaign. Associating an agent to a campaign is also known as creating a "campaign agent".

Note: This procedure is only required, if you use the Outbound Campaign feature.

Start of procedure

1. Select Site Map > Administration Users > Employees.

2. Select an employee who is supposed to function as a campaign agent in regards to the Adapter.

3. Click the Responsibility field.

4. On the left side of the pop-up window, select Campaign Agent.

5. Click Add.

6. Click OK to return to the main screen.

End of procedure

Next Steps

- Create a Position record. See, .

For further information on how to create campaign's Contacts and/or Prospects and synchronize this data with Genesys Outbound Contact, refer to the *Gplus Adapter 7.5 for Siebel CRM* documentation.

Creating Positions, Telesets, and ACD Queues

By creating Positions, Telesets, and ACD Queues, you create the data structures that support the agents. If you are using the Outbound Campaign feature, the agents should be associated with the Voice ACD Queues, and the newly created campaigns should be associated with the Agent Group. For more information on agents, campaigns, and groups, see the Siebel documentation.

Procedure:
Voice Component: Creating a Position record

Purpose: To create a new Position record.

Start of procedure

1. Select Site Map > Groups Administration > Positions.

2. Click New on the Positions applet.

3. Specify the following parameters for the new record:
   - Division: select Default Organization
   - Position: Telemarketing

4. Save the record.

End of procedure

Next Steps

- Add an ACD Queue. See, .

Procedure:
Voice Component: Adding an ACD Queue

Purpose: To add an ACD Queue. An agent should be associated with an ACD Queue, which is configured for operations with Outbound Contact Server (OCS).

Start of procedure

1. Navigate to the List of Values (LOV) table by selecting Site Map > Administration-Data > LOV Explorer

2. For each ACD Queue in your contact center, you must add a new item from the list of items.

3. Select the Item Type, CTI_ACD_QUEUES.

4. Provide the following information for this item:
   - For both the Display Value and Language Independent fields, include the number of the DN of type ACD Queue.
   - For the Language field, select English-American (for other languages, see the Siebel documentation).

End of procedure

Next Steps

- Create a Teleset. See, Procedure: Voice Component: Creating Telesets, on page 229.

Procedure:
Voice Component: Creating Telesets

Purpose: To create a Teleset, which is a set of DNs that represent a single workplace.

Start of procedure

1. In the Siebel Site Map, navigate to Administration - Communications> All Telesets.

2. For each Teleset, provide the following information:
   - Create a new Teleset in the Telesets list.
   - Add your agents' Extension.

3. Give the Teleset a unique name—for example, you may want to name the Teleset based on the cubicle number, or the location of the Teleset.

4. Select the Extensions tab.

5. Add all the Teleset DNs in the list (one, or more DNs). For each DN, you must define the DN type:

S—Standard DN

A—ACD Position

6. Follow Steps 2 to 5 above for each Teleset.

Note: If a Teleset has only DNs of type ACD Position, then you must configure at least one of these DNs as a DN of type S in Siebel**.**

End of procedure

Next Steps

- Use Genesys Framework to synchronize the agent information. See, "Using Genesys Framework to Synchronize Agent Information".

Using Genesys Framework to Synchronize Agent Information

After you have created the agents in the Siebel Call Center application and you have checked to see that the agent information appears in the Genesys environment, check the corresponding Persons properties in Genesys Framework. If necessary, use Genesys Framework to make the necessary updates.

For every employee (agent) configured in Siebel to work with the Voice component of the Adapter, there must be an equivalent Person object with the same Agent Login object in the Genesys configuration. The Genesys Configuration Synchronization component of the Adapter should automatically perform this synchronization.

If the Configuration Synchronization Adapter is not deployed, you must manually duplicate the Siebel agents and places (Telesets) in the Genesys configuration.

After creating the Persons and Place objects in the Genesys environment, either automatically or manually, check the Persons properties in Genesys Framework to verify that the corresponding configuration is correct.

You must define a default Place for the newly created Persons object. Use the following procedure to assign a default Place to a Person object.

Note: If no default Place objects are assigned to an agent, you must perform this procedure manually.

Procedure:
Voice Component: Assigning a default Place object to a Person object

Purpose: To assign a default Place to a Persons.

Start of procedure

1. In Configuration Manager, select Environment > Tenant > Persons.

2. Click Persons.

3. Right-click a specific Person object and select Properties from the drop-down menu.

    The Properties dialog box displays.

4. Select the Agent Info tab.

5. Locate the Default Place drop-down menu.

6. Click Browse.

7. Click Places and select the appropriate place for the Person.

8. Click OK to add the Place.

9. Click Apply to submit changes.

10. Click OK to close the Properties dialog box.

11. Repeat Step 3–Step 10 for all agents.

End of procedure

Next Steps

•   Assign an agent to an Agent Group, if the agents are working with the Outbound Campaign functionality. See, Procedure: Voice Component: Assigning an OCS agent to an Agent Group, on .

Outbound Campaign Agent Configuration

There are additional requirements for agents working with the Outbound Campaign functionality. Agents must be assigned to an Agent Group configured to work with Outbound Contact Server (OCS), which is associated with the Outbound Campaign feature. If an agent is not assigned to a group, you should manually perform this task, as explained in the following procedure.

Procedure:
Voice Component: Assigning an OCS agent to an Agent Group

Purpose: To assign OCS agents to an Agent Group.

Start of procedure

1. In Configuration Manager, select Environment > Tenant > Agent Groups.
2. Right-click a particular Agent Group configured to work with OCS.
3. Select New > Shortcut from the drop-down menu to access an agent.
4. Navigate to the agent entry with which you are working.
5. Click OK to add the agent.
6. Repeat Steps 2–5 for all agents that must be configured to work with OCS.

End of procedure

Next Steps

•   If you are working with the Universal Callback component, see,

•   If you not working with the Universal Callback component, then there are no further steps. You have completed the configuration and installation of the Outbound Campaign feature of the G*plus* Adapter 8.0 for Siebel CRM

    The configuration steps mentioned above are related to the synchronization of information between Siebel and Genesys. For more detailed information about the configuration of the Genesys Outbound Contact Server product, refer to the Genesys' Outbound Contact documentation.

Universal Callback Agent Configuration

All agents working with Universal Callback must be assigned to an Agent group configured to work with the Universal Callback server. If an agent is not assigned to a Agent Group, you should manually perform the following procedure.

Procedure:
Voice Component: Assigning an Universal Callback agent to an Agent Group

Purpose: To assign Universal Callback agents to an Agent Group.

Start of procedure

1. In Configuration Manager, select Environment > Tenant > Agent Groups.

2. Right-click a particular Agent Group configured to work with Universal Callback.

3. Select New > Shortcut from the drop-down menu to access an agent.

4. Navigate to the agent entry with which you are working.

5. Click OK to add the agent.

6. Repeat Step 2–Step 5 for all agents that must be configured to work with Universal Callback.

End of procedure

Next Steps

*   You have competed the configuration and installation of the Universal Callback configuration of the G*plus* Voice Component for Siebel CRM.

Advanced Customization of Voice Component Configurations

This section provide information and instructions for the advanced customization of the Voice Component.

These advanced customization topics are described in the following sections:

Note: Before attempting to make any modifications or customizations, review the basic deployment procedures for any information that may apply to the kind of changes you are making—for example, if your modification requires changes to the configuration (*.def) file, refer to the section, "Editing the Configuration Definition File" on page 218. The information in this section assumes that you understand the context of the implementation procedures and any associated requirements or constraints that may apply.

Creating Activity Records

Each incoming and outgoing interaction initiates the creation of an Activity record in the Siebel application and also, creates a new work item in the Communications (CTI) Toolbar. Many customization projects involve working with these created Activity records. The Activity record creation and the error-handling logic model are implemented in the Siebel *.def file by using the standard methods. See the *Siebel Communications Server Administrator Guide* for more information on this topic.

Warning!   Creating an Activity record affects the performance of the Siebel Server. If you are not using Activity records, remove the Activity Creation code from the event handlers. See the section, "Modifying the Definition File and Resolving Any Related Performance Issues" on page 238, for more information.

This section consists of the following sections:

Creating a Basic Voice Feature Activity Record

Activity records for inbound and outbound calls are created in the EventEstablished event. The Activity record creation is handled by the following two event handlers:

- [EventHandler:OutboundCallReceived] for outbound calls
- [EventHandler:InboundCallReceived] for inbound calls.

The log creation logic model and the set of information fields used are defined in the appropriate event response; the post-call release actions are defined in the same event response as the After Call Work (ACW) elements.

Inbound Calls   For an inbound call, the [EventHandler:InboundCallReceived] event handler creates an activity for an inbound call in the EventEstablished event and displays the contact information on the agent's dashboard. The [EventHandler:InboundCallReceived] event handler then queries the Contact Business Component record and depending on the number of rows returned by the query

specification, 'Work Phone #' LIKE '*{OtherDN}', it executes the following three different log event handlers:

- Multiple contacts

- Single contact

- No contacts found.

Outbound Calls   For an outbound call, the [EventHandler:OutboundCallReceived] event handler creates an activity in the [EventHandler:EventEstablished] event handler and displays the contact information on the agent's dashboard. The event handler queries the Contact Business Component record and depending on the number of rows returned by the query specification, 'Work Phone #' LIKE '*{OtherDN}', it executes the following three different log event handlers:

- Multiple contacts

- Single contact

- No contacts found.

The Display = "true" option must be commented out or removed from the code, if you want to enable the Contact popup screen.

Creating an Expert Contact Feature Activity Record

Activity record creation scenarios are different, depending upon whether the CTI-Less T-Server Preview-interaction mode is turned on or off, as explained below:

Scenario 1: The Preview-interaction mode is turned off   If the Preview-interaction mode is turned off:

? The Activity records for the inbound and outbound calls are created in the same manner as for the Basic Voice feature. See"Creating a Basic Voice Feature Activity Record" on .

Scenario 2: The Preview-interaction mode is turned on   If the Preview-interaction mode is turned on:

- An Activity record is created in the EventKwPreviewIntRequest event with a Requested status.

- The established call reuses both the same work item in the Communication toolbar and the same Activity record.

- The log creation logic model and the set of information fields used are defined in the following associated event response: OnEventKwPreviewIntRequest.

- The post-call release actions are defined in the same event response as the After Call Work (ACW) elements.

Table 16 displays the name of supported event handlers and the status property that they set.

Table 16: Supported Event Handlers for the Expert Contact Feature

Event Handler	Status
EventKwPreviewIntRequest	Requested
All work items releasing events. See, "Creating a Basic Voice Feature Activity Record" on page 234 for more information.	Done

Creating an Outbound Campaign Feature Activity Record

Activity records are created for all incoming Outbound Contact Server interactions. The activity creation logic models for the Preview records and the scheduled calls are defined in the following event responses:

- OnOCSRecord

- OnEventPushPreviewInvited.

The activity creation logic models for the Predictive and Progressive calling modes are implemented in the OnPredictOCSRecord event response and the [EventHandler:OutboundOCSCallEstablished] event handler.

The [EventHandler:PreviewOCSCallEstablished] event handler prevents the duplication of Activity records created for any preview calls. The Preview records (as well as the scheduled calls and the predictive calls) are incoming interactions and they create their own activity record.

Depending on an agent's response, the appropriate status is set in the Activity record. Table 17 displays the name of supported event handlers and the status property that they set.

Table 17: Supported Event Handlers for Outbound Contact Server

Event Handler	Status
EventPreviewRecord EventScheduledCall EventEstablished (for Predictive and Progressive Dialing modes) EventUpdCallComplStatsAck OpenMediaOcsCallCompletionSAck	In Progress
EventRecordRescheduleAck OpenMediaOcsRecordReschedulAck	Scheduled
EventDoNotCallAck EventRecordRejectAck OpenMediaOcsRecordDoNotCallAck OpenMediaQueued	Declined
EventRecordCancelAck EventRecordCancel EventRecordRemove OpenMediaOcsRecordCancelAck OpenMediaOcsRecordRemovedAck	Canceled
EventRecordProcessedAck OpenMediaOcsRecordProcessedAck	Completed
EventReleased	Unassigned

Creating a Universal Callback Feature Activity Record

The [EventCallBackRequest] event handler creates an activity for callback requests in the EventCallBackRequest event and displays the contact information on the agent's dashboard. The event handler queries the Contact Business Component record and then, depending on the number of rows returned by the query specifications, executes the following three different log event handlers:

- Multiple contacts

- Single contact

- No contacts found.

Also, the event handlers shown in Table 18 update the Activity records (created in the [EventHandler:EventCallBackRequest] event handler) and must be

commented out or removed from the code, if you want to disable the creation of the Activity records for the CallBackRequest request.

Table 18 shows a list of supported event handlers and the status that they set.

Table 18: Supported Event Handlers for Universal Callback

Event Handler	Status
EventCallBackRequest	Requested
EventCallBackAccepted	In-Progress
EventCallBackRejected	Declined
EventCallBackCancel	Canceled
EventCallBackDone	Done
EventVCBPreview	Requested
EventAddVCBRequest	Scheduled
EventVCBReject	Declined
EventVCBCancel	Canceled
EventVCBProcessed	Done

Modifying the Definition File and Resolving Any Related Performance Issues

Creating Activity records affects the overall Siebel Server performance. Genesys recommends that you review the requirements for creating Activity records during the implementation phase. Make sure to comment out or to remove the Activity Creation code that you are not using from the GenComm_universal.def file.

For your convenience, the Activity Creation code is marked as follows:

Start:
Lines below are activity creation code <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

End:
Lines above are activity creation code >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

The Siebel Server performance is also affected when you query the Campaign Business Component record for the Siebel SmartScript. If you are not using the Siebel SmartScript, the following code, which is related to the Siebel SmartScript handling, must also be commented out or removed from the GenComm_universal.def file.

For your convenience, the SmartScript-related code is marked as follows:

Start:

; Lines below are Smart Script related code <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

End:

; Lines above are Smart Script related code >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

Transfer Functionality Restrictions

Outbound Contact Server

The transfer functionality for agent-to-agent transfers using Outbound Contact Server (OCS) is limited by the following restrictions:

- Only two-step and mute (blind) transfers are allowed from agent-to-agent. Genesys recommends that when making an agent-to-agent transfer, the first agent should make a consultation call to the second agent before making the transfer. If the agent receiving the transfer has a current call and record on their desktop, the transfer may not be successful.

- The user data should be attached to the call after the agent completes a transfer.

- A call transfer is possible to and from agents and also, to agents logged into a loaded or active campaign group. You can also transfer to agents who are not participating in the active campaign. This option requires that you use Outbound Contact 7.x, or higher.

- Genesys recommends that the number of transferred outbound calls does not exceed 5 to 10 percent of all outbound calls.

Expert Contact Feature

Because of the constraints imposed by CTI-Less T-Servers, the transfer functionality for the Expert Contact feature is limited by the following restrictions:

- Mute (blind) transfers are supported only as an external reroute to the Network T-Server for GenSpec.

- Two-step transfers are supported for both internal CTI-Less T-Server operations and for external transfers between CTI-Less T-Servers.

- The maximum allowed number of active calls is two.

Transferring Outbound Calls to Agents Who are Not in a Campaign

The Voice Component supports Outbound Preview, Predictive, Progressive, and Push Preview campaign transfer calls to agents who are not participating in an active campaign. This feature requires Genesys Outbound Contact, version 7.x, or higher. In this type of transfer scenario, an agent, who is not in the active campaign, can do the following:

- reschedule a record as a campaign callback

- set a new call status

- update the number of dialled attempts

- request a chain of records
- cancel a record
- mark a contact as Do Not Call (DNC)
- process a record

Voice Component Common Configuration Information

This section describes the common configuration information that applies to all Voice Component features.

Switch-Specific Configuration Instructions

You must make the following configuration changes for the following switches:

- the Alcatel A4400/OXE switch
- the Nortel Communication Server 2000/2100 switch
- the Rockwell Spectrum switch.

Alcatel A4400/OXE   The Adapter supports the operation of the T-Server for Alcatel 4400/OXE and the Alcatel A4400/OXE switch with the Agent Substitute feature set to either of the following:

- on or true
- off or false

In either of these two modes, the set of DNs for the agent Teleset in the configuration must have the following defined:

- DNs of type Extension, Teleset DN of type S (Standard DN)
- DNs of type ACD Position, Teleset DNs of type A (ACD Position DN)

When you use the Agent Substitute feature set to on or true, you must define the following two parameters in the CTI configuration:

- Service:AgentSubstitute = "TRUE"
- Service:ACDDNList = "{@ACDDNList }"

If you use the Agent Substitute feature set to off or false, you must define the following two parameters in the CTI configuration:

- Service:AgentSubstitute = "False"
- Service:ACDDNList = "{@AgentID}"

Nortel Communication Server 2000/2100   In order to allow agents to make outbound calls using the Siebel Communications toolbar, the Nortel Communication Server 2000/2100 (previously known as Nortel DMS-100) switch must be configured to make outbound calls from DNs of both types, ACD Positions and Extensions.

Nortel Communication Server 1000   When using the Nortel Communication Server 1000 with SCCS/MLS (previously known as Nortel Symposium or Nortel Meridian) switch, the

T-Server option, nrdy-after-login, must be set in accordance with your switch configuration, otherwise T-Server does not send the proper notifications to its clients regarding the Ready/NotReady state of the agents.

Configuring the Communication DN

The Communication DN is used to send and broadcast an EventUserEvent message.

Sending EventUserEvent Messages through a Communication DN

The Siebel CRM configuration for G*plu*s Adapter can define a special DN, which is called the Communication DN. Third-party applications (for example, T-Server clients) can communicate with the G*plus* Adapter using this Communication DN to send EventUserEvent messages with an attached UserData attribute to all agents currently logged into the Adapter. These third-party applications are defined as any client application other than the G*plus* Voice Component.

The Communication DN for this feature is configured in Siebel by defining the following communications driver parameter:

Driver:CommunicationDN <DN>

The value of this parameter must be equal to a DN that is configured in Configuration Server as a Communication DN for the same T-Server and switch.

In a Communication DN configuration, an EventUserEvent message is distributed by T-Server to all Siebel CRM agent desktops that have the same Communication DN configuration as the third-party application.

The data flow for the Communication DN configuration is described in the following scenario:

1. Whenever any third-party application registered to the Communication DN sends an EventUserEvent message with an attached UserData attribute to this Communication DN, T-Server distributes this EventUserEvent message with the same attached UserData attribute

2. This EventUserEvent message is then sent through the Communication DN and distributed to all Siebel CRM agent desktops.

3. The EventUserEvent message is then sent to Siebel, if the corresponding event handler with the EventUserEvent device event is defined in the Siebel Communications configuration.

Figure 26 displays the data flow for this configuration.
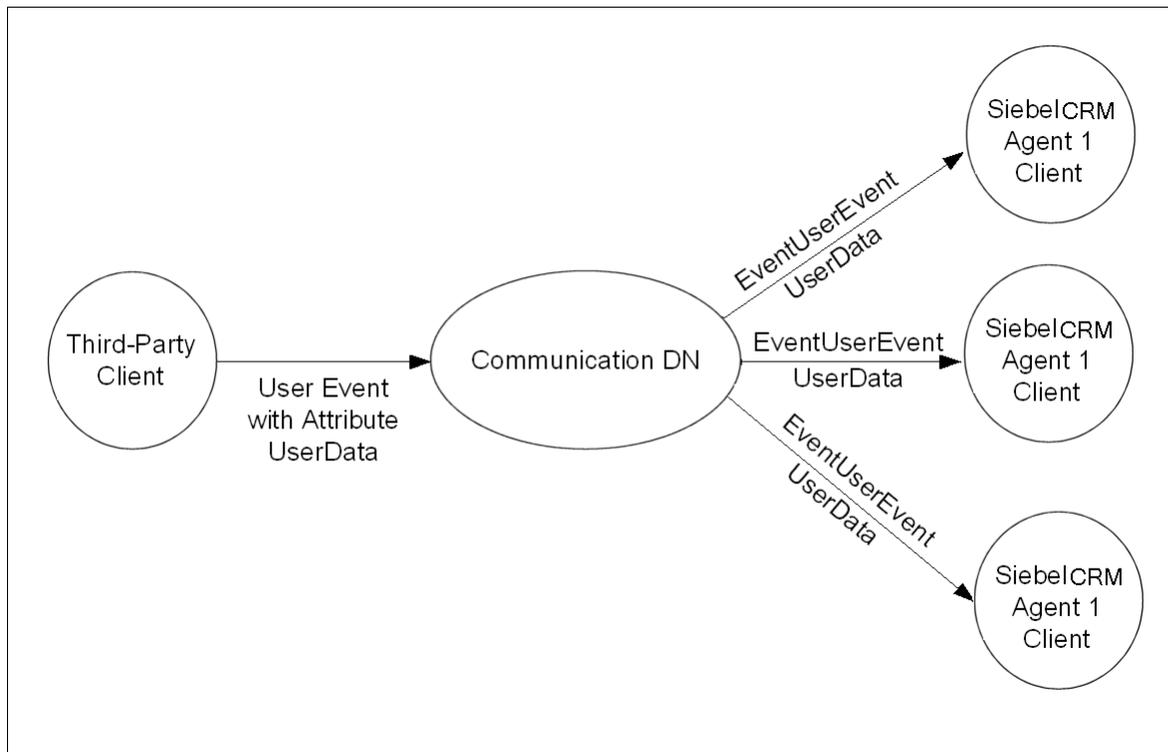
Figure 26: Data Flow of an EventUserEvent Message Sent Through a Communication DN

Broadcasting EventUserEvent Messages Through a Communication DN

You can generate an EventUserEvent message (with an attached UserData attribute) with the Voice Component and broadcast it to both third-party applications, as well as all of the agents logged into the Siebel CRM G*plus* Voice Component with the same Communication DN configuration. Figure 27 displays the data flow of broadcasting EventUserEvent messages.

Note: You can also define additional parameters with the SendUserEventToCommDN command in the Voice Component.
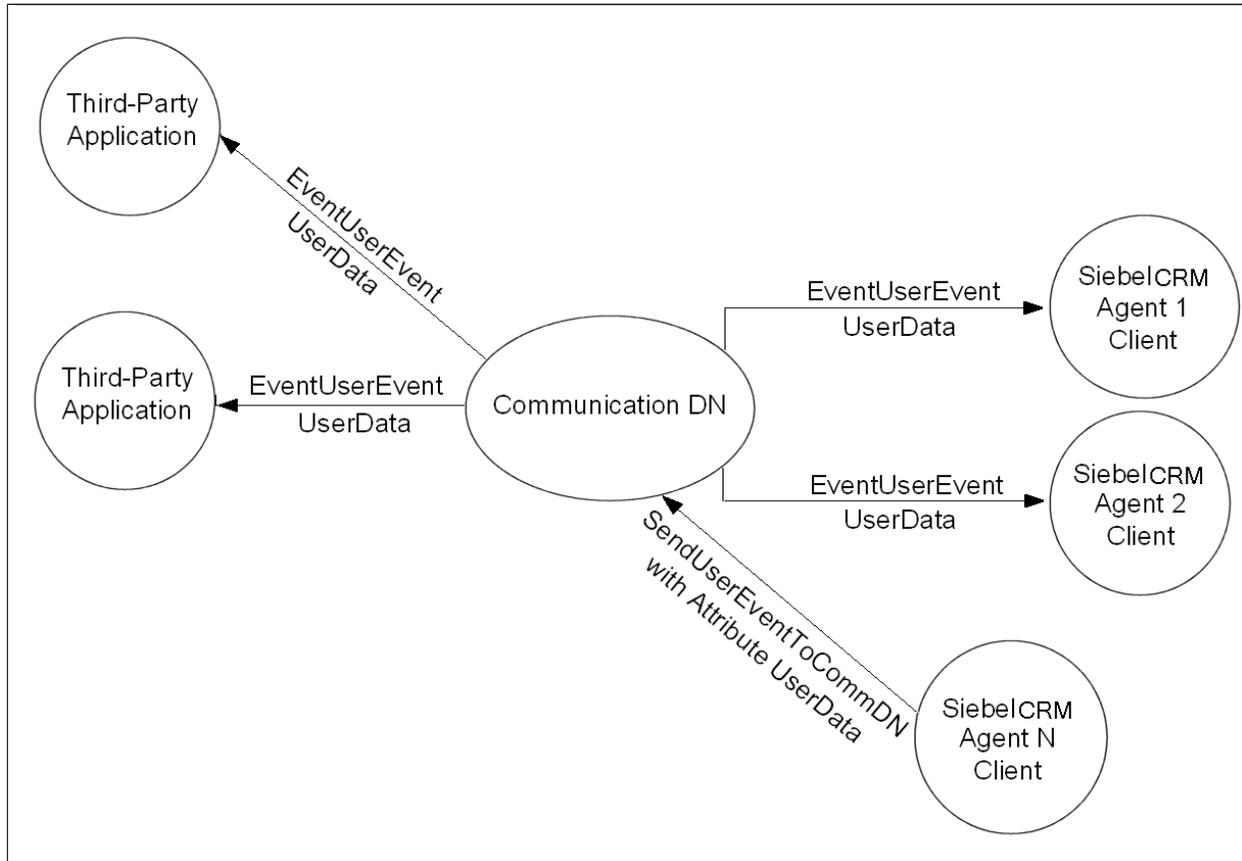
Figure 27: Data Flow for Broadcasting an EventUserEvent Message through a Communication DN

The Communication DN for this feature is configured in Siebel by defining the following communications driver parameter:

[Driver:BroadcastCommdnUserEvents2Agents]

The value of this parameter must be set to true.

Siebel can send the EventUserEvent message with an attached UserData attribute by using the SendUserEventToCommDN command.

This command is defined in the sample GenComm_universal.def file described in the following section, "Editing the Configuration Definition File" on page 218. If the [Driver:BroadcastCommdnUserEvents2Agents] driver parameter is not defined, or if its value is set to false, then the EventUserEvent message is delivered by T-Server to the third-party applications only. See, Figure 28.
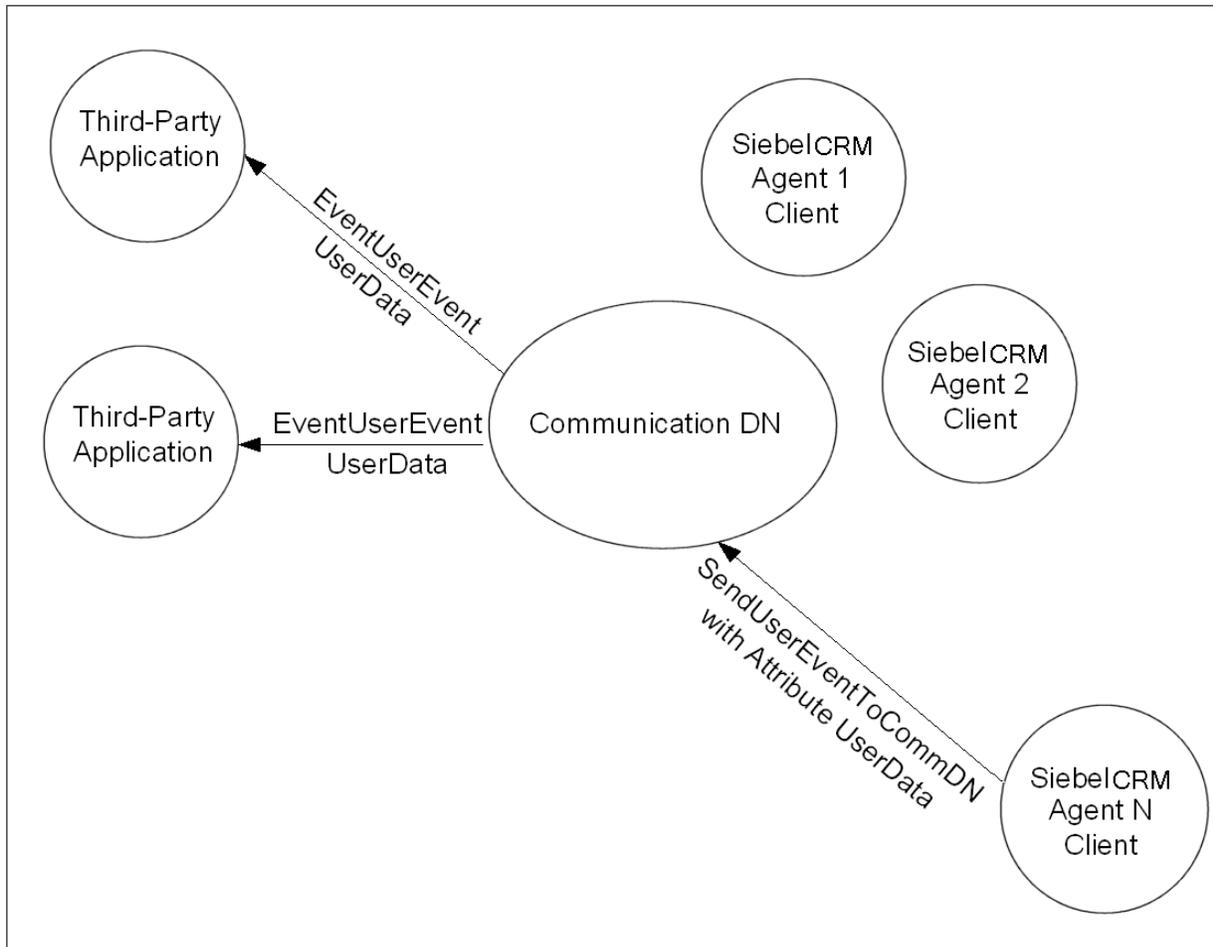
Figure 28: Data Flow for Broadcasting User Events to Third-Party Applications Only

After you configure the Communication DN and set the
[Driver:BroadcastCommdnUserEvents2Agents] driver parameter to true, then you
must define the driver command in Siebel CRM that is associated with the
SendUserEventOnCommDN device command. The GenComm_universal.def file
contains a sample of the SendUserEventToCommDN configuration command.

The SendUserEventOnCommDN device command is used in the
GenComm_universal.def file to define the [Command:SendUserEventToCommDN]
parameter and the corresponding [CmdData:ConnDataForSendUserEventToCommDN]
parameter.

The device command must be defined in order to distribute the EventUserEvent
message with an attached UserData attribute to the Communication DN. For
example, in the GenComm_universal.def file, the Siebel CRM
[Command:SendUserEventToCommDN] command parameter is configured to
invoke the Adapter SendUserEventOnCommDN device command with the
following attached UserData attribute key-value pair list:

- AdditionParam1/This is Param1
- AdditionParam2/This is Param2

In the GenComm.def file, this configuration is shown in Example 1 below:

Example 1   [Command:SendUserEventToCommDN]

      Description = "Send UserData To Comm DN"

      DeviceCommand = "SendUserEventOnCommDN"

      Hidden = "true"

      Profile = " "

      Comments = " "

      CmdData = "ConnDataForSendUserEventToCommDN"

    [CmdData:ConnDataForSendUserEventToCommDN]

      Param.AdditionParam1 = "This is Param1"

      Param.AdditionParam2 = "This is Param2"

      Comments = " "

When you import the above definitions into the configuration, Siebel can use the SendUserEventToCommDN command parameter to deliver the EventUserEvent message (with the following attached UserData attribute as the key-value pair list: AdditionParam1/This is Param1, AdditionParam2/This is Param2) to other applications that have been registered to receive the same Communication DN events.

The AttributeExtensions Parameter

The AttributeExtensions parameter allows you to send an Extensions attribute with a request to T-Server. In this scenario, you define the AttributeExtensions parameter in the Siebel Communications Server command data. To define this parameter, two additional parameters must be defined as shown in Table 19.

Table 19: Additional Parameters for the AttributeExtensions Parameter

Parameter	Value
AttributeExtensions	<NameOfAttributeExtensionsValue>
NameOfAttributeExtensionsValue	<ValueOfExtension> where the value of the extension is a number

Example 2 describes how you can define the MakeCallToPhone command with the AttributeExtensions parameter in the GenComm.def file.

Example 2

[Command:MakeCallToPhone]

  DeviceCommand = "MakeCall"

  Description   = "Make Call to "{@Phone}""

  CmdData   = "MakeCallToPhone"

  OnControl = "TRUE"

  Hidden   = "TRUE"

  [CmdData:MakeCallToPhone]

    AttachContext= "TRUE"

    RequiredField.@Phone= "?*"

    Param.PhoneNumber= "{@Phone}"

    Param.CallNotifyText= "Call from {@UserName}..."

    Param.AttributeExtensions   = "GCTI_PARAMETER"

    Param.GCTI_PARAMETER= "111"

In this example, the Param.AttributeExtensions parameter defines the name of the key-value, "GCTI_PARAMETER", and the Param.GCTI_PARAMETER parameter defines the value of the key-value, "111".

As a result, whenever Siebel calls the MakeCallToPhone command, the Adapter sends the request to T-Server with the following Extensions attribute key-value pair:

GCTI_PARAMETER/111.

Sending Extensions Attributes with String and/or Integer Value Types (Method 1)

The G*plus* Adapter can send Extensions attributes with multiple extensions of string and/or integer value types in requests to T-Server.

Extensions Attribute With a String Value Type

To send the Extensions attribute with a multiple extensions of a string value type, the Siebel configuration must use command data that includes the StringAttributeExtensions parameter and its associated parameters.

The StringAttributeExtensions parameter defines the list of parameters that are used as Extensions attributes in requests to T-Server. The names and values of the associated parameters are used as the keys and values of the Extensions attributes included in the requests to T-Server. If a parameter that is included in the StringAttributeExtensions list does not exist, this parameter is ignored by the Adapter.

In the StringAttributeExtensions list, you should separate each parameter by a comma, "," and optional spaces (that is, spaces or tabs) as shown in Example 3 on page 247.

Example 3   [CmdData:CmdDataExample]

Param.StringAttributeExtensions = "str1 , Str2 ,String3,string4"

Param.str1 = "String_str1"

Param.Str2 = "String_Str2"

Param.String3 = "String_String3"

Param.string4= "String_String4"

Example 4 shows a command that includes the ReasonCode key with a string value of 7623 in the Extensions attribute of a request to T-Server:

Example 4   ; ////////////Not Ready command

[Command:NotReadyForGVoice]

FilterSpec = "[$GetCommandStatus(NotReady)] = 'Enabled'"

Hidden = "TRUE"

DeviceCommand = "NotReady"

Description = "Set not ready for Voice"

Profile = "Gplus Universal Profile"

CmdData = "NotReadyWithReasonCode"

[CmdData: NotReadyWithReasonCode]

Param.ReasonCode = "7623"

Param.StringAttributeExtensions = "ReasonCode"

Extensions Attribute Using the Value Type Integer   To send the Extensions attribute with multiple extensions of a integer value type, the Siebel configuration must use command data that includes the IntAttributeExtensions parameter and its associated parameters.

The IntAttributeExtensions parameter defines the list of parameters which are used as Extensions attributes in requests to T-Server. The names and values of the associated parameters are used as the keys and values of the Extensions attribute included in the requests to T-Server. If a parameter that is included in the IntAttributeExtensions list does not exist, then a default value of -1 is used.

In the IntAttributeExtensions list, you should separate each parameter by a comma, "," and optional spaces (that is, spaces or tabs) as shown in Example 5:

Example 5   ; ////////////SendDTM command

[Command:SendDTMFComm]

  DeviceCommand = "SendDTMF"

; "Description" is provided by driver

  CmdData   = "DTMFData"

  Hidden   = "TRUE"

  Profile   = "Gplus Universal Profile"

  CmdChannelOnFocus = "TRUE"

[CmdData:DTMFData]

```
Param.Digits    = "12345"
Param.AttributeExtensions = "GCTI_SendDTMF"
Param.GCTI_SendDTMF= "111"
Param.IntAttributeExtensions = "ToneDuration, Int2,PauseDuration"
Param.ToneDuration = "12"
Param.Int2 = "222"
Param.PauseDuration = "8"
```

The length of the StringAttributeExtensions or IntAttributeExtensions parameters is defined by the length of the Siebel field, Param.*, and can be a maximum of 251 characters. The number of associated parameters in the list cannot be more than 40. The rest of the parameters are ignored.

Both types of Extensions attributes can be used in the same request. The earlier approach to define the Extensions attributes for a single integer value was kept (see the description of "The AttributeExtensions Parameter" on page 245). It can be used for backward compatibility (see Param.AttributeExtensions and the corresponding Param.GCTI_SendDTMF in Example 6 below).

In addition to this, two parameters in a command data define the names of integer and string extensions which are going to be included as Attribute Extensions in the request to T-Server as shown in Example 6:

Example 6
```
[Command:SendDTMFComm]
  DeviceCommand = "SendDTMF"
; "Description" is provided by driver
  CmdData     = "DTMFData"
  Hidden      = "TRUE"
  Profile     = "Gplus Universal Profile"
  CmdChannelOnFocus = "TRUE"
[CmdData:DTMFData]
    Param.Digits    = "12345"
    Param.AttributeExtensions = "GCTI_SendDTMF"
    Param.GCTI_SendDTMF= "111"
    Param.IntAttributeExtensions = "ToneDuration, Int2,PauseDuration"
    Param.ToneDuration = "12"
    Param.Int2 = "222"
    Param.PauseDuration = "8"
    Param.StringAttributeExtensions = "str1 , Str2 ,String3,String4"
    Param.str1 = "String_str1"
    Param.Str2 = "String_Str2"
    Param.String3 = "String_String3"
    Param.String4 = "String_String4"
```

Sending Extensions Attributes with String and/or Integer Value Types (Method 2)

Where requests to T-Server could be sent with either attached UserData attributes, Reasons attributes or Extensions attributes, the G*plus* Adapter is able to attach integer and/or string type values in these parameters when invoking the corresponding device commands.

These parameters are defined in a [CmdData:…] section of the corresponding DeviceCommand.

The names of the parameters that must be included as Reasons attributes are provided by the Reasons parameter—for example:

Param.Reasons = "reason_i1, reason_str1,reasons_i2"

The names of the parameters that must be included as Extensions attributes are provided by the Extensions parameter Extensions—for example:

Param.Extensions = "Int2, Str4, Str1, reason_i1"

The way to define integer and/or string type values with Param.IntAttributeExtensions and Param.StringAttributeExtensions command parameters, as described earlier in this section, can also be used to define the Extensions attributes.

The rest of the CmdData parameters that have the prefix Param are treated as attached UserData attributes.

How to Differentiate Integer Type Values from String Type Values

The backslash symbol ("\") in the first position of the parameter value is used to denote an integer type value. Any other symbol in the first position denotes that this symbol is a string type value. If the string type value needs to have a backslash in its first position, then use a double backslash as follows: \\StringValue.

If the value of a parameter, as defined in the Siebel Communications Server command data, with a Param.* prefix begins with a single "\" (backslash), then this type value is treated as an integer. A single backslash is considered as a string type value and its value is single backslash ("\").

In the following examples:

Param.res-Bslash = "\" in [CmdData:NotReadyWithPopup]

is included in Reasons attribute as the string value type: res-Bslash   \ .

If the value of a parameter is defined by "\" as an integer value type, but cannot be converted to an integer value type, then the parameter value is 0 (zero) as shown in Example 7 on page 250.

Example 7   Param.reasons_i2 = "\i453"

will have the value 0.

[Command:MakeCallToPhone]
  Description   = "Make Call to "{@Phone}""
    DeviceCommand = "MakeCall"
    FilterSpec   = "NOT [$GetCommandStatus(IsInteractionView)] = 'Enabled'"
  Hidden   = "TRUE"
  OnEditControl = "TRUE"
  Comments   = ""
  Profile   = "Gplus Universal Profile"
  CmdData   = "MakeCallToPhone"

[CmdData:MakeCallToPhone]
    Param.CallNotifyText   = "Call from {@UserName}..."
    Param.Extensions   = "Int2, Str4, Str1, reason_i1"
    Param.Int1   = "\9832"
    Param.Int2   = "\12678"
    Param.IntAttributeExtensions = "Int1, Int2,Str1, reasons_i2, Str4"
       Param.PhoneNumber   = "{@Phone:PhoneTypeLookup}"
    Param.Reasons   = "reason_i1, reason_str1,reasons_i2"
     Param.Str1   = "\\Str With 1 backslash"
    Param.Str2   = "_Without Backslashes"
    Param.Str4   = "\\OneMoreBack"
    Param.StringAttributeExtensions = "Int1,Int2,Str1"
    Param.reason_i1   = "\8645"
    Param.reason_str1   = "qwerty"
    Param.reasons_i2   = "\i453"
    Param.str3   = "again w\ithout"
    RequiredField.@Phone   = "?*"
    Comments   = ""

[Command:NotReadyForGVoiceGroup]
  Description   = "Set not ready for Voice"
  DeviceCommand = "NotReady"
  FilterSpec   = "[$GetCommandStatus(NotReady)] = 'Enabled'"
  Hidden   = "TRUE"
  Comments   = ""
  Profile   = "Gplus Universal Profile"

  CmdData   = "NotReadyWithPopup"
[CmdData:NotReadyWithPopup]
    Param.ReasonCode = "[Name]"
    Param.ReasonValue = "[Value]"

```
      Param.Reasons    = "Res1_int, Res3_str,  Res2_str,res-Bslash,  res5"
       Param.Res1_int   = "\1237"
      Param.Res2_str    = "//r2_str"
      Param.Res3_str    = "r3_str"
      Param.res-Bslash  = "\"
      Param.res5        = "\\with\\baslash"
      SelectApplet      = "Value Type Pick Applet"
      SelectBusComp     = "List Of Values"
      SelectBusObj      = "List Of Values"
      SelectParam       = "TRUE"
      SelectQuerySpec   = "[Type] = 'REASON_CODE' AND [Active] = 'Y'"
       SelectTitle = "Please select the reason for changing status to Not-Ready"
       Comments         = ""
```

How to Invoke the UpdateUserData Device Command

This section provides an example of the definition of device commands that can be used for call attached data modification, as shown in Example 8:

Example 8

```
;//////////////////////////////////////////////;// "TestAttachData" would be called from
EventResponse
;//of some EventHandler:
;// for test purposes only

[Command:TestAttachData]
Description   = "Attach UserData To call"
; DeviceCommand = "AttachData"
; DeviceCommand = "DeleteUserData"
DeviceCommand = "UpdateUserData"
   Hidden      = "true"
;// Please define real Profile how it is defined in real ;//configuration instead of  "Gplus
Universal Profile"
Profile      = "Gplus Universal Profile"
Comments     = ""
CmdData      = "AssociateAttachData"

[CmdData:AssociateAttachData]
Param.AdditionParam1 = "This is Param1"
Param.AdditionParam2 = "This is Param2"
Param.AdditionParam3 = "This is Param3"
Comments             = ""
```

For testing purposes, you may create a new button on the Siebel Communications toolbar and associate this button with the TestAttachData

command as defined above. For detailed information about how to create this button on the Siebel Communications toolbar, see the Siebel documentation.

The following information is important to remember when working with data attached to a call:

- For incoming calls, the Adapter enables the following device commands after the EventEstablished event is sent:

  ◆ AttachData

  ◆ UpdateUserData

  ◆ DeleteUserData

  For outgoing calls, the same device commands are available after the EventDialing event is sent

- The AttachData device command should not be used to update any already existing data attached to a call. In this scenario, the command simply adds the specified parameters, which can lead to duplicated entries in the data attached to the original call. The UpdateUserData device command creates a new entry, or updates an existing one. For more information, see the T-Server documentation related to the switch you are using.

- You may also use other device commands (MakeCall, TransferMute, TransferInit, and TransferStep) to attach data to a call by defining additional parameters for these commands. All additional parameters for these commands are considered as user data that has to be attached to a call.

Network Attended Transfer/Conference Feature

This release supports the Network Attended Transfer/Conference (NAT/C) feature, which is designed to enable agents working in multi-site contact centers to consult with each other before making call transfers or conferences, regardless of whether both agents work at the same of different sites. It also enables the agent who requests the consultation to maintain their conversation with the customer while the system is looking for an available agent and setting up the consultation call.

In this scenario, a network call request is relayed to the premise T-Server (Premise A), and then this request is transferred to another premise T-Server (Premise B) by using the NAT/C requests sent from Premise A. The network call request is initially routed to Premise A, which is considered as the starting point for the network transfer on Premise A.

When the original calling party has established a connection with an agent (Agent 1) on Premise A, the network multi-party functionality can be invoked. In this scenario, Agent 1 can make a request for a consultation with another agent (Agent 2) on Premise B. The NAT/C request is initiated on Premise A with the required parameter of the remote location. When this request is answered on Premise B, the consultation call leg is created on the network.

After the consultation call leg is initiated, Agent 1 is then able to swap voice connections between Agent 2 and the original calling party. Agent 2 can also

swap the voice connection, but there is a limitation, which is that the voice connection can only be swapped in one of the following ways:

- Between Agent 1 and Agent 2.
- Between Agent 1 and the original calling party.

After the consultation call leg is initiated, an agent can do the following:

- Complete the transfer of the network call to a remote location (Premise B).
- Reconnect the network call to Premise A.
- Merge the network call into an existing conference call.

The NAT/C request can be initiated when the TransferInit or ConferenceInit device commands are invoked. The remote location information is described by the Param.RemoteConnectStr command data parameter.

The value of the Param.RemoteConnectStr command data parameter defines the name of the premise switch where the destination DN resides. This premise switch name is the name of the remote contact center and must be defined exactly as it is in Genesys Configuration Manager.

Two other Siebel Configuration parameters define the name of the local premise switch where the initial DN resides (this name is the T-Server switch, defined in the current Siebel configuration). These Siebel Configuration parameters are as follows:

- ConnectString in the [Configuration Parameters] section
- Driver:LocalConnString in the [Driver: …] section.

Both of these parameters are treated the same and must be defined as follows:

- In the [Configuration Parameters] section, define the ConnectString parameter as a premise switch name *exactly* as it is defined in Genesys Configuration Manager for T-Server—for example:

  ConnectString = "Prem-1-G3_72"

- In the [Driver: …] section, define the ConnectString parameter as a premise switch name *exactly* as it is defined in Genesys Configuration Manager for T-Server—for example:

  Driver:LocalConnString = "Prem-1-G3_72"

If the RemoteConnectStr parameter of the TransferInit, ConferenceInit, and LocalConnString device commands define the different switches (remote contact centers), and there is a network call on the DN, the Adapter tries to initiate a network attended transfer when one of these device commands is invoked.

If the RemoteConnectStr parameter and the LocalConnString device command are the same, the Adapter initiates a local consultation call on the same switch.

To invoke this functionality with a received network call from an originating call party, the Adapter uses the approach recommended in the *Siebel Communications Server Administration Guide*.

If the ConnectString parameter was defined in the Siebel configuration, then the Siebel macro, $RemoteConnectStr2, allows the retrieval of the name of the remote contact center from the Employee Business Component record.

As described in the *Siebel Communications Server Administration Guide*:

"...$RemoteConnectStr2. The name of the remote call center. This macro, which can be used with transfers and conference calls between call centers, derives from the name of a remote call center's communications configuration from either:

• The ConnectString configuration parameter (if defined), or

• The employee ID of the agent to be called..."

In a sample GenComm_universal.def file, the [CmdData:…] section of the following commands defines the value of the RemoteConnectStr parameter as a $RemoteConnectStr2 macro (see an example below):

• Command:ConsultativeTransferToEmployee

• Command:ConsultativeTransferToPopupEmployee

• Command:ConferenceTransferToEmployee

• Command:ConferenceTransferToPopupEmployee

When a two-step transfer or a conference call is initiated, the Adapter compares the name of the contact center of the initiator as defined by the Driver:LocalConnString parameter with the value of the RemoteConnectStr parameter. If these values are not equal and are not 0 (zero), and there is a network call on the initiating DN, then the Adapter initiates a network call transfer to the remote contact center defined by the value of the RemoteConnectStr parameter. Otherwise, a local consultation call is initiated.

The same Communication toolbar buttons that initiate a local consultation call, can also invoke the Adapter to complete the network call transfer, to reconnect the network call, or to merge the network call into a conference call. If the active work item is a network call, and the call state enables the related feature, then the network call, a reconnected network call, or a merged network call is invoked.

Because there is only *one* call on the premise switch/T-Server in a Network Attended Transfer scenario, there is only one work item in Siebel. The network alternate is invoked by clicking the NtwkAlternateConsultCall button on the Communications toolbar when the call state enables this operation.

Example 9 shows how to define the RemoteConnectStr2 parameter in the command device data, which can be found in the sample GenComm_universal.def file provided with the Adapter:

Example 9   [Command:ConsultativeTransferToPopupEmployee]

     DeviceCommand   = "TransferInit"

     Description   = "Consultative transfer to employee selected from popup list"

     Title   = "Consultative Transfer"

     CmdData   = "ConsultativeTransferToPopupEmployee"

```
    Hidden        = "TRUE"
  CmdChannelOnFocus = "TRUE"
    Profile       = "Gplus Universal Profile"
[CmdData:ConsultativeTransferToPopupEmployee]
    AttachContext     = "TRUE"
    SelectParam       = "TRUE"
    SelectBusObj      = "Employee"
    SelectBusComp     = "Employee"
    SelectApplet      = "ACD Transfer Call Applet"
    SelectTitle       = "Begin Consultative Transfer to:"
    Param.PhoneNumber    = "[Phone #:Lookup]"
    Param.CallNotifyText = "Consultative transfer from {@UserName}..."
    Param.TrackingID     = "{@SelectedWorkItem:DriverWorkTrackID}"
    Param.RemoteConnectStr    = "[$RemoteConnectStr2(Id)]"
```

Note: The $RemoteConnectStr2 Siebel macro returns the non-null name of a remote contact center when an agent is logged into Siebel. When a destination agent is not logged into Siebel, or is in the same Siebel configuration that the initiator of the transfer or conference call is, the Adapter initiates a local consultation call.

Network Attended Transfer/Conference with Optional Automatic Reconnection

When a network transfer TNetworkDestState request is initiated by the agent, but the consultation call has failed returning a NetworkDestState event with the NetworkDestStateFailed attribute with a Busy or NoAnswer cause, then the network call should be reconnected back to the agent.

This reconnection can be accomplished in one of the following two ways:

*   Manually, by clicking the Resume work item button on the Siebel Communications toolbar, or

*   Automatically, by configuring a profile in Siebel to perform an automatic reconnection of a network call after a specified timeout has elapsed.

The Driver:NtwkReconnectTimeout configuration parameter set to a default value of 0 defines two things: whether or not an automatic reconnection is configured and the value for the timeout.

If the value of the Driver:NtwkReconnectTimeout parameter is greater than 0 (zero), then the value represents the timeout (in seconds) before the Adapter invokes a network reconnect request (TNetworkReconnect) to the T-Server. In this case, if an agent attempts to perform a manual reconnection for the network call from the Siebel Communications toolbar during the timeout, the automatic reconnection is not invoked.

If the value of the Driver:NtwkReconnectTimeout parameter is 0 (zero - the default), then the Adapter does *not* automatically invoke a TNetworkReconnect request. The agent will need to invoke a manual reconnection as described above.

Customizations for the Basic Voice Feature

Network Attended Transfer Support

The premise T-Server uses a TSingleStepTransfer request to route a call to the network T-Server. So, the G*plus* Adapter uses a TSingleStepTransfer request when it is performing a network transfer for all switches.

To ensure that the Adapter uses the TransferSStep device command for network transfers, the value of the Service:useSStepTransferForNetworkTransfer parameter must be set to true.

When performing a network transfer, the TransferSStep device command must be configured, so that the value of the RemoteConnectStr device command parameter defines the name of the network switch, as shown below in Example 10 on page 250.

Example 10
```
[Command: SingleStepTransferCallToNetworkTS]
  DeviceCommand    = "TransferSStep"
  CmdData = " SingleStepTransferCallToNetworkTS "
  Hidden         = "TRUE"
  CmdChannelOnFocus = "TRUE"
  Profile        = "Profile Name Here"

  [CmdData: SingleStepTransferCallToNetworkTS]
   Param.RemoteConnectStr = "<AttributeLocation_Value>"
   Param.TrackingID =
   "{@SelectedWorkItem:DriverWorkTrackID}"
```

where <AttributeLocation_Value> is the name of the network switch as defined in the Genesys configuration.

Customizations for the Expert Contact Feature

Configuring the Set of User Data Available to an Agent

The user data attached to a call is stored in the Description field of the Activity record. The required data is defined in the GenComm_universal.def file in the EventKwPreviewIntRequest event handler. A query searches the list of contacts using the following specification: Work Phone #' LIKE '*{ThisDN}. Depending on the query results, the log creation logic model executes one of the following types of log handlers: SingleLog, MultipleLog, or Log (in scenarios where no contact is found) and displays the appropriate Siebel view to the agent.

The set of fields that are available to the agent should be specified in the following line within each log handler:

LogField.'Description' = "Preview interaction request to {ThisDN}"

Customizations for the Universal Callback Feature

There are no customization for this release.

Customizations for the Outbound Campaign or OCS Feature

The Outbound Campaign feature is associated with the Genesys Outbound Contact Server (OCS) product. Refer to the Genesys Outbound Contact documentation for additional details about this product.

Logout Control for the Predictive Dialing Mode

To schedule agent logout time and to prevent abandoned calls in Predictive Dialing mode, use the AgentLogOutControl option in the GenComm_universal.def file. The default value of this configuration option is false. If you set this configuration option to true, you will enable the agent state control (logout) functionality. This functionality notifies agents requesting a logout, the time left until their actual logout. Upon reaching this time, the agent is automatically logged out from communication services. See the Genesys Outbound Contact documentation for more information about this feature. Also for more information, see "Parameter Name: Service:AgentLogOutControl" on . This functionality is supported only by Genesys Outbound Contact version 7.x.

OCS Server Timeout Requests

The G*plus* Adapter controls the response time from Outbound Contact Server (OCS). When the requested timeout is over, it erases the obsolete records from the agent's desktop. The timeout period is specified in the RequestTimeout option in the GenComm_universal.def file. The default value for this option is 30 (in milliseconds)—for example, Service:RequestTimeOut = "30".

Possible use cases include:

- Configurations with one OCS, when the primary OCS is down, or not responding;
- Configurations with primary and backup OCS's, both running in the Warm Standby mode—for example, when an agent has records from the primary OCS, but the backup OCS's role is switched to primary due to the failure of the primary server.

OCS SmartScript Code Customization—Event Response and Event Handler Information

The handling and processing logic in the SmartScript code has changed. The SmartScript ID is not delivered as the user data key-value pair, CRM_SMART_SCR_ID. Instead, as it is extracted from the Campaign Business Component record. The Adapter assumes that the default SmartScript code is used for handling all campaign records.

The SmartScript-related code is included in the following event responses:

- [EventResponse:OnOCSRecord]—this event handler works for OCS records received on the following user events

    PreviewRecord—(for the Preview dialing mode)

    ScheduledCall—(for scheduled records)

- [EventResponse:OnPredictOCSRecord]—this event handler works for OCS records received on the following user events:

    OutboundOCSCallEstablished—(for both Predictive/Progressive dialing modes)

    OCSPartyChanged—(for transferred calls)

Both event responses perform the following operations:

1. Queries the Campaign Business Component record.

2. Stores the SmartScript ID and the language code in the work-tracking object.

3. Stores the campaign information in the custom field of the work-tracking object.

4. Creates an Activity record.

See the GenComm_universal.def file for implementation details. The following placeholders mark the start of the SmartScript related code:

    ; Lines below are SmartScript related code <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

The following placeholders mark the end of the SmartScript related code:

    ; Lines above are SmartScript related code >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

If you are not using the SmartScript functionality, the related code should be removed from the following event responses: [EventResponse:OnOCSRecord] and the [EventResponse:OnPredictOCSRecord]. Also, the [EventHandler:EventCurrentSmartScriptWorkItemChanged] event handler should be commented out or removed from the universal.def file.

For an example of how to use the SmartScript code in the EventCurrentSmartScriptWorkItemChanged event handler, see "Supported Event Handlers for the Expert Contact Feature" on and see the GenComm_universal.def file for details about the current implementation. This event handler is intended for SmartScript execution when an agent switches between work items on the Siebel Toolbar. If you are using the SmartScript

code for screen pops, this event handler should be commented out. Also, the appropriate SmartScript code should be developed and specified for use with a campaign.

Specifying the Do Not Call Command Attribute Keys

There are different ways to apply the Do Not Call (DND) command. The purpose of using the Do Not Call command is to modify a record so that the associated phone numbers are not called. There are different ways to modify the Do Not Call command attributes, so that it refers to either phone numbers or customer IDs and applies the Do Not Call command based on the specified attribute keys.

The Do Not Call command can be modified by specifying the following attribute keys: USE_RECORD_HANDLE, USE_PHONE, USE_CUSTOMER_ID, and GSW_CHAIN_ATTR.

The default value for the GSW_CHAIN_ATTR attribute key is defined in the GenComm_universal.def file by the GService:OCSDoNotCallScope parameter with the value set to AllChain.

The options for the Do Not Call command can modify the scope of the Do Not Call request. You can specify one of the following attribute keys:

- USE_RECORD_HANDLE

- USE_PHONE

- USE_CUSTOMER_ID

The following attribute keys: USE_RECORD_HANDLE, USE_PHONE, and USE_CUSTOMER_ID correspond to the appropriate use cases from the Outbound Contact Server Desktop Protocol definitions, and are described below:

USE_RECORD_HANDLE   The G*plus* Adapter sends a request to the Outbound Contact Server to mark a record as Do Not Call based on the record handle ID. The following additional attribute keys: GSW_CHAIN_ATTR = "RecordOnly" or "AllChain" can be specified to mark a chain or a record as Do Not Call. The default value for the USE_RECORD_HANDLE attribute key is specified in the Service: OCSDoNotCallScope = "AllChain" service parameter, and is effective for all agents under a specific profile. This results in a specific phone number from a record (or all phone numbers from a chain) being marked as Do Not Call and is then included in the Do Not Call list.

USE_PHONE   The G*plus* Adapter sends a request to the Outbound Contact Server to mark a record as Do Not Call based on the phone number. As a result, the phone number is included in the Do Not Call list and records, and this phone number is never called again. See the Outbound Contact Server documentation for more information.

USE_CUSTOMER_ID   The G*plus* Adapter sends a request to the Outbound Contact Server to mark a record as Do Not Call based on the customer ID. As a result, the customer ID

is included in the Do Not Call list and records, and this customer ID is never called again. See the Outbound Contact Server documentation for more information.

Setting the value of the Service:OCSEnableCustomerId service parameter to false disables the use of the contact ID for compatibility with releases earlier than 6.5.2 of Outbound Contact Server. If you set this service parameter to false and specify the USE_CUSTOMER_ID attribute key in the Do Not Call command, then the USE_RECORD_HANDLE use case is used instead. The default value of the Service:OCSEnableCustomerId service parameter is true.

If No Attribute Key isSpecified If none of the following attribute keys, USE_RECORD_HANDLE, USE_PHONE, or USE_CUSTOMER_ID, are specified, then the G*plus* Adapter assumes that the Adapter customer manually provided all of the necessary attributes keys, specified in the command description, and without modifying the Siebel *.def file, sent them to Outbound Contact Server.

See the Outbound Contact Server Desktop Protocol Do Not Call command definition on page 341 for more information about the Do Not Call key attributes.

For example, in the sample GenComm_universal.def file (as shown in Example 11 on page 261):

- The Service:OCSDoNotCallScope service parameter defines the default value of the Param.GSW_CHAIN_ATTR parameter for the Do Not Call device command.

- The GSW_CHAIN_ATTR attribute key is sent with the Do Not Call request in a scenario, where the value of the Param.USE_RECORD_HANDLE parameter is equal to "TRUE", is defined for the Do Not Call device command.

- The Do Not Call device command can be invoked with the GSW_CHAIN_ATTR attribute key values AllChain or RecordOnly, if this command is defined in the CmdData section of the command, and the value of the Param.USE_RECORD_HANDLE parameter equals "TRUE".

- In a scenario where the value of the Param.USE_RECORD_HANDLE equals "TRUE", but GSW_CHAIN_ATTR attribute key is not defined, the default value, AllChain, is used.

Example 11

```
;   DoNotCall
[Command:DoNotCall]
  Description   = "Mark phone number as Do Not Call"
  DeviceCommand = "DoNotCall"
  Hidden        = "TRUE"
  Profile       = "Gplus Universal Profile"
  Comments      = ""
  CmdData       = "DoNotCall"
  CmdChannelOnFocus = "TRUE"


[CmdData:DoNotCall]
;; Param.USE_CUSTOMER_ID    = "TRUE"
   Param.GSW_CHAIN_ATTR     = "AllChain"
;; Param.GSW_CHAIN_ATTR     = "RecordOnly"
;; Param.USE_PHONE          = "TRUE"
   Param.USE_RECORD_HANDLE  = "TRUE"
   Comments                 = "Do Not Call"
```

Specifying Cancel Command Attributes

The OCSCancelScope configuration option for the OCSRequestRecordCancel command, as it is defined in the sample GenComm_universal.def file, is used to modify the scope of a cancel request. The OCSRequestRecordCancel command behavior can be modified by specifying the following Send attribute keys: USE_PHONE, USE_RECORD_HANDLE, and GSW_CHAIN_ATTR. If the GSW_CHAIN_ATT attribute is not specified in the *.def file, then the Adapter uses the default value, AllChain. In the configuration, the default value, OCSCancelScope, in the GenComm_universal.def file is AllChain. Or, you can change the value to RecordOnly, which cancels one record based on the cancel command. For more information, see the Outbound Contact Server documentation.

The USE_RECORD_HANDLE and USE_PHONE attribute keys correspond to the appropriate use cases described in the following Outbound Contact Server Desktop Protocol definitions:

**USE_RECORD_
HANDLE
Use Case**

The G*plus* Adapter sends a request to the Outbound Contact Server to mark a record as Cancel based on the record handle. As a result, the phone number is marked Cancel in the calling list, and is not called for this campaign. See the Outbound Contact documentation for more information. An additional attribute GSW_CHAIN_ATTR with the values of "RecordOnly" or "AllChain" can be specified to mark all records in a chain, or mark a single record as Do Not Call. The default value for the USE_RECORD_HANDLE attribute is specified in the OCSCancelScope service parameter and is effective for all agents under a specific profile. The resulting record (or the whole chain) is marked as Canceled and is not called within a particular calling list.

USE_PHONE Use Case

The G*plus* Adapter sends a request to the Outbound Contact Server to mark a record as Cancel based on a contact's phone number. As a result, the phone number is marked as Cancel in the calling list and is not called for this campaign. See Outbound Contact documentation for more information.

If neither of the USE_RECORD_HANDLE or USE_PHONE attributes are specified, then the G*plus* Adapter assumes that the Adapter customer manually provided all of the necessary attributes keys, specified in the command description, and without modifying the Siebel *.def file, sent them to Outbound Contact Server.

See the Outbound Contact Server Desktop Protocol Cancel command definition on page 340 for information about the RequestRecordCancel key attributes.

For example, in the sample GenComm_universal.def file (as shown in Example 12 on page 261):

- The Service:OCSCancelScope service parameter defines the default value of the Param.GSW_CHAIN_ATTR parameter for the CancelInteraction device command.

- The GSW_CHAIN_ATTR attribute key is sent with the Cancel request in a scenario, where the value of the Param.USE_RECORD_HANDLE parameter is equal to "TRUE", is defined for the CancelInteraction device command.

- The [Command:OCSRequestRecordCancel_] device command defines the request to cancel the OCS record.

- The CancelInteraction device command can be invoked with the GSW_CHAIN_ATTR attribute key values AllChain or RecordOnly, if this command is defined in the CmdData section of the command, and the value of the Param.USE_RECORD_HANDLE parameter equals "TRUE".

- In the case when USE_RECORD_HANDLE = "TRUE" but GSW_CHAIN_ATTR is not defined, the default value AllChain is used.

- The Do Not Call device command can be invoked with the GSW_CHAIN_ATTR attribute key values AllChain or RecordOnly, if this command is defined in the CmdData section of the command, and the value of the Param.USE_RECORD_HANDLE parameter equals "TRUE".

- In a scenario where the value of the Param.USE_RECORD_HANDLE equals "TRUE", but GSW_CHAIN_ATTR attribute key is not defined, the default value, AllChain, is used.

Example 12   [Command:OCSRequestRecordCancel_]
  Description   = "Cancel Interaction"
  DeviceCommand = "CancelInteraction"
  BusComp   = "Campaign List Contact"
  AllViews   = "FALSE"
View   = "Campaign Detail - Position"
  Hidden   = "TRUE"
  Profile   = "Gplus Universal Profile"
  CmdData   = "OCSRequestRecordCancel_"
  CmdChannelOnFocus = "TRUE"

[CmdData:OCSRequestRecordCancel_]
  Param.USE_RECORD_HANDLE = "TRUE"
; Param.USE_PHONE   = "TRUE"
  Comments = "Request Record Cancel - Cancel Current Work Item"
  Param.GSW_CHAIN_ATTR   = "AllChain"
  or
  Param.GSW_CHAIN_ATTR   = "RecordOnly"

Disabling the Record Rescheduling Beyond a Campaign's Expiration Date

By default, Siebel CRM allows for record rescheduling beyond the campaign's expiration date. Genesys Outbound Contact can restrict record rescheduling by hours. Additionally, Outbound Contact prevents a record from being rescheduled in the past, which is also supported by default. To add more sophisticated rescheduling time and date control algorithms, you must customize the Siebel CRM application.

You can implement extended functionality in the Campaign List Contact business component in the PreSetFieldValue server script function. A sample script is shown below (see the topic, "PreSetFieldValue Server Script Function – Example" on).

(For further information about programming and customizing Siebel CRM, see *Siebel Communications Server Guide, Siebel Tools Guide,* and *Siebel E-Script Manual.*)

PreSetFieldValue Server Script Function – Example

In this example, Time_Is_Valid_Campaign_Date (FieldValue) is a customer defined function, which returns TRUE if a particular date is between the campaign Start and End dates.

Example 13
function BusComp_PreSetFieldValue (FieldName, FieldValue)
{
/*
  BusComp_PreSetFieldValue (Campaign List Contact):

```
                Checks the name of the field being updated to see if is one
                of the fields that should be synchronized with the Genesys
                Outbound Framework. If so, the function calls UpdateGenesysOCS( )
                to attempt to connect to Genesys and change the value on the
                Genesys server. If a Genesys error occurs (for example, if Genesys
                cannot be contacted or does not recognize the field name) then
                this function returns CancelOperation so that the value of the field
                remains the same, signaling to the user that a problem has occurred.
                Otherwise ContinueOperation is returned and the update proceeds
             normally.

                Generally no error message will be shown if there is a problem
                contacting Genesys, but the value of the field cannot be updated in
                this case.
    */
        var retVal = ContinueOperation;
        var ViewName = TheApplication().ActiveViewName();
        if ( (ViewName != "Campaign Detail" && ViewName != "Campaign Detail -
    Position" ))
            {
                return (ContinueOperation);
            }

        // Debug code to show calls to this function in the status bar
        // CallCommunicationsCommand("ShowStatusText", "Text", FieldName + " set to
    " + FieldValue);

      switch(FieldName) {
      case "Dial Attempts":
          if(gDialAttempts == null)
              gDialAttempts = GetFieldValue("Dial Attempts");
        break;
      case "Status":
          if(gStatusValue == null)
              gStatusValue = GetGenesysStatusCode("Status");

        break;

    case "Reschedule Time":
            if ( Time_Is_Valid_Campaign_Date( FieldValue ) )
            {
                retVal = CallCommunicationsCommand("RecordRescheduleWithArg",
                "RSTime", FieldValue);
            }
            break;
      case "CallCompleted":
              if (FieldValue == "Y") {
                  //retVal = CallCommunicationsCommand("RecordProcessed", "", "");
              }
              break;
        }
```

```
if (retVal != ContinueOperation)
{
        DisplayGenesysError();
}

return (retVal);
}
```

Selecting an Alternative Destination Phone Number

It is possible to use an alternative destination phone number when making a call to the campaign contact. By default, the G*plus* Outbound Contact feature places a call to the work phone number specified in the Siebel campaign contact. Setting the IgnorePhoneNumber = TRUE key-value pair in the command definitions file (GenComm_universal.def) overwrites the default behavior and forces the use of a Genesys outbound campaign phone number as the destination. The prefix to reach the outside phone network can be specified as the value of the OutsideCallPrefix key—see Example 14:

Example 14
```
;;//////////////////////////////////////////////////////////////////
; MakeCall
; CRM_CAMP_CON_ID - Campaign List Contact
; IgnorePhoneNumber - if set to true, ignores PhoneNumber value and calls number
; from Genesys campaign, otherwise calls PhoneNumber.
; OutsideCallPrefixæPrefix to reach outside phone network; if specified will be
; added to Genesys campaign phone number.
[Command:MakeCallToCampaignContact]
Description = "Make call to campaign contact"
DeviceCommand = "MakeCall"
Hidden = "TRUE"
CmdData = "MakeCallToCampaignContact"
[CmdData:MakeCallToCampaignContact]
BusComp = "Campaign List Contact"
Param.CRM_CAMP_CON_ID = "{Id}"
Param.PhoneNumber = "{Work Phone #:Lookup}"
Param.IgnorePhoneNumber = "TRUE"
Param.OutsideCallPrefix = ""
RequiredField.'Work Phone #' = "?*"
```

Outbound Contact Multiple Campaigns Agents

In Outbound Contact, agents can be shared among multiple campaigns.

When running a campaign in Predictive or Progressive modes, for every agent assigned to more than one active CampaignGroup DBID, Outbound Contact designates only *one* CampaignGroup for this agent. Outbound Contact Server changes the CampaignGroups designation depending on the configured

campaign priorities, the agent activity history, the campaign statistics, and notifies an agent about the new assignment by sending an EventCampaignGroupAssigned event. For further details, see the *Genesys Outbound Contact Solution Deployment Guide*.

Agents in Siebel can work in multiple campaigns.

When running a campaign in Predictive or Progressive modes, the EventCampaignGroupAssigned event notifies an agent of an assignment to a new campaign.

The status text area on the Siebel desktop is displayed for a short time and is used for the agent notification that a new campaign was assigned to the agent.

The campaign name is displayed on the Siebel popup screen and displays the campaign information and contact.

For campaigns running in Preview mode, an agent receives the records by clicking the Record Request button on the Communications toolbar, and the PreviewRecordRequest device command is sent to OCS. When the PreviewRecordRequest device command is invoked, an agent gets the record from the first campaign, which was reported by T-Server to the Adapter as being loaded and started in Outbound Contact Server in the Preview mode.

For campaigns running in Push Preview mode, an agent automatically receives a record. The agent must accept the record to start working with it.

Invoking the SetActiveCampaign device command can change the current Preview mode campaign. Subsequent PreviewRecordRequest device commands retrieve the preview records from the new campaign. The graphical user interface (GUI) for Preview mode is not changed in this release.

If the current Preview mode campaign is unloaded, the Adapter sets as the current (active) campaign, the first Preview mode campaign found in the Adapter internal storage which has the state marked as loaded.

After an agent logs into Siebel when one or more campaigns was already started in Outbound Contact, or after the Adapter reconnects to T-Server, the current state of each campaign is reported when the agent logs into the queue(s) associated with the corresponding Agent Group(s). For activation of the Outbound Contact functionality in this scenario, Genesys recommends invoking the login (or logout, then login) command from the Communications toolbar.

Configuring the Adapter to Support Hoteling (Free Seating)

Hoteling (or free seating) allows agents to log in using any workplace and operate with the DNs associated with that place. This feature is especially useful for hardware phones.

Siebel supports this feature, as you can associate a particular host with a particular teleset. Also, there is a special macro within the CTI configuration that can select the correct DN based on agent login name.

However, there is a functional limitation in that a universal CTI configuration cannot be used in the event hoteling is both on and off. Therefore, you must use a special CTI configuration when hoteling is on.

Note: Only one voice extension on a teleset is supported if the Hoteling feature is enabled. Push Preview is not supported. For more information, please refer to the Siebel documentation.

Procedure:
Voice Component: Configuring the hoteling feature

Purpose: To configure the hoteling feature.

Start of procedure

1. Configure Siebel telesets to enable hoteling. For details, refer to the Siebel documentation.

2. Update the CTI configuration by replacing the 'Phone #' macro with the '$HotelingPhone(Login Name)' macro for all commands that perform Call or Transfer to Employee.
   Also, you must update the corresponding RequiredField and SelectQuerySpec parameter values. Specifically, modify the following commands in GenComm_universal.def:

Table 20: Commands to Modify for Hoteling

Hoteling is OFF (out-of-the-box)	Hoteling is ON
[Command:MakeCallToEmployee] ... [CmdData:MakeCallToEmployee] ... RequiredField.'Phone #'="?*" Param.PhoneNumber = "{Phone #:Lookup}" ...	[Command:MakeCallToEmployee] ... [CmdData:MakeCallToEmployee] ... RequiredField.'Login Name' = "?*" Param.PhoneNumber = "{$HotelingPhone(Login Name):Lookup}" ...
[Command:MakeCallToPopupEmployee] ... [CmdData:MakeCallToPopupEmployee] ... Param.PhoneNumber = "[Phone #:Lookup]" ...	[Command:MakeCallToPopupEmployee] ... [CmdData:MakeCallToPopupEmployee] ... Param.PhoneNumber = "[$HotelingPhone(Login Name):Lookup]" ...
[Command:MakeCallToPopupEmployeeNoContext] ... [CmdData:MakeCallToPopupEmployeeNoContext] ... Param.PhoneNumber = "[Phone #:Lookup]" ...	[Command:MakeCallToEmployeeNoContext] ... [CmdData:MakeCallToPopupEmployeeNoContext] ... Param.PhoneNumber = "[$HotelingPhone(Login Name):Lookup]" ...
[Command:BlindTransferCallToEmployeeVoice] ... [CmdData:BlindTransferCallToEmployeeVoice] ... RequiredField.'Phone #'="?*" Param.PhoneNumber = "{Phone #:Lookup}" ...	[Command:BlindTransferCallToEmployeeVoice] ... [CmdData:BlindTransferCallToEmployeeVoice] ... RequiredField.'Login Name' = "?*" Param.PhoneNumber = "{$HotelingPhone(Login Name):Lookup}" ...
[Command:BlindTransferCallToPopupEmployeeVoice] ... [CmdData:BlindTransferCallToPopupEmployeeVoice] ... SelectQuerySpec = "[Phone #] IS NOT NULL" Param.PhoneNumber = "[Phone #:Lookup]" ...	[Command:BlindTransferCallToPopupEmployeeVoice] ... [CmdData:BlindTransferCallToPopupEmployeeVoice] ... SelectQuerySpec = "[Login Name] IS NOT NULL" Param.PhoneNumber = "[$HotelingPhone(Login Name):Lookup]" ...

Table 20: Commands to Modify for Hoteling (Continued)

Hoteling is OFF (out-of-the-box)	Hoteling is ON

[Command:ConsultativeTransferToEmployeeVoice_
Ordinary]
...
[CmdData:ConsultativeTransferToEmployeeVoice_
Ordinary]
 	
   ...
   RequiredField.'Phone #'="?*"
   Param.PhoneNumber = "{Phone #:Lookup}"
   ...
	[Command:ConsultativeTransferToEmployeeVoice_
	Ordinary]
	...
	[CmdData:ConsultativeTransferToEmployeeVoice_
	Ordinary]
	
	   ...
	   RequiredField.'Login Name'="?*"
	   Param.PhoneNumber = "{$HotelingPhone(Login
	Name):Lookup}"
	   ...

[Command:ConsultativeTransferToEmployeeVoice_
Campaign]
...
[CmdData:ConsultativeTransferToEmployeeVoice_C
ampaign]
 	
   ...
   RequiredField.'Phone #'="?*"
   Param.PhoneNumber = "{Phone #:Lookup}"
   ...
	[Command:ConsultativeTransferToEmployeeVoice_
	Campaign]
	...
	[CmdData:ConsultativeTransferToEmployeeVoice_C
	ampaign]
	
	   ...
	   RequiredField.'Login Name'="?*"
	   Param.PhoneNumber = "{$HotelingPhone(Login
	Name):Lookup}"
	   ...

[Command:ConsultativeTransferToPopupEmployee
Voice_Ordinary]
...
[CmdData:ConsultativeTransferToPopupEmployeeV
oice_Ordinary]
 	
   ...
   SelectQuerySpec = "[Phone #] IS NOT NULL"
   Param.PhoneNumber = "[Phone #:Lookup]"
   ...
	[Command:ConsultativeTransferToPopupEmployee
	Voice_Ordinary]
	...
	[CmdData:ConsultativeTransferToPopupEmployeeV
	oice_Ordinary]
	
	   ...
	   SelectQuerySpec = "[Login Name] IS NOT
	NULL"
	   Param.PhoneNumber = "[$HotelingPhone(Login
	Name):Lookup]"
	   ...

[Command:ConsultativeTransferToPopupEmployee
Voice_Campaign]
...
[CmdData:ConsultativeTransferToPopupEmployeeV
oice_Campaign]
 	
   ...
   SelectQuerySpec = "[Phone #] IS NOT NULL"
   Param.PhoneNumber = "[Phone #:Lookup]"
   ...
	[Command:ConsultativeTransferToPopupEmployee
	Voice_Campaign]
	...
	[CmdData:ConsultativeTransferToPopupEmployeeV
	oice_Campaign]
	
	   ...
	   SelectQuerySpec = "[Login Name] IS NOT
	NULL"
	   Param.PhoneNumber = "[$HotelingPhone(Login
	Name):Lookup]"
	   ...

Table 20: Commands to Modify for Hoteling (Continued)

Hoteling is OFF (out-of-the-box)	Hoteling is ON

[Command:ConferenceTransferToEmployee]
...
[CmdData:ConferenceTransferToEmployee]
 	
    RequiredField.'Phone #'="?*"
    Param.PhoneNumber = "{Phone #:Lookup}"
    ...
	[Command:ConferenceTransferToEmployee]
	...
	[CmdData:ConferenceTransferToEmployee]
	
	    RequiredField.'Login Name'="?*"
	    Param.PhoneNumber = "{$HotelingPhone(Login Name):Lookup}"
	    ...

[Command:ConsultativeTransferToPopupEmployee]
...
[CmdData:ConsultativeTransferToPopupEmployee]
 	
    Param.PhoneNumber = "[Phone #:Lookup]"
    ...
	[Command:ConsultativeTransferToPopupEmployee]
	...
	[CmdData:ConsultativeTransferToPopupEmployee]
	
	    Param.PhoneNumber = "[$HotelingPhone(Login Name):Lookup]"
	    ...

End of procedure

Next Steps

- There are no further steps.

Checking Installations

This section includes the preliminary information and instructions for checking your implementation of each Voice feature to verify that it is correctly installed and working. This section includes the following procedures:

Disclaimer

Only a thorough testing of every possible configuration and variation can provide a complete software application test. The tests listed below provide only a suggested starting point. If these tests are acceptable, Genesys recommends that you test each individual feature and variations by applying the procedures listed for that feature in the *Gplus Adapter 8.0 for Siebel CRM User's Guide*. These preliminary tests are not intended to substitute for a professionally designed and administered panel of tests.

Procedure:
Voice Component: Checking the basic Voice feature installation

Purpose: To check the functionality of the Basic Voice feature.

Start of procedure

1. Log in as an agent.

2. If are in the Not Ready state, select the Ready button.

3. Make a direct call to the agent Teleset.
    The ringing icon on the Siebel Communication toolbar should start blinking, indicating that a new call is waiting for the agent to answer.

4. Click the Answer incoming call button to establish a connection.

5. Confirm that the normal telephone or headset operations and the CTI toolbar buttons are present.

6. Confirm that the basic voice functions are coordinated with the basic voice controls on the CTI toolbar.

7. After the communication with the caller is completed, click the Hangup call button to release the call.

8. Type a phone number into the edit field of the Communication toolbar and click the Make Call to "<Phone Number>" button and then do one of the following:
    *   If this outgoing call is answered, then, after the communication is completed, click the Hangup call button to release the call.
    *   If this call is not answered, click the Hangup call button to release the call.

9. Try out other call scenarios—for example, a conference and a transfer call. Refer to the *G*plus *Adapter 8.0 for Siebel CRM User's Guide* for examples of these scenarios.

End of procedure

Next Steps

*   Check the Expert Contact installation, if this feature is installed. See, Procedure: Voice Component: Checking the Expert Contact installation, on page 272.

Procedure:
Voice Component: Checking the Expert Contact installation

Purpose: To check the Expert Contact installation feature-enabled Siebel Call Center.

Start of procedure

1. Log in as an agent and select the Ready button.

2. Click the On Call button on the Siebel Communication toolbar.
    A new incoming call from an unknown recipient appears in the work items list on the CTI toolbar. Proceed with the call as if it were an incoming call so Genesys can track this call. From this point onward, use the same checking process applied to the Basic Voice feature, with the exception of testing of the confirm-status option.

3. If the Confirm-status configuration option is implemented, then a popup window should appear after the defined time-out period. That is, if an expert is on the call after the length of time set by the timer, a dialog box appears asking if the expert is still on the call and if the agent wants to continue.

4. To test the Preview-interaction option (Preview mode) installation when the Preview mode is turned ON:

    a. When an agent receives an incoming call, a new work item, Preview Request, is created on the Siebel Communication toolbar, and the Accept Incoming Interaction button starts blinking. To accept the interaction, click the blinking button and use the standard procedures for the call. The newly created call reuses the existing work item.

    b. If the Siebel preview-bell option is implemented, then you can test it by sending an interaction to an expert. Confirm that a sound is played when the Preview Interaction dialog box appears.

5. To test the Preview mode installation when the Preview mode is turned off:

    •   If the Preview mode is set to OFF, then the agent's work flow options are exactly the same as in the Basic Voice feature. The agent can place outbound calls, receive incoming calls, and make transfers.

End of procedure

Next Steps

•   Check the Voice Callback installation. See, Procedure: Voice Component: Checking the Voice Callback installation, on page 273.

Checking the Voice Callback Installation

This section presents simple scenarios that show different ways of working with the Voice Callback Server from the Siebel application.

To check the functionality of the Universal Callback Server (UCB) -enabled Siebel Call Center, you must create a callback request from either the Siebel desktop or the interactive voice response (IVR) side. After creating the callback request, the next step is to make an agent ready and wait for a callback request to be distributed from the voice callback queue.

Procedure:
Voice Component: Checking the Voice Callback installation

Purpose: To check the Voice Callback installation by following the agent workflow scenarios described in the procedure.

Start of procedure

1. To initiate a callback session:
   - Click the Session start button on the Communication toolbar.

2. To initiate a callback call (in Callback Preview mode).
   a. After the callback interaction is delivered to the agent and the Contact screen displays, click the Initiate Callback button on the toolbar, which initiates a call to the number that was specified in the VCB_CONTACT key.
   b. To initiate a callback call to a different number:
      i. Enter a new phone in the Phone field on the CTI toolbar.

      or

      ii. Select and highlight any text on the agent's desktop and click Initiate Callback.

      Either of these options initiates a call to the selected number.

3. To add a callback session:
   a. Create a new Activity record.
   b. Set type to Call.
   c. Set the Priority.
      - Set to ASAP.

      or

      - Set the priority level to a value different than ASAP and also set the Due time value.
   d. Save the record.

      e. Click Add.

4. To cancel the callback session:

      a. Locate the callback record.

      b. Save the record.

      c. Click the Cancel button.

5. To reschedule a callback session:

      a. Locate the callback record.

      b. Set the Due time value.

      c. Change the priority value to a non-ASAP value.

      d. Save the record.

      e. Click Reschedule.

End of procedure

Next Steps

- Check the Outbound Contact installation. See,

Checking the Outbound Contact Installation

To check the functionality of the Outbound Contact Server (OCS) -enabled Siebel Call Center, you must create campaign entries that are exported by the G*plus* Campaign Synchronization Component into the Genesys OCS database and start a particular campaign from the Genesys Outbound Management Console.

Procedure:
Voice Component: Checking the Outbound Contact installation

Purpose: To check the Outbound Contact installation.

Start of procedure

1. Log into the Siebel Call Center application as one of the agents that you created during the procedure. Scenarios are processed differently, depending on the OCS

campaign mode and are described below. For further information on how to work with the Genesys Outbound Contact Server, refer to the Genesys *Outbound Contact Getting Started Guide.*

Campaign Running in Preview Mode

2. When a campaign is running in the Preview mode, log into the Siebel Call Center, and then log into PBX/ACD on the Communications toolbar.

   The Campaign [Campaign_Name] started message appears on the status line.

3. Click Preview Mode Start, to enable the following campaign control buttons on the Communications Toolbar: Request Record, Request Chain, Reject Record, Record Processed, Record Cancel, Mark Phone Do Not Call, and Preview Mode Stop.

Campaign Running in Push Preview Mode

4. When a campaign is running in the Push Preview mode, log into the Siebel Call Center, then log into PBX/ACD, then log into Interaction Server (the G*plus* Multimedia component must be installed), and set Ready for Push Preview on the Communications Toolbar.

   On every requested preview record, the Campaign Overview screen with a particular Contact/Prospect, is displayed and the following additional controls, Make Call, Done Reject, and Do Not Call, are available on the interface. Also, all changes to the following fields, Call Result, Attempts, and Schedule Time, are written to the Genesys Outbound Contact Server and the Siebel database.

   The Done control (on the interface) and the Record Processed button (on the Communications toolbar) write the operation and initiate the final OCS database update. After this command, the Preview record is removed from the Communications drop-down Work Items list. An active call

corresponding to this contact remains active and can be finished by clicking Release Call. Alternatively, the call can be released and the command, Record Processed, can be issued later.

Figure 29 on page 276 shows an OCS Preview record with an initiated outbound call. The Preview campaign name also displays.
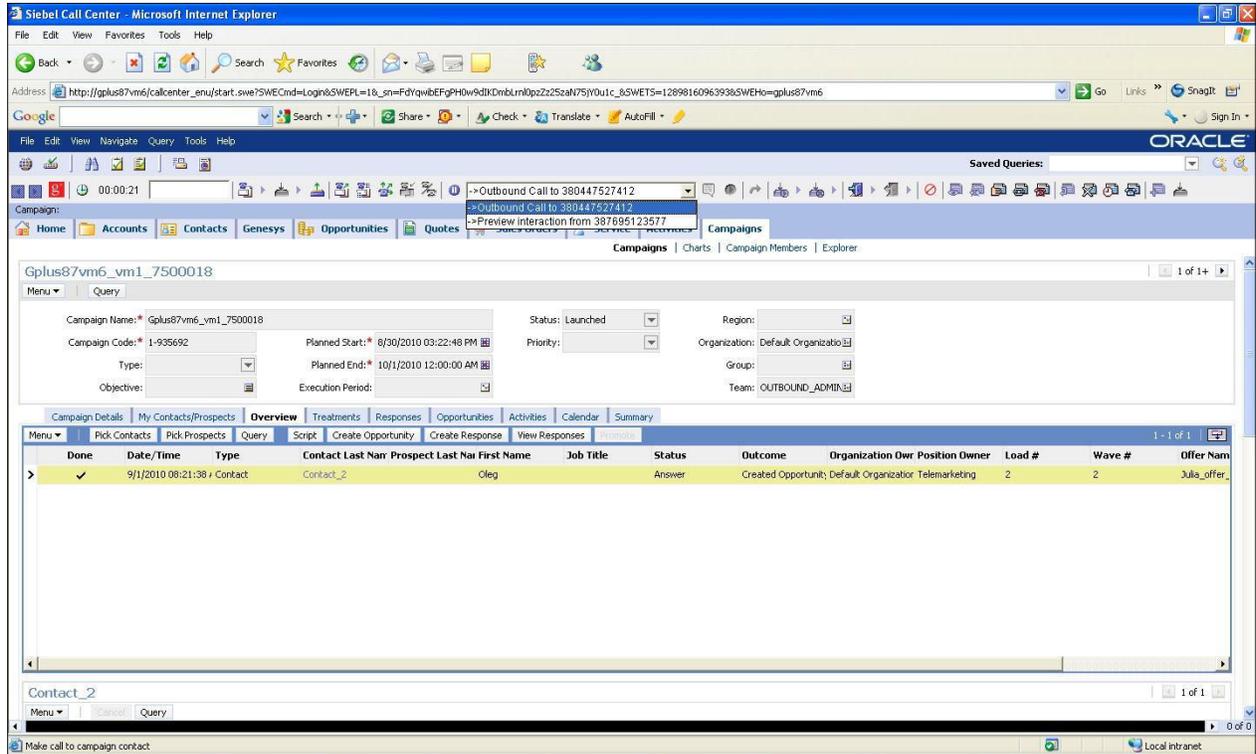
Figure 29: OCS Preview Record Window

Campaign Running in Predictive or Progressive Mode

5. When a campaign is running in Predictive or Progressive dialing mode, log into the Siebel Call Center application, and then log into the PBX/ACD, the initial state is Agent Not Ready.

A new outbound call is delivered to the agent when the agent's status changes from Not Ready to Ready. The campaign control buttons Record Processed, Request Chain, Record Cancel, and DoNotCall become available on the Communications toolbar.

On every delivered call, the Campaign Contact Overview screen with a particular Contact/Prospect is displayed and the following additional controls, Done and DoNotCall, are active and available on the interface. Also, all changes to the Call Result and Attempts, Schedule Time fields are written to the Genesys Outbound Contact Server (OCS) and the Siebel database.

The Done and DoNotCall controls (on the interface) and the Record Processed and DoNotCall controls (on the Communications toolbar) save the operation and initiate the final OCS database update. After that, the active call command remains active and can be finished by clicking Release Call. Also,

the call can be released and the Record Processed command can be issued later. In this scenario, a new work item
Wrap Up Record ->[Phone number] appears on the Communications toolbar drop-down list and is removed after the agent clicks Done, Record Processed, Record Cancel, or DoNotCall.

Figure 30 shows the initial Siebel Call Center screen after the agent logs in. The campaign name also displays.
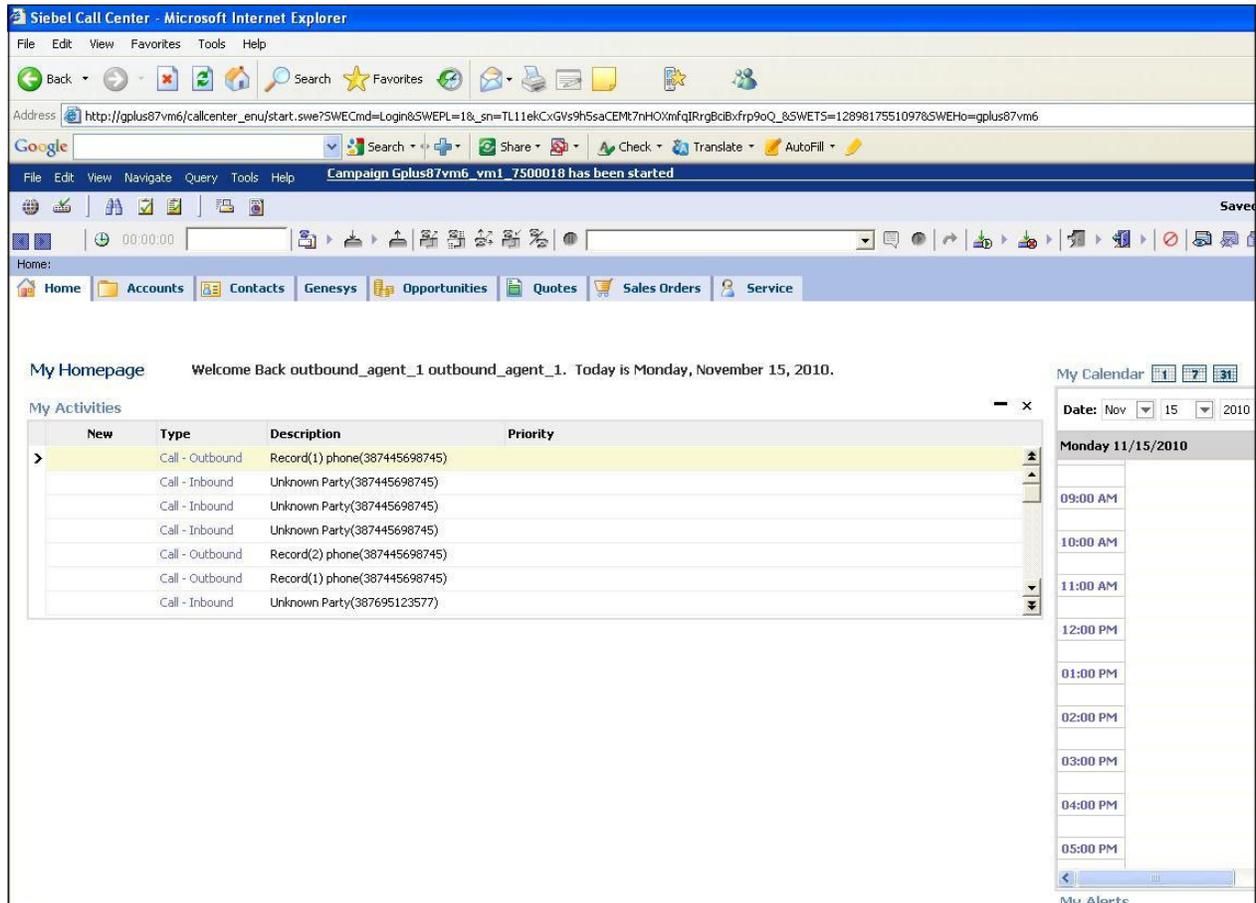
Figure 30: Siebel Call Center Window

Figure 31 on page 278 shows an outbound call delivered to the agent. The campaign name also displays.
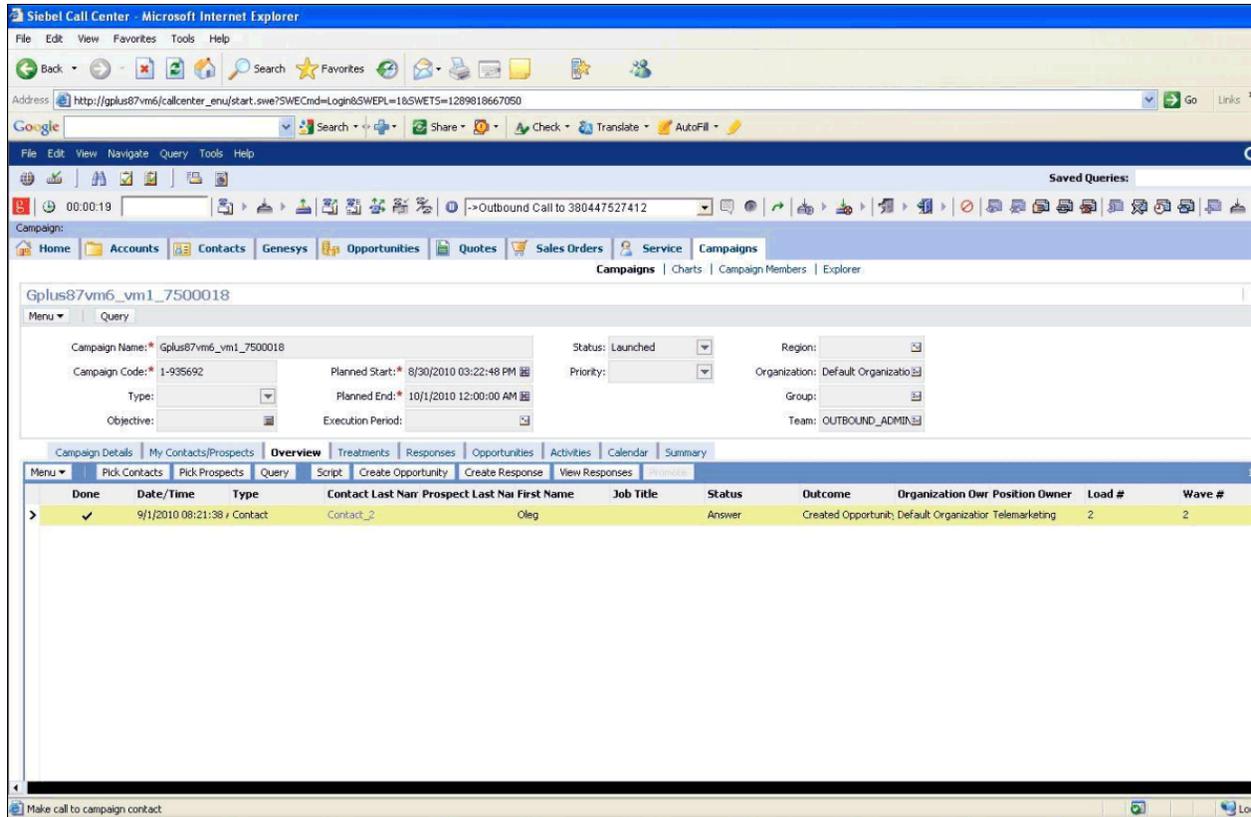
Figure 31: Siebel Call Center Window, Outbound Call

End of procedure

Next Steps

- There are no further steps.

The Driver Settings for the Voice Component

The Voice Component Driver Parameters

The driver parameters define the set of values used by the Siebel Communications Server to work with communications systems such as the CTI middleware. The G*plus* Adapter driver is one of these systems and an appropriate set of parameters must be defined for it. A sample configuration file GenComm_universal.def contains a set of parameters to be defined for the Voice Component.

Table 21 lists the supported driver parameters for the G*plus* Adapter driver for the Siebel Communications Server. The table indicates which parameters are required and displays applicable default values. You view and modify these driver parameters in the Communications Drivers and Profiles view in the Administration - Communications screen). The channel type for this driver is G*plus* Universal.

Note: A Siebel administrator must configure the Siebel environment to work with G*plus* Adapter.

Each parameter is prefaced with a keyword indicating how it is used:

- Parameters prefaced with Driver: are sent to the driver handle when the driver is initialized. These parameters are sent to the CreateISCDriverInstance method.
- Parameters prefaced with Service: are sent to the driver handle when it requests a service (creates the service handle). These parameters are sent to the RequestService method.

Note: You must change the value of each mandatory parameter with the default value CHANGE_ME as these parameters depend on your configuration and cannot be used with default values.

For more information about the driver handle and its methods, refer to Siebel documentation.

Table 21: The Voice Component Driver Parameters

Default Value	Required	Comment
Parameter Name: Driver:AdapterAppName		
Gplus adapter	No	The application name that is sent to T-Server when the Adapter opens a connection to this T-Server. (For compatibility with the pre-7.5 scheme of connection to T-Server only.)
Parameter Name: Driver:AnswerCallDelay		
500 milliseconds	No	Defines the delay between TMakeCall and TAnswerCall requests for outbound calls on a DMS-100 switch.
Parameter Name: Driver:BackupGenCommServerURL		
CHANGE_ME	No	Specifies the URL of the backup G*plus* Communication Server.

Table 21: The Voice Component Driver Parameters (Continued)

Default Value	Required	Comment
Parameter Name: Driver:BackupServerHost		
CHANGE_ME	No	Specifies the name of the host machine on which the backup Genesys T-Server is running. (For compatibility with the pre-7.5 scheme of connection to T-Server only.)
Parameter Name: Driver:BackupServerPort		
CHANGE_ME	No	Specifies the TCP/IP port number the client must use to establish a connection with the backup Genesys T-Server. (For compatibility with the pre-7.5 scheme of connection to T-Server only.)
Parameter Name: Driver: BroadcastCommdnUserEvents2Agents		
FALSE	No	The value True enables the distribution of the EventUserEvent.
Parameter Name: Driver:CommunicationDN		
empty string ("")	No	Determines the name of a communication DN. The communication DN is used to send and/or broadcast a user event. See "Configuring the Communication DN" on page 241.
Parameter Name: Driver:HangupOnEventReleased		
TRUE	No	Applicable only when using the NEC NEAX switches—for example, NEC NEAX 2000 IPS, NEC NEAX ICS/IMS (7600, 7400). If the switch configuration requires the phone to be put on hook after call is released on the other side, then the *Gplus* Adapter should send an additional Release request after receiving EventReleased, and the value of Driver: HangupOnEventReleased must be set to "TRUE" (default). Driver:HangupOnEventReleased = "TRUE" If the switch configuration does not require you to put the phone on the hook, then set this parameter to "FALSE"
Parameter Name: Driver:InboundPartyContainer		
empty string ("")	No	Name of the TEvent attribute for substitution in the work item name on the Siebel Toolbar for incoming calls

Table 21: The Voice Component Driver Parameters (Continued)

Default Value	Required	Comment
Parameter Name: Driver:LibraryName		
UniComm	Yes	The name of the remote driver module which is loaded by *Gplus* Communication Server to handle requests for the Siebel Communication Driver.
Parameter Name: Driver:LocalConnString		
CHANGE_ME	Yes, if Network Attended Transfer is used.	Determines the name of this premise location when the Network Attended Transfer feature is used. The value of this parameter must be set as a premise switch name exactly as it is defined in the Genesys Configuration Manager for the T-Server. See "Network Attended Transfer/Conference Feature" on page 252.
Parameter Name: Driver:NtwkReconnectTimeout		
0 (zero)	No	Defines whether automatic reconnection of the initiated Network Transfer is configured and sets the timeout value. See "Network Attended Transfer/Conference with Optional Automatic Reconnection" on page 255.
Parameter Name: Driver:OutboundPartyContainer		
empty string ("")	No	Name of the TEvent attribute for substitution in the work item name on the Siebel Toolbar for outgoing calls.
Parameter Name: Driver:PrimaryGenCommServer URL		
CHANGE_ME	Yes	Specifies the Universal Resource Locator (URL) of the primary *Gplus* Communication Server.
Parameter Name: Driver:ReconnectTimeout		
10	No	Defines the time in seconds for reconnection to T-Server or if defined, to the backup T-Server when T-Server disconnects from the Adapter. (For compatibility with the pre-7.5 scheme of connection to T-Server only.)
Parameter Name: Driver:RemoteTimeout		
10	No	This parameter defines a time interval, in seconds, during which the T-Server waits for an ADDP response from the Adapter. (For compatibility with the pre-7.5 scheme of connection to T-Server only.)

Table 21: The Voice Component Driver Parameters (Continued)

Default Value	Required	Comment
Parameter Name: Driver:ServerHost		
CHANGE_ME	No	Name of the host that runs Genesys T-Server. (For compatibility with the pre-7.5 scheme of connection to T-Server only.)
Parameter Name: Driver:ServerPort		
CHANGE_ME	No	TCP/IP port number for T-Server. (For compatibility with the pre-7.5 scheme of connection to T-Server only.)
Parameter Name: Driver:SupervisorExtensions		
Empty string (No Supervisor Extensions)	No	Defines a set of Supervisor Extensions which should not be shown on the agent desktop when a supervisor is connected to the agent's current interaction. Valid values are: • a string parameter consisting of a set of DNs • a simple file name (See information about this parameter below this table.)
Parameter Name: Driver:SwitchType		
0 (zero)	Yes	The type of switch for your contact center. Values include: 1–Nortel Meridian 2–Rockwell Spectrum 4–Nortel DMS-100 5–Lucent Definity G3 6–Aspect Call Center 7–Siemens Hicom 300E 8–EADS PointSpan, EADS ESeries 9–Ericsson MD110 12–NEC NEAX 2400 12-NEC NEAX 3600i 14–Teltronics 20-20 IXP 17–Siemens Hicom 300 18–Phillips Sopho IS3000

Table 21: The Voice Component Driver Parameters (Continued)

Default Value	Required	Comment
		23—Alcatel 4400 29—Nortel Meridian Call Center Symposium 31—Siemens Realitis DX (iSDX) 49—Avaya INDeX (Lucent SDX) 50—Siemens Hicom 300H 51—Siemens HiPath 4000 52—Alcatel A4200 OmniPCX Office (OXO) 58—Knowledge Worker Gateway 62—Cisco CallManager 67—Mitel MN—3300 69—Siemens HiPath 3000 70—eOn eQueue 72—SIP Server Also see the document, *Genesys Supported Media Interfaces Reference Manual,* which is available on the Genesys Documentation website. Contact Genesys for a current list of switches and SwitchType values. (For compatibility with the pre-7.5 scheme of connection to T-Server only.)
Parameter Name: Driver:Timeout		
0 (zero)	No	This parameter defines a time interval, in seconds, during which the Adapter waits for an ADDP response from the T-Server. ADDP is disabled if this parameter is not defined or if the value is set to 0. (For compatibility with the pre-7.5 scheme of connection to T-Server only.)
Parameter Name: Driver:TServerAppName		
"CHANGE_ME"	Yes	Defines the T-Server Application Name of the Primary T-Server as it is defined in the Genesys Configuration Manager. See "T-Server Connections" on page 223.
Parameter Name:	Driver:BCTServerAppName (SIP BC configuration only)	
<empty>	Yes	Defines the T-Server Application Name of the Primary T-Server on the alternative SIP BC Peer Site, as it is defined in the Genesys Configuration Manager. See "T-Server Connections" on page 223.
Parameter Name: Name		
Gplus_Universal	Yes	Name of the driver (can be any suitable name).

Table 21: The Voice Component Driver Parameters (Continued)

Default Value	Required	Comment
Parameter Name: Service:ACDDNList		
{@ACDDNList}	No	Uses the macro @ACDDNList to obtain a list of ACD DNs (Extensions of type A) associated with the current agent.
Parameter Name: Service:AgentId		
{@AgentId}	No	This parameter manages agent ID logging for server-side statistics on the outgoing connections usage. If the log level is set to DEBUG and if this parameter is specified, the Agent ID is logged. Otherwise, the session ID is logged.
Parameter Name: Service:AgentLogOutControl		
False	No	Applies only to the Predictive dialing mode. When this option is set to TRUE, the agent logout control functionality is enabled and all agents requesting logout are notified regarding the time left until actual logout and upon reaching this time the agents are automatically logged out from the Communication services.
Parameter Name: Service:AgentPreviewModeStart		
False	Yes (when working with Outbound campaigns in Preview mode)	Defines whether the agent is able to receive preview records without first submitting a PreviewDialingModeStart request to OCS from the desktop. In the Siebel Configuration, the Service:AgentPreviewModeStart parameter must be set to correspond with the value of the OCS option agent_preview_mode_start in Configuration Manager. See details in the *Genesys Outbound Contact Deployment Guide*.
Parameter Name: Service:AgentSubstitute		
True	Yes (when using the Alcatel A4400 switch)	For the Alcatel A4400 switch, this option must be set in accordance with T-Server settings for the Agent Substitute option. See more details in "Switch-Specific Configuration Instructions" on page 240.
Parameter Name: Service:AgentWorkMode		

Table 21: The Voice Component Driver Parameters (Continued)

Default Value	Required	Comment
AgentWorkMode Unknown	No	Agent work mode. Should match one of the T-Library TAgentWorkMode enumeration values: 0-AgentWorkModeUnknown 1-AgentManualIn 2-AgentAutoIn 3-AgentAfterCallWork 4-AgentAuxWork 5-AgentNoCallDisconnect The parameter can be defined either as a number corresponding to the T-Library TAgentWorkMode enumeration values or as a string (for example, 1 or AgentManualIn).
Parameter Name: Service:AutoLogout		
FALSE	No	If the value of this parameter is TRUE, when the agent logs out from Siebel or closes the browser, then first the agent logs out from the ACD Queue group, and then the agent logs out from Siebel. When the value of this parameter is FALSE, the agent just logs out from Siebel.
Parameter Name: Service:AutoRejectRecords		
TRUE	No	Defines whether the Adapter sends requests to reject records when an agent logs out from Siebel. If the value of this parameter is TRUE, when an agent logs out from Siebel the Adapter sends requests to reject all records. If the value of this parameter is FALSE, the Adapter does not send requests to reject records. **Note:** Applicable only for Preview campaigns.
Parameter Name: Service:DNList		
"{@DNList}"	Yes	Uses the macro @DNList to obtain a list of DNs (standard extensions of type S) associated with the current agent.
Parameter Name: Service:DeliveryMarkUserDataKey		
AgentProcessing	No	Defines the User Data key that is used to mark interaction as delivered to an agent. For details about this option, see Chapter 11, "Deploying the Multimedia Session Failover Handling," on page 545.
Parameter Name: Service:EntranceTimestamp		
Empty string	No	If this option value is populated (not empty), the Adapter tries to apply the Compute-TimeInQueue detection mode to obtain the TimeInQueue attribute for a given call.

Table 21: The Voice Component Driver Parameters (Continued)

Default Value	Required	Comment
Parameter Name: Service:FrozenTimeInQueue		
Empty string	No	If this option value is populated (not empty), the Adapter tries to read the corresponding value from the call-attached data (by using the Frozen–TimeInQueue detection mode) with this option value as its key. After reading the value, the Adapter sends the results to Siebel in the form of the TimeInQueue attribute for the call.
		Definitions:
		• Compute-TimeInQueue detection mode—From the CAD, the Adapter reads a value that is associated with the key that is provided by the EntranceTimestamp parameter. This value is a string that must conform to the Coordinated Universal Time (UTC) format: YYYY-MM-DDThh:mm:ss. The Adapter then calculates the TimeInQueue attribute (in seconds) for the call (The difference between the current time, converted to UTC, and the time that the Adapter read from the CAD.)
		• Frozen–TimeInQueue detection mode—From the CAD, the Adapter reads a value that is associated with the key that is provided by the FrozenTimeInQueue parameter. This value is an integer (in seconds) that represents the TimeInQueue attribute of the call, which was calculated by an external application or software. The Adapter sends this value to Siebel with no changes (as is).
		• PBX-TimeInQueue detection mode—The Adapter reads from the Extensions call attribute to obtain the PBX/switch-specific TimeInQueue parameter and sends the value to Siebel. (This attribute is not supported by all PBXs or switches.)
		Note: If none of these options is defined, or if there are no keys related to them in the call attached data, the Adapter uses PBX-TimeInQueue detection mode (the existing Adapter mode available prior to this release).
		If both of these options are defined, and both have keys present in the CAD, then the Adapter uses Frozen–TimeInQueue detection mode, which has a higher priority.
		Example: If both of the new options are set in Siebel, as in the following example:
		[Driver Parameters:Gplus_Voice] Required:Name = "Gplus_Voice" … Service:FrozenTimeInQueue = "Key1" Service:EntranceTimestamp = "Key2"
		then there are four possible ways in which the Adapter might interpret this, and determine the TimeInQueue value that is sent to Siebel.

Table 21: The Voice Component Driver Parameters (Continued)

Default Value	Required	Comment
Parameter Name: Service:FrozenTimeInQueue (Continued)		
Empty string	No	For example: • If only the pair with the value of Key1 is present in the CAD, the Frozen-TimeInQueue detection mode is used. • If only the pair with the value of Key2 is present in the CAD, the Compute–TimeInQueue detection mode is switched to ON. • If both pairs have their respective values, Key1 and Key2, in the CAD, the Frozen-TimeInQueue detection mode is used because of its higher priority. • If both of the Key1 and Key2 values are absent from the pairs in the CAD, the PBX-TimeInQueue detection mode is used.
Parameter Name: Service:HasAgentBusy		
FALSE	No	Can be TRUE or FALSE. Specifies whether the connected switch supports the Do Not Disturb (DND) function. You can also disable the DND function if it is supported by the switch, but you do not want your agents to use it. If the value is FALSE, the associated feature is disabled. For a disabled feature, an administrator can remove the associated (DND) toolbar button from the repository using Siebel Tools.
Parameter Name: Service:HasAnswer		
TRUE	No	Upon an incoming call, if the value of this parameter is TRUE, the Answer Incoming call button on the toolbar is available. If the value is FALSE, this button is unavailable.
Parameter Name: Service:HasDisconnect		
TRUE	No	For the current active call, if the value of this parameter is TRUE, the Hang-up call button on the toolbar is available. If the value is FALSE, this button is unavailable.
Parameter Name: Service:HasForward		
TRUE	No	If both the switch and T-Server support call forwarding, and this value is set to TRUE, the button on the toolbar that turns call forwarding on or off is available. If this value is FALSE, this button is unavailable.
Parameter Name: Service:HasHold		
TRUE	No	If the value of this parameter is TRUE, the Hold button or Retrieve button is available, depending on call state. If the value of this parameter is FALSE, the Hold or Retrieve button is not available.

Table 21: The Voice Component Driver Parameters (Continued)

Default Value	Required	Comment
Parameter Name: Service:HasKWProtocol		
FALSE	No	Enables or disables the Expert Contact functionality support.
Parameter Name: Service:HasOCSProtocol		
TRUE for Outbound Contact configurations	No	Enables or disables the Outbound Contact functionality support.
Parameter Name: Service:HasVCBProtocol		
FALSE	No	Enables or disables the Voice Callback functionality support.
Parameter Name: Service:KwDefaultPartyStatus		
released	No	Used by *Gplus* Adapter when the party status info is missing in the KwPartyStatusRequest request from Siebel.
Parameter Name: Service:KwPiProtocolVersion		
6.5.000.01	No	The Preview Interaction protocol version.
Parameter Name: Service:KwProtocolVersion		
6.5.000.01	No	The Knowledge Worker protocol version.
Parameter Name: Service:NetworkCallTransferMode		
Network	No	This parameter has two valid values: • Network • Local If the value of this parameter is Network, the Adapter uses the Network T-Server operation (Network Consult/Transfer/Conference) for creating and completing two-step operations (consult, transfer and conference) for network calls; all other calls are processed using the standard T-Server operations. If the value of this parameter is Local, the Adapter uses only the standard T-Server operations for two-step operations (consult, transfer and conference) for all types of calls.

Table 21: The Voice Component Driver Parameters (Continued)

Default Value	Required	Comment
Parameter Name: Service:OCSCallBackType		
Personal	No	Specifies the callback type: Personal, or Campaign. These can be chosen from the Siebel popup list, and the corresponding strings will be sent as GSW_CALLBACK_TYPE values to OCS. For more information, see "Advanced Customization of Voice Component Configurations" on page 233.
Parameter Name: Service:OCSCancelScope		
AllChain	No	Specifies the scope of a cancel: AllChain, or RecordOnly. For more information, see "Advanced Customization of Voice Component Configurations" on page 233 and "USE_RECORD_ HANDLE Use Case" on page 261.
Parameter Name: Service:OCSDoNotCallScope		
AllChain	No	Specifies the scope of a Do Not Call: AllChain, or RecordOnly. For more information, see "Advanced Customization of Voice Component Configurations" on page 233 and "Specifying the Do Not Call Command Attribute Keys" on page 259.
Parameter Name: Service:OCSEnableCustomerId		
TRUE	No	Controls the use of Contact ID for compatibility with release 6.5.2 or earlier of Outbound Contact Server. For more information, see "Advanced Customization of Voice Component Configurations" on page 233.
Parameter Name: Service:OCSTreatment		
RecordTreat Personal	No	Specifies the treatment type: RecordTreatPersonal, RecordTreatCampaign, or None. These can be chosen from the Siebel popup list as Personal, Campaign, or None, and for the first two, the corresponding strings will be sent as GSW_CALLBACK_TYPE values to OCS. For None, GSW_CALLBACK_TYPE is not sent to OCS. For more information, see "Advanced Customization of Voice Component Configurations" on page 233.

Table 21: The Voice Component Driver Parameters (Continued)

Default Value	Required	Comment
Parameter Name: Service:ReleaseCallOnAutoLogout		
TRUE	No	When the value of this parameter is set to FALSE, the Adapter will not release an active call if an agent closes the browser or logs out from Siebel. If the value of this parameter is TRUE and the option AutoLogout = "TRUE", when the agent logs out from Siebel or closes the browser, then:
		1. The Adapter sends a request to release the call if the ReleaseCall command is enabled in the current call state;
		2. The Adapter sends a request to log out the agent from the ACD group if the LogOut command is enabled in the current state;
		3. Only then does the agent log out from Siebel.
		Note: This functionality is switch and T-Server dependent. Some T-Servers may require changes to options in order to keep an active call on logout.
Parameter Name: Service:ReleaseEstablishedCallOnAutoLogout		
TRUE	No	If the value of this parameter is set to FALSE, the Adapter will not release an established call if an agent closes the browser or logs out from Siebel.
		If the value of this parameter is TRUE and the option AutoLogout = "TRUE", when the agent logs out from Siebel or closes the browser, then:
		1. The Adapter sends a request to release an established call if the ReleaseCall command is enabled in the current call state;
		2. The Adapter sends a request to log out the agent from the ACD group if the LogOut command is enabled in the current state;
		3. Only then does the agent log out from Siebel.
		Note: This functionality is switch and T-Server dependent. Some T-Servers may require changes to options in order to keep an active call on logout.
Parameter Name: Service:RequestTimeOut		
30	No	The timeout period for requests to Outbound Contact Server.

Table 21: The Voice Component Driver Parameters (Continued)

Default Value	Required	Comment
Parameter Name: Service:SelectDN		
{@SelectedDN}	No	Specifies a DN to be used to issue commands such as MakeCall and so forth. Otherwise, the first DN in the list specified with the parameter Service:DNList is used.
		This parameter is also used for SendUserEvent command; it defines the DN on which the user event is sent.
Parameter Name: Service:SwitchName		
CHANGE_ME	No	Voice switch name, which Voice Callback Server uses for submitting the Voice callbacks.
Parameter Name: Service:UsePendingWorkmode		
FALSE	No	Applicable only when using the Avaya Definity G3 switch. Parameter Service: UsePendingWorkmode must be set according to the T-Server option use-pending-work-mode. Both parameters must be set to the same value: true or false. When the T-Server option use-pending-work-mode is set to true, the Pending Work Mode functionality is enabled.
Parameter Name: Service:UseSStepTransferForNetworkTransfer		
FALSE	No	To perform a network transfer with the device command parameter TransferSStep for all switches, set this parameter to TRUE. The device command parameter RemoteConnectStr must define the name of the Network Switch where call should be transferred.
		When this parameter is set to FALSE, for the switches that support Mute Transfer and do not support Single Step Transfer, the Adapter will invoke the request to T-Server to perform Mute Transfer with the Device Command, TransferSStep, for the switches that support Mute Transfer and do not support Single Step Transfer.
		If you do not perform network transfers, leave the value set to FALSE (the default value).
Parameter Name: Service:VCBDefaultCallbackType		
ASAP	No	The default Callback Type which the G*plus* Adapter uses in case the Callback type information is missing in the request from Siebel.
Parameter Name: Service:VCBDefaultRoutingPoint		
CHANGE_ME	No	The default routing point, which the G*plus* Adapter uses in case the routing point information is missing in the request from Siebel.

Table 21: The Voice Component Driver Parameters (Continued)

Default Value	Required	Comment
Parameter Name: Service:VCBProcessedOnCallRelease		
FALSE	No	Indicates that the work item is kept on the call release event and the agent must manually click the Interaction Processed button on the Communication toolbar. This parameter must be synchronized with the VCB server option callback_processed.

The Driver:SupervisorExtensions Driver Parameter

The Driver:SupervisorExtensions driver parameter defines a set of Supervisor Extensions that should not be shown on the agent desktop when the supervisor is connected to the agent's current Voice interaction (call).

The valid values for this parameter are defined by the following rules:

- A valid value is a string parameter consisting of: a set of DNs separated by commas, a range of DNs in which a hyphen (-) is used as a separator and the first number is less than the last number, or a mix of both.

- The maximum length of the string parameter is 250 characters.

- Spaces are allowed after commas, and before and after a hyphen.

  For example:

  2201, 2205, 2300 - 2374

- A valid value can also be a simple file name, or a path and a file name. In any environment, you can use a simple filename, such as SupervisorExtensions.txt, for example. Other examples are:

  - On a Windows platform - C:\siebsrvr\SupervisorExtensions.txt.

  - On a UNIX platform - /opt/siebsrvr/SupervisorExtensions.txt

  - The named file itself must contain the set of Supervisor Extensions. The file must contain a set of DNs separated by commas, or a range of DNs, or a mix, with valid values as defined in the rules above; and in addition:

    - New lines can also serve as separators.

    - The maximum number of Supervisor Extensions is limited by the allowed size of the file in the operating system.
      For example:
      2201,2210
      2211, 2212
      3
      2213-2220,
      2303 - 2310
      7123
      8234
      9345

◆ If the file name is provided without a path, then the G*plus* Adapter will search for the file in locations defined by the operating system environment variable ‹path›. If the file is not found, then the G*plus* Adapter will search for the file in the following locations:

◆ On a Windows platform - ‹Siebel Server folder›\BIN\

◆ On a UNIX platform - ‹Siebel Server folder›/lib/

The Driver: HangupOnEventReleased Driver Parameter

The Driver:HangupOnEventReleased driver parameter is used only when the G*plus* Adapter works with NEC NEAX/APEX switches—for example, NEC NEAX 2000 IPS, NEC NEAX ICS/IMS (7600, 7400). This configuration parameter must be set up accordingly when a NEC NEAX/APEX switch is used.

If the switch configuration requires the phone to be put on the hook after a call is released on the other side, then the G*plus* Adapter must send an additional release request after receiving an EventReleased event. In this case, the value must be set to "TRUE" (the default value), as follows:

Driver:HangupOnEventReleased = "TRUE"

If the switch configuration does not require the phone to be put on the hook, then set this parameter to "FALSE", as follows:

Driver:HangupOnEventReleased = "FALSE"

Note: The default value is "TRUE".

The Service:UsePendingWorkmode Service Parameter

The Service:UsePendingWorkmode service parameter is used to enable or disable Ready/NotReady device commands when an agent has a call on the Avaya Communication Manager (previously known as Avaya Definity G3) switch.

When using the Avaya Communication Manager switch, the Service:UsePendingWorkmode service parameter must be set according to the use-pending-work-mode T-Server configuration option. Both parameters must be set to the same value: true or false. When the T-Server use-pending-work-mode option is set to true, the pending work mode functionality is enabled, which allows agents to change their Ready/NotReady state when a call is in progress.

The Driver:InboundPartyContainer and Driver:OutboundPartyContainer Driver Parameters

The G*plus* Adapter displays (with certain TEvents) an adjusted (modified) work item name on the Siebel Toolbar by using the value of some other TEvent attribute.

The Siebel configuration parameters, Driver:InboundPartyContainer and Driver:OutboundPartyContainer, allow adjustments depending on their value. The

implemented functionality for both parameters is the same; the only difference is that the Driver:InboundPartyContainer parameter defines the adjustment (if any) for incoming calls, while the Driver:OutboundPartyContainer parameter defines the adjustment (if any) for outgoing calls.

If the Driver:InboundPartyContainer or Driver:OutboundPartyContainer parameters are defined in the Siebel configuration and if the values described by the parameters are found in the incoming TEvent attributes, then the value defined by this attribute is shown in a work item on the Communications toolbar.

If the Driver:InboundPartyContainer and/or Driver:OutboundPartyContainer parameters define the attribute name for an adjustment, but these attributes are not in the TEvent, then the OtherDN attribute is used.

The acknowledged values for Driver:InboundPartyContainer and Driver:OutboundPartyContainer and their actions are shown in Table 22:

Table 22: Driver:InboundPartyContainer and Driver:OutboundPartyContainer Values and Actions

Value	Action
No parameter in the configuration, or <empty string>	Keeps the representation of the work item list without the substitution of the value defined by the Driver:InboundPartyContainer or Driver:OutboundPartyContainer configuration parameters.
OtherDN, or Native	Keeps the representation of the work item list without the substitution of the value defined by the Driver:InboundPartyContainer or Driver:OutboundPartyContainer configuration parameters.
ANI	Substitutes the value of the TEvent ANI attribute in a work item representation on the Communications toolbar.
DNIS	Substitutes the value of the TEvent DNIS attribute in a work item representation on the Communications toolbar.
<Name of parameter defined in UserData of TEvent>	Substitutes the value of the <string or integer parameter defined in UserData of TEvent> in a work item representation on the Communications toolbar.

Important note!   The Driver:InboundPartyContainer and/or Driver:OutboundPartyContainer parameter substitution in the work item name is not performed in the conference scenarios when the number of parties displayed is greater than 1.

When using the UserData parameter to make a substitution in the work item name, the parameter described by the Driver:InboundPartyContainer and/or Driver:OutboundPartyContainer parameters must be included as a string or an integer pair in the UserData list and not hidden in the inner lists as shown in Example 15 on page 295.

Example 15   If the parameters are defined in a Siebel configuration as follows:

Driver:OutboundPartyContainer = "Destination"

Driver:InboundPartyContainer = "Originator"

UserData:

     (Int) Destination   43262

     (Int) Originator   87654

And/or

     (Str) Destination   str_43262

     (Str) Originator   str_87654

Important note!   If the modification of the work item name is going to be used and if the DNIS and ANI attributes are represented in the TEvents, then it is recommended to use the following values:

InboundPartyContainer   ANI

OutboundPartyContainer   DNIS

It is possible to use the modified appearance of the work item name on the Communications toolbar with a single command using the command data.

Example 16 is an example of the of Command:ConferenceTransferToPhone device command.

Example 16   If the parameters are defined in the Siebel configuration as follows:

Driver:OutboundPartyContainer = "Destination"

Driver:InboundPartyContainer = "Originator"

then the Param.Originator and Param.Destination parameters can be defined in the CmdData parameter to define the destination and the originator of the call.

So, the Command:ConferenceTransferToPhone parameter can be defined as follows:

[Command:ConferenceTransferToPhone]

  DeviceCommand   = "ConferenceInit"

  Description   = "Conference Transfer to "{@Phone}""

  CmdData   = "ConferenceTransferToPhone"

  OnEditControl   = "TRUE"

  Hidden   = "TRUE"

  CmdChannelOnFocus = "TRUE"

  Profile   = "Gplus Universal Profile"

[CmdData:ConferenceTransferToPhone]

   AttachContext   = "TRUE"

RequiredField.@Phone = "?*"

Param.PhoneNumber   = "{@Phone:PhoneTypeLookup}"

Param.CallNotifyText = "Conference transfer from {@UserName} ..."

Param.TrackingID   = "{@SelectedWorkItem:DriverWorkTrackID}"

Param.Originator   = "{@SelectedDN}"

Param.Destination   = "{@Phone:PhoneTypeLookup}"

When a conference call is initiated from the Siebel toolbar, then the phone number from the EditControl box on the toolbar and the SelectedDN is the call destination and the call originator.

When using parameters in the command data (UserData) as a substitution, the T-Server consult-user-data configuration option must be set to separate and not set to joint. If the value is set to separate, it separates the UserData attached to the call and this attached data is kept separate and not overwritten by the other data from the consultation calls.

The G*plus* Adapter Application Name

The parameter Driver:AdapterAppName configuration defines the application name that is sent to T-Server when the Adapter opens a connection to this T-Server.

If the parameter was not defined in the Siebel configuration, then the default value is used, which is Gplus adapter.

For example:

Driver:AdapterAppName=   "GP Adapter"

The Advanced Disconnect Detection Protocol (ADDP) Support

The G*plus* Adapter supports the Advanced Disconnect Detection Protocol (ADDP). ADDP allows the Adapter to detect disconnections from T-Server. To enable this protocol and configure it in the G*plus* Adapter, you must define the following parameters in the Adapter's configuration. (These parameters are also described in Table 21 on .)

- Driver:Timeout - This parameter defines a time interval, in seconds, during which the Adapter waits for an ADDP response from the T-Server. ADDP is disabled if this parameter is not defined or if the value is set to 0 (zero).

- Driver:RemoteTimeout - This parameter defines a time interval, in seconds, during which the T-Server waits for an ADDP response from the Adapter.

- Driver:TraceMode - This parameter defines an ADDP trace mode in T-Server and/or in Adapter log files. The value of this parameter affects the log generation only. The ADDP polling messages and the algorithm for the disconnection detection remain the same, regardless of the value set here.

Warning! To avoid the false detection of the disconnect states that might occur because of normal delays in the data network, Genesys recommends that you set the ADDP timeouts to values greater than 10 seconds, and that you set all local ADDP timeouts to the same value.

Emulated Agent Work Modes

The G*plus* Adapter can emulate the AgentManualIn and AgentAutoIn work modes for some switches that normally do not support these modes. For these switches, the value of the Service:AgentWorkMode parameter can be set to allow the emulation of an agent work mode.

Emulation means that the an agent state could be changed by the G*plus* Adapter according to the work mode defined in the Siebel configuration. The agent work mode is defined in a Siebel configuration by the value of the Service:AgentWorkMode or AgentWorkMode parameters sent with the following device commands: NotReady, Ready, ChangeNotReadyState, or Login.

The default value of the Service:AgentWorkMode is AgentWorkModeUnknown parameter.

The agent work mode could be changed when an agent invokes the NotReady, Ready, ChangeNotReadyState or Login device commands with the AgentWorkMode parameter set to the new value.

The valid values of the AgentWorkMode parameter and the corresponding desktop behavior can be summarized as follows:

- If the AgentWorkMode parameter is equal to the AgentWorkModeUnknown parameter, then the G*plus* Adapter does not change the agent state after login or when an agent has a a call on a the agent Teleset. It is possible that some switches may change the agent's state by distributing T-Events according to the switch configuration and functionality.

- If the AgentWorkMode parameter is equal to the AgentManualIn parameter, then (if the agent state is Ready) the Adapter sets an agent state to Not Ready after login or after an the agent answers a call on the agent's Teleset. This scenario happens only if the switch and T-Server are allowed to set the agent to the NotReady state when the agent has a call on the DN.

- If the AgentWorkMode parameter is equal to the AgentAutoIn parameter, then (if the agent state is Not Ready) the Adapter changes an agent state to Ready immediately after login and after a call is released. The Adapter changes the state only if the state after login was NotReady, or if the agent state on the call was NotReady.

For the switches that do not support the emulation of the work modes, the value of the Service:AgentWorkMode configuration parameter must be set to either "CHANGE_ME" or "AgentWorkModeUnknown". The Ericsson MD-110, the Siemens HiPath DX [previously known as Siemens Realtis-DX (iSDX)], and the Aspect Call Center are examples of such switches.

Agent States on Telesets with Multiple Positions and ACD Queues

Some switches, such as the Ericsson MD-110, support Telesets with multiple positions and ACD Queues.

In such environments, an agent can be logged into multiple ACD Queues. Also, before logging into Siebel, an agent can be in different and even opposite states on different Positions residing on the same Teleset.

Siebel cannot support and maintain the agent state, the DN forwarding status, and the DoNotDisturb (DND) state individually for each DN for such switches.

For such switches, the agent state, the DN forwarding status, and the DoNotDisturb (DND) status are maintained by the G*plus* Adapter only for the whole Teleset.

The general rules for this environment are as follows:

- An agent is considered to be in a Ready(NotReady) state if the agent is in the Ready(NotReady) state on at least one of the agent's Teleset DNs.
- The forwarding state of the agent's Teleset is considered to be "ON" if forwarding is set on one of the agent's Teleset DNs.
- The DoNotDisturb (DND) state of the agent's Teleset is considered to be "ON" if DND is set on one of the agent's Teleset DNs.

If, before logging into Siebel, an agent has a different Ready states on different DNs that belong to the same Teleset, then, after logging into Siebel, the agent can synchronize their state on all Teleset DNs by using the Ready/NotReady commands on the Communications toolbar, which is generally recommended. When you perform this synchronization of agent states on different DNs, you may get an error message such as "Set is in target state". This indicates that you were already in the requested state on some DN or DNs on this Teleset.

If, before logging into Siebel, an agent has a DN forwarding state set on any DN that belongs to the same Teleset, then the DN forwarding state on all Teleset DNs can be synchronized, which is generally recommended. After

logging into Siebel, you can do this by invoking the ForwardCall or CancelForwardCall device commands on the Communications toolbar.

If, before logging into Siebel, an agent has set DoNotDisturb (DND) states on any DN that belongs to the same Teleset, then the DoNotDisturb (DND) states on all Teleset DNs can be synchronized, which is generally recommended. After logging into Siebel, the agent can do this by invoking the SetBusy or CancelBusy device commands on the Communications toolbar.

The Voice Component Log Files

All G*plus* Adapter for Siebel CRM Voice Component log messages are now processed by the Genesys G*plus* Communication Server. The file names of the log files are defined in the Log section of the Genesys G*plus* Communication Server application in Genesys Configuration Manager. For more details on the configuration of the log file output, see "Log Section" on page 57 of Chapter 7, "Deploying the Communication Server for Siebel CRM."

The Voice Component Driver Commands and Events

This section provides a complete list of all driver commands and events for the Voice Component, including the Outbound Contact feature and the Expert Contact feature. For additional information, refer to the Deployment Guides and the Developer's Guides for the related products.

The Voice Component Driver Commands

Table 23 contains a complete list of all of the driver commands for the Voice Component. These commands may be used in the Siebel Communications Configuration as device commands for communication commands.

Note: Changes for the 8.0.1 release are represented in the subsection "The Voice Component Driver Commands (Changes for 8.0.1)" on page 303.

Table 23: The Voice Component Driver Commands

Name	Component/ Feature	Description
AnswerCall	Voice	Answer incoming call.
ReleaseCall	Voice	Release call.

Table 23: The Voice Component Driver Commands (Continued)

Name	Component/ Feature	Description
TransferSStep	Voice	Single-step transfer of the selected call.
TransferMute	Voice	Mute transfer of the selected call.
TransferInit	Voice	Initiate two-step transfer.
TransferComplete	Voice	Complete two-step transfer.
ReconnectCall (previously, RetrieveCall)	Voice	Reconnects the original call initiated using ConferenceInit or TransferInit before ConferenceComplete or TransferComplete.
HoldCall	Voice	Places the selected call on hold.
ResumeCall	Voice	Reconnects a call that was on hold.
ResumeInactiveCall	Voice	Reconnects a callback that was on hold.
MakeCall	Voice	Make an outbound call.
ConferenceSStep	Voice	Initiate a single-step conference.
ConferenceInit	Voice	Initiate a two-step conference.
ConferenceComplete	Voice	Complete a two-step conference.
DeleteFromConference	Voice	Deletes a party from a conference call.
LogIn	Voice	Logs in an agent.
LogOut	Voice	Logs out an agent.
Ready	Voice	Switches an agent's readiness state to Ready on all agent DNs if agent's state was NotReady.
NotReady	Voice	Switches an agent's readiness state to NotReady on all agent DNs if the agent's state was Ready.
ChangeNotReadyState	Voice	Switches the agent's readiness state from Ready to NotReady or from NotReady to Ready on all agent DNs.
SetActiveCall	Voice	Notify driver that currently selected call has changed.
ForwardCall	Voice	Initiates call forwarding.
CancelForwardCall	Voice	Cancels call forwarding.

Table 23: The Voice Component Driver Commands (Continued)

Name	Component/ Feature	Description
ToggleForward	Voice	Turns call forwarding on or off.
SetBusy	Voice	Sets the Do Not Disturb mode for all of an agent's DNs.
CancelBusy	Voice	Cancels the Do Not Disturb mode for all of an agent's DNs.
ChangeBusyState	Voice	Turns the Do Not Disturb mode on or off.
AttachData	Voice	Attaches the user data to the selected call.
SendUserEvent	Voice	Sends the user event to the selected DN.
UpdateUserData	Voice	Updates the user data that is attached to the selected call.
DeleteUserData	Voice	Removes the specified user data from the attached data for the selected call.
QueryAddress	Voice	Sends a query to T-Server to return information about a specified DN and ACD Queue.
SendUserEventOnCommDN	Voice	Sends a user event to a Communication DN.
CallSupervisor	Voice	When the agent is on an ACD call, this command initiates a conference leg to the agent's supervisor. The supervisor must push the Answer Agent key to complete the call leg to the supervisor.
EmergencyKey	Voice	When the agent is on an ACD call, this command initiates a conference leg to the agent's supervisor and to a recording device, if such a device is installed. The supervisor must push the Answer Emergency key to establish the call leg.
Expert Contact Feature		
KwOnCall	Expert Contact	Informs the CTI-less T-Server that the expert is on call.
KwPreviewIntReject	Expert Contact	Rejects incoming Preview interaction.

Table 23: The Voice Component Driver Commands (Continued)

Name	Component/ Feature	Description
KwPartyStatusResponse	Expert Contact	Agent provides the current call status in response to the KwPartyStatusRequest request. Configurable on the CTI-less T-Server side.
Outbound Contact Feature		
PreviewDialingModeStart	Outbound Contact	Starts the Preview dialing mode.
PreviewDialingModeOver	Outbound Contact	Ends the Preview dialing mode.
PreviewRecordRequest	Outbound Contact	Requests a Preview record.
UpdateCallCompletionStats	Outbound Contact	Updates the call statistics and values.
RecordProcessed	Outbound Contact	Completes the record processing.
RecordReject	Outbound Contact	Rejects a record.
RequestRecordCancel	Outbound Contact	Cancels a record for a particular campaign.
RecordReschedule	Outbound Contact	Reschedules a record.
ScheduleRecordReschedule	Outbound Contact	Reschedules a record (same as RecordReschedule).
DoNotCall	Outbound Contact	Marks the phone, record, chain, or customer as Do Not Call.
ChainedRecordRequest	Outbound Contact	Requests the rest of the chain.
AddRecord	Outbound Contact	Adds a record to the chain.
Voice Callback Feature		
VCBServiceStatus	Voice Callback	Starts the Voice Callback session.
VCBCancel	Voice Callback	Cancels the Voice Callback session and deletes the request from Voice Callback server.
VCBReschedule	Voice Callback	Reschedules the Voice Callback session.
VCBAdd	Voice Callback	Creates a new Voice Callback session and submits for processing.
VCBProcessed	Voice Callback	Marks the Callback record as done.
VCBReject	Voice Callback	Rejects the Callback record for rerouting to other agents.

The Voice Component Driver Commands (Changes for 8.0.1)

Table 24 contains a list of changed/added driver commands for the Voice Component. These commands may be used in the Siebel Communications Configuration as device commands for communication commands.

Table 24: The Voice Component Driver Commands (Changes for 8.0.1)

Name	Component/Feature	Description
SetActiveCall	Voice	Notifies the driver that the call has been selected.
DeselectActiveCall	Voice	Notifies the driver that the recent active call has been deselected.

Support for the Emergency/Supervisor Key

The G*plus* Adapter 8.0 for Siebel CRM has the capability to create a no-hold conference call leg to a supervisor or emergency DN. This feature is supported only on the Nortel Communication Server 1000 SCCS/MLS Symposium switch, which is a functional hybrid of the Meridian 1 and the Symposium switch.

With a properly customized Communications toolbar, an agent can press a button to initiate this special type of conference request. By doing so, the agent issues a TPrivateService request to initiate a no-hold conference call. The no-hold conference call allows the agent to dial-in and conference without putting the customer on hold.

Invoking the CallSupervisor feature when the agent is in an ACD call initiates a conference leg to the agent's supervisor. The supervisor must click the Answer Agent button on the phone, which establishes that call leg to the supervisor. Invoking the EmergencyKey feature initiates a conference leg to the agent's supervisor, and, if installed, to a recording device. The acting supervisor must answer the call by clicking the Answer Emergency button on the supervisor's phone set.

Note: This command is supported by the Adapter; but the customer must create a custom button to implement this feature.

This new functionality includes the following two device commands, CallSupervisor and EmergencyKey, to invoke the corresponding functions. These functions are described below.

Note: Both device commands (CallSupervisor and EmergencyKey) can be invoked without parameters. The TrackingID parameter can be defined in the [CmdData:] section of Command. For details about this functionality, see "The TrackingID Parameter" on page 305.

The CallSupervisor DeviceCommand Object

Invoking the CallSupervisor DeviceCommand object when an agent is in an automatic call distribution (ACD) call initiates a conference leg to the agent's supervisor. The supervisor must press the Answer Agent button, which establishes that call leg to the supervisor. For details about the CallSupervisor configuration, refer to Example 17 below.

If the CallSupervisor service is invoked:

- If the agent releases the call then the call is released from the agent DN. The call between the contact's and the supervisor's phones continues.

- If the customer releases the call, an EventReleased message is received on the agent DN, but the call between the agent's and supervisor's phones continues.

- The calls through the ACD Queue are not distributed to the agent.

- After this call is released, the ACD Queue continues to distribute calls to the agent DN.

Example 17 displays a CallSupervisor DeviceCommand object configuration sample:

Example 17   [Command:CallSupervisor]

;   For Nortel Meridian CallCenterSymposium ONLY!

  DeviceCommand   = "CallSupervisor"

  Description   = "Call Supervisor request"

  CmdData   = "CallSupervisor"

  Hidden   = "TRUE"

  CmdChannelOnFocus = "TRUE"

  Profile   = "Gplus Universal Profile"

             [CmdData:CallSupervisor]
             Param.TrackingID   = "{@SelectedWorkItem:DriverWorkTrackID}"

The EmergencyKey DeviceCommand Object

Invoking the EmergencyKey DeviceCommand object initiates a conference leg to the agent's supervisor, and, if installed, to a recording device. The acting supervisor must answer the call by pressing the Answer Emergency button on the supervisor's phone set. For details about the EmergencyKey DeviceCommand object configuration, refer to Example 18 below.

If the EmergencyKey service is invoked:

- If the agent releases the call, then the call is released on the supervisor's phone, the contact's phone, and the agent's phone;

- If the contact releases the call, there is no computer-telephony integration (CTI) messaging, and the call between the agent and supervisor's phones continues. After releasing this call, an EventReleased message is received on the agent DN.

Example 18 displays a EmergencyKey DeviceCommand object configuration sample:

Example 18

```
[Command:EmergencyKey]
;    For Nortel Meridian CallCenterSymposium ONLY!
  DeviceCommand    = "EmergencyKey"
  Description       = "Emergency request"
  CmdData           = "EmergencyKey"
  Hidden            = "TRUE"
  CmdChannelOnFocus = "TRUE"
  Profile           = "Gplus Universal Profile"


            [CmdData:EmergencyKey]
                    Param.TrackingID       =
"{@SelectedWorkItem:DriverWorkTrackID}"
```

The TrackingID Parameter

Both the CallSupervisor and EmergencyKey DeviceCommand objects can be invoked without parameters. The TrackingID parameter can be defined in the [CmdData:] section of Command, or it can be omitted. If the TrackingID parameter is defined in the [CmdData:] section, then the G*plus* Adapter uses this value to identify the call to which the CallSupervisor or EmergencyKey DeviceCommand object is applicable. If the Parameter TrackingID parameter is not defined, or if Siebel Communication Server supplies the empty string value for this parameter, then the CallSupervisor or EmergencyKey DeviceCommand objects are called for the current call.

The DoNotDisturb Status

The DoNotDisturb (DND) status can be set to ON or OFF on the DN with the following DeviceCommand objects: SetBusy, CancelBusy, and ChangeBusyState.

When the DND status is established for a DN, then the calls are not distributed to or received by this DN.

Some switches and T-Servers (Avaya INDeX, for example) do not provide this information about the DND status when they register the DNs. When these switches are used, if a DND state was set on the DN to ON, and then an agent logs into Siebel, the Communications toolbar indicates that the DND state is OFF (the default value). In order to synchronize the DND state on the switch for this DN with the DND state shown on the Siebel Communications toolbar, the agent must use the phone to switch off the DND state.

When the G*plus* Adapter works with these switches, the agent must perform the same synchronization of the DND state if a T-Server restarts after the DND state was set to ON on the switch.

The Driver Events for the Voice Component

This section describes the device events implemented by the Voice Component. A device event is a notification of a communications occurrence that the Siebel client receives from the Communications driver (the G*plus* Adapter in this case) along with some data fields (parameters).

For example, when a connection between the agent's phone and the destination is established for an incoming or outbound call, the Siebel client can be notified with the EventEstablished device event with the associated data (parameters), such as OtherDN, ANI and others shown in Table 26, "The Parameters Used with the Device Events of the Voice Component," on page 310. For further information, see the *Siebel Communications Server Administration Guide.*

The parameters used with the device events are listed in Table 26 on page 310. Descriptions of the device events begin with the topic, "The Voice Component Device Events" on page 348.

Table 25 contains the description of the Communications driver events. These events may be used in the Siebel Communications configuration as device events for communications events.

Table 25: Communications Driver Events

Name	Feature/ Configuration	Description
Voice Component		
EventDialing	Voice	Dialing notification when making a call.

Table 25: Communications Driver Events (Continued)

Name	Feature/ Configuration	Description
EventRinging	Voice	Incoming call ringing.
EventEstablished	Voice	Call established (inbound or outbound).
EventReleased	Voice	Call released or abandoned.
EventHeld	Voice	Call is held.
EventRetrieved	Voice	Call is retrieved.
EventUserDataChanged	Voice	Call user data changed.
EventRegistered	Voice	DN is registered.
EventUnregistered	Voice	DN is unregistered.
EventCallForwardSet	Voice	Call forwarding is set.
EventCallForwardCancel	Voice	Call forwarding is canceled.
EventAgentBusy	Voice	Do Not Disturb mode has been set.
EventAgentNotBusy	Voice	Do Not Disturb mode has been canceled.
EventAgentLogin	Voice	Agent is logged in.
EventAgentLogout	Voice	Agent is logged out.
EventAgentReady	Voice	Agent became ready.
EventAgentNotReady	Voice	Agent became not ready.
EventServerConnected	Voice	Connection to T-Server is established.
EventServerDisconnected	Voice	Connection to T-Server is lost.
EventError	Voice	Error received from T-Server.
EventACK	Voice	T-Server has acknowledged a request received from a client application. The event is a response to a TSendUserEvent event.
EventUserEvent	Voice	User event from another client application.
EventPartyChanged	Voice	Call party was replaced by another party.
EventPartyAdded	Voice	Party was added to the conference call.

Table 25: Communications Driver Events (Continued)

Name	Feature/ Configuration	Description
EventPartyDeleted	Voice	Party was deleted from the conference call.
EventPartyInfo	Voice	Information about call parties.
Expert Contact Feature		
EventKwPreviewIntError	Expert Contact	Preview Interaction error.
EventKwPreviewRequest	Expert Contact	Preview interacting request.
EventKwPreviewIntCancel	Expert Contact	Preview interaction canceled by T-Server.
EventKwPreviewIntAccepted	Expert Contact	Response from T-Server that the interaction was accepted.
EventKwPreviewIntRejected	Expert Contact	Response from T-Server that the interaction was rejected.
EventKwError	Expert Contact	CTI-Less T-Server error.
EventKwAck	Expert Contact	T-Server response.
EventKwOnCallResponse	Expert Contact	Response from CTI-Less T-Server after agent clicks On Call.
EventKwPartyStatusRequest	Expert Contact	Request from CTI-Less T-Server to provide current call status.
EventKwPartyStatusAck	Expert Contact	Acknowledge that party status request was submitted to CTI-less T-Server.
Outbound Contact Feature		
EventAddRecordAck	Outbound Contact	Record was added.
EventCampaignLoaded	Outbound Contact	Campaign was loaded.
EventCampaignModeChanged	Outbound Contact	Campaign mode was changed.
EventCampaignStarted	Outbound Contact	Campaign has started.
EventCampaignStopped	Outbound Contact	Campaign has stopped.
EventCampaignUnloaded	Outbound Contact	Campaign was unloaded.
EventChainedRecord	Outbound Contact	Chained record delivered.
EventChainedRecordDataEnd	Outbound Contact	Whole chain delivered.

Table 25: Communications Driver Events (Continued)

Name	Feature/ Configuration	Description
EventChainedWorkItemChanged	Outbound Contact	Chained work item in the Work Items list changed.
EventCurrentWorkItemChanged	Outbound Contact	Work item in the Work Items list changed.
EventOCSError	Outbound Contact	Outbound Contact Server error.
EventScheduleCall	Outbound Contact	Scheduled record delivered.
EventPreviewModeOverAck	Outbound Contact	Preview Dialing session is over.
EventPreviewModeStartAck	Outbound Contact	Preview Dialing session has been started.
EventPreviewRecord	Outbound Contact	Preview record delivered.
EventPreviewRecordEmpty	Outbound Contact	No more records exists in the Outbound Contact Server database.
EventRecordProcessedAck	Outbound Contact	Record has been processed.
EventRecordRejectAck	Outbound Contact	Record has been rejected.
EventRecordCancelAck	Outbound Contact	Record has been canceled.
EventRecordRemove	Outbound Contact	Request to remove record from desktop.
EventRecordRescheduleAck	Outbound Contact	Record has been rescheduled.
EventSchRecordRescheduleAck	Outbound Contact	Scheduled record has been rescheduled (same as RecordRescheduledAcknowledge).
EventUpdCallComplStatsAck	Outbound Contact	Call statistics and values have been updated.
EventLogOutTime	Outbound Contact	Notification from Outbound Contact Server about the time left until an agent logs out in Predictive dialing mode.
EventLogOutAck	Outbound Contact	Automatic logout acknowledge.

The T-Library Events Parameters

Table 26 on page 310 lists the parameters of the T-Library events that are used by the Voice Component as parameters sent to Siebel when the event handler is invoked. Use this list in combination with the device events for the Voice Component topic under the following section, "The Voice Component Device Events" on page 348.

Table 26: The Parameters Used with the Device Events of the Voice Component

Parameter Name	Type	Description
ConnID	Mandatory	The ConnectionID attribute.
CallType	Mandatory	The call types are Incoming, Outbound, Internal, ConsultTransfer, ConsultConference, Consult or Unknown. See Table 27 on page 312 for a description of the call types.
ThisDN	Optional	The ThisDN attribute of the TEvent parameter (the agent's DN that received the call).
OtherDN	Optional	The OtherDN attribute of the TEvent parameter, the OriginationDN attribute in case of an internal call.
ThirdPartyDN	Optional	The ThirdPartyDN attribute of the TEvent parameter, origination of the transferred or conferenced call.
ANI	Optional	ANI (Automatic Number Identification).
DNIS	Optional	DNIS (Dialed Number Identification Service).
PreviousConnID	Optional	The previous ConnectionID attribute: the ConnectionID attribute of the held call, if any.
ErrorCode	Optional	The error code if the TEvent is an EventError event.
ErrorMessage	Optional	Error message if the TEvent is an EventError event.
AgentID	Optional	This parameter uniquely identifies the agent.
ThisQueue	Optional	The directory number of the most significant ACD Queue group in relation to a specific event.
CollectedDigits	Optional	The digits that have been collected by the device collecting the digits—for example, an IVR.
Server Version	Optional	The version (release number) of the running T-Server.
CustomerID	Optional	A pointer to the string containing the assigned customer (Tenant) identifier through which the processing of the call was initiated.
HomeLocation	Optional	A pointer to the name of the host where T-Server is running.
AccessNumber	Optional	A pointer to a number derived when a client application dials from the specified switch, and allows the user to reach a specific external Routing Point.

Table 26: The Parameters Used with the Device Events of the Voice Component (Continued)

Parameter Name	Type	Description
Place	Optional	The Place attribute of a TEvent.
ReferenceID	Optional	The ReferenceID attribute is a unique identifier generated by a T-Server client and is attached to the request this client sends to T-Server.
CallState	Optional	The current status of the call the event relates to.
WorkMode	Optional	An agent work mode.
ThisTrunk	Optional	The identifier or the most significant trunk in relation to a specific event.
ThisDNRole	Optional	The role of the telephony object specified by the ThisDN attribute.
MediaType	Optional	The MediaType attribute of the TEvent parameter.
All others	Optional	The call's user data.

The Call Types

The G*plus* Adapter reports the call types to Siebel using the CallType and CallTypeExt fields. The original information about the call types comes in the CallType attribute of the TEvent parameter. The G*plus* Adapter defines the CallTypeExt field based on the call type and call flow. In the CallType field, the Adapter sends integer values, while the CallTypeExt field sends values of type string. Table 27 lists the possible call types for the G*plus* Adapter.

Table 27: The Call Type Fields

CallType	CallTypeExt	Description
0 (for Unknown)	Unknown or any other CallTypeExt value	The call type is unknown. The Adapter cannot define the call type based on the call flow.
1 (for Internal)	InternalTo	An internal call—for example, agent-to-agent call.
1 (for Internal)	InternalFrom	An internal call—for example, agent-to-agent call.
1 (for Internal)	Internal	An agent is on a call with another agent. The Adapter cannot define who initiated the call.
1 (for Internal)	Conference	There are more than two parties on the call.
2 (for Inbound)	Inbound	The inbound call from a customer.
2 (for Inbound)	Conference	There are more than two parties on the call.
3 (for Outbound)	Outbound	The outbound call to a customer.
3 (for Outbound)	Conference	There are more than two parties on the call.
4 (for Consult)	ConsultTransfer	A consultation call when an agent tries to transfer a call to another agent.
4 (for Consult)	ConsultConference	A consultation call when an agent tries to initiate a conference call with another agent.
4 (for Consult)	ConsultTo	When an agent initiates a transfer call or a conference call by using the phone (not the Siebel Communications toolbar). When the Consult call type is used, the G*plus* Adapter is not notified by T-Server of the specific action in progress—for example, if it is a transfer or a conference call, or if an agent put the current call on hold and made another call.
4 (for Consult)	ConsultFrom	A consultation call to the agent's DN.

The Detailed Descriptions of the Device Commands

This section provides the detailed descriptions of the device commands that are supported by the G*plus* Adapter Voice Component. A device command is a feature of the Communications driver (G*plus* Adapter in this case) that can execute a particular function, which is implemented in a driver.

Usually the command is a request from the Siebel client to the Communications driver to make a call, accept a call, transfer a call, or make a conference with another agent and so on.

The following tables use the terms <datasetParam>, <AttributeExtensions>, and <Other parameters>, which are defined below:

- The term <datasetParam> is used in the tables to describe the set of parameters for a particular device command. The datasetParam parameter is a set of parameters that are defined in the command data with the prefix Param.

- The term <AttributeExtensions> designates a specific subset of the datasetParam parameter. This subset of the parameters is defined in the datasetParam parameter in a specific way. See the description of "The AttributeExtensions Parameter" on . If the AttributeExtensions parameter is defined in the datasetParam parameter, then these parameters are sent as an Extensions attribute with a request to T-Server invoked by this device command.

- The term <Other parameters> designates a subset of the datasetParam parameter. This subset of the parameters defined in the datasetParam parameter is included as an UserData attribute (with some exclusions) in the T-Server request. Any exclusions in these parameters are identified in the command descriptions.

The following commands are implemented in the G*plus* Adapter and may be used in the Siebel Communications configuration as device commands. For information about Siebel device commands, see the Siebel documentation.

The Basic Voice Device Commands

This section provides detailed descriptions of the Basic Voice feature device commands (See Table 28).

Note: Changes for the 8.0.1 release are represented in the subsection "The Voice Component Driver Commands (Changes for 8.0.1)" on .

Table 28: The Basic Voice Feature Device Commands

AnswerCall			
Description	Answers incoming call.		
stringParam	Not used		
datasetParam	TrackingID	Optional	The work item TrackingID. If not defined, the internally maintained TrackingID is used.
ReleaseCall			
Description	Releases (disconnects) current call.		
stringParam	Not used		
datasetParam	TrackingID	Optional	The work item TrackingID. If not defined, the internally maintained TrackingID is used.
	ReasonCode	Optional	The value of the ReasonCode is sent to T-Server as a reasons parameter. The default value is -1.
datasetParam	Attribute Extensions	Optional	The Extensions attribute is sent with a request to T-Server as described in the following sections: "The AttributeExtensions Parameter" on page 245 and "Sending Extensions Attributes with String and/or Integer Value Types (Method 1)" on page 246.
TransferMute			
Description	Performs a mute transfer of the selected call. The CallNofifyText parameter conveys status information to the second agent. The RemoteConnectStr parameter is required for transfers to another contact center. For the switches that do not support mute transfer, but do support a single-step transfer, the Adapter invokes the request to T-Server to perform a single-step transfer.		
stringParam	Destination number	Either the stringParam or PhoneNumber must be defined.	

Table 28: The Basic Voice Feature Device Commands (Continued)

datasetParam	TrackingID	Optional	The work item TrackingID. If not defined, the internally maintained TrackingID is used.
	PhoneNumber	Either PhoneNumber or stringParam must be defined.	The destination number. If not present, the stringParam is used.
	ReasonCode	Optional	The value of the ReasonCode is sent to the T-Server as a reasons parameter. The default value is -1.
	Attribute Extensions	Optional	The Extensions attribute is sent with a request to T-Server as described in the following sections:
			"The AttributeExtensions Parameter" on page 245 and "Sending Extensions Attributes with String and/or Integer Value Types (Method 1)" on page 246.
	<Other parameters>	Optional	Exclusion from the user data: RemoteConnectStr, PhoneNumber, ThisDN, Digits, MonitorType, ReasonCode, HeldCallID, AttributeExtensions, StringAttributeExtensions, IntAttributeExtensions
	<Other parameters>	Optional	And corresponding to the AttributeExtensions set of parameters, see the descriptions on Page 245 and Page 246.
TransferInit			
Description	Initiates a two-step transfer of the selected call. The caller is put on hold, and the current agent dials another agent's extension. The CallNofityText parameter conveys status information to the second agent. The RemoteConnectStr parameter is required for transfers to another contact center.		

Table 28: The Basic Voice Feature Device Commands (Continued)

stringParam	Destination number	Either stringParam or PhoneNumber must be defined.	
datasetParam	TrackingID	Optional	The work item TrackingID. If not defined, the internally maintained TrackingID is used.
	PhoneNumber	Either PhoneNumber or stringParam parameter must be defined.	The destination number. If not present, the stringParam parameter is used.
	ReasonCode	Optional	The value of the ReasonCode is sent to T-Server as a reasons parameter. The default value is -1.
	Attribute Extensions	Optional	The Extensions attribute is sent with a request to T-Server as described in the following sections: "The AttributeExtensions Parameter" on page 245 and "Sending Extensions Attributes with String and/or Integer Value Types (Method 1)" on page 246.
	<Other parameters>	Optional	Exclusions from the user data: RemoteConnectStr, PhoneNumber, ThisDN, Digits, MonitorType, ReasonCode, HeldCallID, AttributeExtensions, StringAttributeExtensions, IntAttributeExtensions And corresponding to the AttributeExtensions set of parameters, see the descriptions on Page 245 and Page 246.

Table 28: The Basic Voice Feature Device Commands (Continued)

TransferComplete			
Description	Completes a two-step transfer of the selected call. Completes the consultative transfer. The current agent is disconnected and the caller is connected to the agent to whom the call was transferred.		
stringParam	Not used		
datasetParam	TrackingID	Optional	The work item TrackingID. If not defined, the internally maintained TrackingID is used.
ReconnectCall			
Description	Reconnects the original call initiated using the ConferenceInit or TransferInit requests before the ConferenceComplete or TransferComplete requests.		
stringParam	Not used		
datasetParam	TrackingID	Optional	The work item TrackingID. If not defined, the internally maintained TrackingID is used.
HoldCall			
Description	Places the selected call on hold.		
stringParam	Not used		
datasetParam	TrackingID	Optional	The work item TrackingID. If not defined, the internally maintained TrackingID is used.
ReleaseCall			
Description	Releases (disconnects) current call.		
stringParam	Not used		

Table 28: The Basic Voice Feature Device Commands (Continued)

datasetParam	TrackingID	Optional	The work item TrackingID. If not defined, the internally maintained TrackingID is used.
	ReasonCode	Optional	The value of the ReasonCode is sent to T-Server as a reasons parameter. The default value is -1.
	Attribute Extensions	Optional	The Extensions attribute is sent with a request to T-Server as described in the following sections: "The AttributeExtensions Parameter" on page 245 and "Sending Extensions Attributes with String and/or Integer Value Types (Method 1)" on page 246.
ResumeCall			
Description	Reconnects a call that was on hold.		
stringParam	Not used		
datasetParam	TrackingID	Optional	The work item TrackingID. If not defined, the internally maintained TrackingID is used.
ResumeInactiveCall			
Description	Reconnects a callback that was on hold.		
stringParam	Not used		
datasetParam	TrackingID	Optional	The work item TrackingID. If not defined, the internally maintained TrackingID is used.
MakeCall			
Description	Places an outbound call. The CallNotifyText parameter is used only for internal calls.		
stringParam	Destination number	Either the stringParam or PhoneNumber parameter must be defined.	

Table 28: The Basic Voice Feature Device Commands (Continued)

datasetParam	TrackingID	Optional	The work item TrackingID. If not defined, the internally maintained TrackingID is used.
	PhoneNumber	Either the PhoneNumber or stringParam must be defined.	The destination number. If not present, the stringParam parameter is used instead.
	ReasonCode	Optional	The value of the ReasonCode is sent to T-Server as a reasons parameter. The default value is -1.
	Attribute Extensions	Optional	The Extensions attribute is sent with a request to T-Server as described in the following sections: "The AttributeExtensions Parameter" on page 245 and "Sending Extensions Attributes with String and/or Integer Value Types (Method 1)" on page 246.
	<Other parameters>	Optional	Exclusions from the user data: RemoteConnectStr, PhoneNumber, ThisDN, Digits, MonitorType, ReasonCode, HeldCallID, AttributeExtensions, StringAttributeExtensions, IntAttributeExtensions And corresponding to the AttributeExtensions set of parameters, see the descriptions on Page 245 and Page 246.
ConferenceSStep			
Description	Begin a single-step conference. The CallNofityText parameter conveys status information to the second agent. The RemoteConnectStr parameter is required for conferences between contact centers.		
stringParam	Destination number	Either stringParam or PhoneNumber must be defined.	

Table 28: The Basic Voice Feature Device Commands (Continued)

datasetParam	TrackingID	Optional	The work item TrackingID. If not defined, the internally maintained TrackingID is used.
	PhoneNumber	Either the PhoneNumber or stringParam parameter must be defined.	The destination number. If not present, the stringParam parameter is used instead.
	ReasonCode	Optional	The value of the ReasonCode is sent to T-Server as a reasons parameter. The default value is -1.
	Attribute Extensions	Optional	The Extensions attribute is sent with a request to T-Server as described in the following sections: "The AttributeExtensions Parameter" on page 245 and "Sending Extensions Attributes with String and/or Integer Value Types (Method 1)" on page 246.
	\<Other parameters\>	Optional	Exclusions from the user data: RemoteConnectStr, PhoneNumber, ThisDN, Digits, MonitorType, ReasonCode, HeldCallID, AttributeExtensions, StringAttributeExtensions, IntAttributeExtensions And corresponding to the AttributeExtensions set of parameters, see the descriptions on Page 245 and Page 246.
ConferenceInit			
Description	Initiates a two-step conference. The caller is put on hold, and the current agent dials another agent's extension. The CallNofityText parameter conveys the status information to the second agent. The RemoteConnectStr parameter is required for conferences between contact centers.		
stringParam	Destination number	Either stringParam or PhoneNumber must be defined.	

Table 28: The Basic Voice Feature Device Commands (Continued)

datasetParam	TrackingID	Optional	The work item TrackingID. If not defined, the internally maintained TrackingID is used
	PhoneNumber	Either the PhoneNumber or stringParam must be defined.	The work item TrackingID. If not defined, the internally maintained TrackingID is used.
	ReasonCode	Optional	The value of the ReasonCode is sent to T-Server as a reasons parameter. The default value is -1.
	Attribute Extensions	Optional	The Extensions attribute is sent with a request to T-Server as described in the following sections: "The AttributeExtensions Parameter" on page 245 and "Sending Extensions Attributes with String and/or Integer Value Types (Method 1)" on page 246.
	<Other parameters>	Optional	Exclusions from the user data: RemoteConnectStr, PhoneNumber, ThisDN, Digits, MonitorType, ReasonCode, HeldCallID, AttributeExtensions, StringAttributeExtensions, IntAttributeExtensions And corresponding to the AttributeExtensions set of parameters, see the descriptions on Page 245 and Page 246.
ConferenceComplete			
Description	Completes a two-step conference. The caller and the agents in conference can now talk to each other at the same time.		
stringParam	Not used		
datasetParam	TrackingID	Optional	The work item TrackingID. If not defined, the internally maintained TrackingID is used.

Table 28: The Basic Voice Feature Device Commands (Continued)

DeleteFromConference			
Description	Deletes a party that is specified by the PhoneNumber parameter from a conference call. If the PhoneNumber parameter is not defined, then the stringParam parameter is used instead.		
stringParam	Destination number	Either the stringParam or PhoneNumber parameters must be defined.	Defines the party to be deleted from a conference call.
datasetParam	TrackingID	Optional	The work item TrackingID. If not defined, the internally maintained TrackingID is used.
	PhoneNumber	Either the stringParam or PhoneNumber parameters must be defined.	Defines the party to be deleted from a conference call. If not present, the stringParam parameter is used instead.
	ReasonCode	Optional	The value of the ReasonCode is sent to T-Server as a reasons parameter. The default value is -1.
	Attribute Extensions	Optional	The Extensions attribute is sent with a request to T-Server as described in the following sections: "The AttributeExtensions Parameter" on page 245 and "Sending Extensions Attributes with String and/or Integer Value Types (Method 1)" on page 246.
LogIn			
Description	Logs in an agent to one or multiple queues. If an agent's Teleset contains multiple DNs, the agent is only logged into the first DN.		
stringParam	Not used		

Table 28: The Basic Voice Feature Device Commands (Continued)

datasetParam	ACDQueue	Optional	List of ACD Queue numbers, separated by commas.
	AgentID	Mandatory	AgentID
	AgentPin	Optional	Agent password
	AgentWork Mode	Optional	The workmode that is to be set for an agent. AgentWorkMode is sent to T-Server as a workmode parameter. The default value is 0 (zero).
	ReasonCode	Optional	The value of the ReasonCode is sent to T-Server as a reasons parameter. The default value is -1.
	Attribute Extensions	Optional	The Extensions attribute is sent with a request to T-Server as described in the following sections: "The AttributeExtensions Parameter" on page 245 and "Sending Extensions Attributes with String and/or Integer Value Types (Method 1)" on page 246.
LogOut			
Description	Logs out an agent from one or all of the queues. If the agent's teleset contains multiple DNs, the agent will only be logged out of the first DN.		
stringParam	Not used		
datasetParam	ReasonCode	Optional	The value of the ReasonCode is sent to T-Server as a reasons parameter. The default value is -1.

Table 28: The Basic Voice Feature Device Commands (Continued)

	Attribute Extensions	Optional	The Extensions attribute is sent with a request to T-Server as described in the following sections: "The AttributeExtensions Parameter" on page 245 and "Sending Extensions Attributes with String and/or Integer Value Types (Method 1)" on page 246.
	ACDQueue	Optional	The ACD Queue from which an agent logs out.
Ready			
Description	Switches an agent's readiness state to NotReady on all agent DNs if an agent's state was Ready.		
stringParam	Not used		
datasetParam	AgentWork Mode	Optional	The workmode that is to be set for the agent. AgentWorkMode is sent to T-Server as a workmode parameter. The default value is 0 (zero).
	ReasonCode	Optional	The value of the ReasonCode is sent to T-Server as a reasons parameter. The default value is -1.
	Attribute Extensions	Optional	The Extensions attribute is sent with a request to T-Server as described in the following sections: "The AttributeExtensions Parameter" on page 245 and "Sending Extensions Attributes with String and/or Integer Value Types (Method 1)" on page 246.
	ACDQueue	Optional	A list of ACD Queue numbers, separated by commas.
NotReady			
stringParam	Not used		

Table 28: The Basic Voice Feature Device Commands (Continued)

datasetParam	AgentWork Mode	Optional	The workmode that is to be set for the agent. AgentWorkMode is sent to T-Server as a workmode parameter. The default value is 0 (zero).
	ReasonCode	Optional	The value of the ReasonCode is sent to T-Server as a reasons parameter. The default value is -1.
	Attribute Extensions	Optional	The Extensions attribute is sent with a request to T-Server as described in the following sections: "The AttributeExtensions Parameter" on page 245 and "Sending Extensions Attributes with String and/or Integer Value Types (Method 1)" on page 246.
	ACDQueue	Optional	A list of ACD Queue numbers, separated by commas.
ChangeNotReadyState			
Description	Switches an agent's readiness state from Ready to NotReady or from NotReady to Ready on all agent DNs.		
stringParam	Not used		
datasetParam	AgentWork Mode	Optional	The workmode that is to be set for the agent. AgentWorkMode is sent to T-Server as a workmode parameter. The default value is 0 (zero).
	ReasonCode	Optional	The value of the ReasonCode will be sent to T-Server as a reasons parameter. The default value is -1.

Table 28: The Basic Voice Feature Device Commands (Continued)

	Attribute Extensions	Optional	The Extensions attribute is sent with a request to T-Server as described in the following sections: "The AttributeExtensions Parameter" on page 245 and "Sending Extensions Attributes with String and/or Integer Value Types (Method 1)" on page 246.
	ACDQueue	Optional	A list of ACD Queue numbers, separated by commas.
SetActiveCall			
Description	Notifies the driver that a call, currently selected in the Communications toolbar, has been changed. The driver updates all the command statuses for the newly selected call.		
stringParam	Not used		
datasetParam	TrackingID	Mandatory	The work item TrackingID attribute.
ForwardCall			
Description	Initiates call forwarding.		
stringParam	Destination number	Either the stringParam or PhoneNumber parameter must be defined.	
datasetParam	PhoneNumber	Either PhoneNumber or stringParam must be defined.	The destination number. If not present, the stringParam parameter is used.
	Mode	Optional	The Call-Forwarding mode (numeric value). The default value is 1.
CancelForwardCall			
Description	Cancels call forwarding.		
stringParam	Not used		
datasetParam	Mode	Optional	The Call-Forwarding mode (numeric value). The default value is 1.

Table 28: The Basic Voice Feature Device Commands (Continued)

ToggleForward			
Description	Turns call forwarding on or off.		
stringParam	Destination number	Either stringParam or PhoneNumber must be defined.	
datasetParam	PhoneNumber	Either the PhoneNumber or stringParam parameter must be defined.	The destination number. If not present, the stringParam parameter is used instead.
	Mode	Optional	The Call-Forwarding mode (numeric value). The default value is 1.
SetBusy			
Description	Initiates the Do Not Disturb mode for all DNs an agent is logged into.		
stringParam	Not used		
datasetParam	Not used		
CancelBusy			
Description	Cancels the Do Not Disturb mode.		
stringParam	Not used		
datasetParam	Not used		
ToggleForward			
Description	Turns call forwarding on or off.		
stringParam	Destination number	Either the stringParam or PhoneNumber parameter must be defined.	

Table 28: The Basic Voice Feature Device Commands (Continued)

datasetParam	PhoneNumber	Either the PhoneNumber or stringParam parameter must be defined.	The destination number. If not present, the stringParam parameter is used instead.
	Mode	Optional	The Call-Forwarding mode (numeric value). The default value is 1.
ChangeBusyState			
Description	Turns the Do Not Disturb mode on or off.		
stringParam	Not used		
datasetParam	Not used		
AttachData			
Description	Attaches the user data to the selected call. Note that the user data name *TrackingID* is invalid.		
stringParam	N/A	N/A	
datasetParam	TrackingID	Optional	The work item TrackingID. If not defined, the internally maintained TrackingID is used.
	\<Other parameters\>	Optional	Exclusion from the user data: PhoneNumber

Table 28: The Basic Voice Feature Device Commands (Continued)

SendUserEvent	
Description	Sends a user event to the selected DN. The selected DN is either the currently selected extension in the Communications view in the User Preferences screen or taken from the stringParam parameter.
	Can be configured to send AgentID and ConnID attributes in the related SendUserEvent. To configure this, the CmdData section of the SendUserEvent device command must include parameters Attribute.AgentID and Attribute.ConnID, for example:
	[Command:SendUserEvent] Hidden = "TRUE" Description = "Send user event" DeviceCommand = "SendUserEvent" CmdData = "SendUserEvent"
	[CmdData:SendUserEvent] AttachContext = "TRUE" Param.ThisDN = "{@SelectedDN}" Param.Attribute.AgentID = "Attribute_AgentID" Param.Attribute.ConnID = "12340d0000000000"
	The Attribute.ConnID value is a 16 digit hexadecimal that defines the ConnID.
	If it is required to set the Attribute.AgentID and Attribute.ConnID parameters in the UserData of the UserEvent, then a backslash (\) must be used as an escape symbol before the parameter name.
	The backslash (\) directs the Adapter to send a parameter with the name beginning after the backslash as UserData with a SendUserEvent request. A single backslash without a string is treated as a UserData parameter with key \, for example:
	[CmdData:SendUserEvent] AttachContext = "TRUE" Param.ThisDN = "{@SelectedDN}" Param.Attribute.AgentID = "Attribute_AgentID" Param.Attribute.ConnID = "12340d0000000000" Param.\Attribute.AgentID = "UserData_Attribute.AgentID" Param.\Attribute.ConnID = "UserData_84321" Param.\ = "This_Is_\" Param.AgentID = "UserData_AgentID"
	Then the UserEvent includes the attributes, AgentID and ConnID: UserData; (Str) \ This_Is_\ (Str) Attribute.ConnID UserData_84321 (Str) Attribute.AgentID UserData_Attribute.AgentID (Str) AgentID UserData_AgentID

Table 28: The Basic Voice Feature Device Commands (Continued)

stringParam	Selected DN (see "Description")	Optional	
datasetParam	TrackingID	Optional	The work item TrackingID. If not defined, the internally maintained TrackingID is used.
	<Other parameters>	Optional	Exclusion from the user data: ThisDN
UpdateUserData			
Description	Updates the user data that is attached to the selected call. **Note:** The user data name TrackingID is invalid.		
stringParam	N/A	N/A	
datasetParam	TrackingID	Optional	The work item TrackingID. If not defined, the internally maintained TrackingID is used.
	<Other parameters>	Optional	Exclusion from the user data: PhoneNumber
DeleteUserData			
Description	Removes user data, specified by <Other parameters> from the user data attached to the selected call. **Note:** The user data name TrackingID is invalid.		
stringParam	N/A	N/A	
datasetParam	TrackingID	Optional	The work item TrackingID. If not defined, the internally maintained TrackingID is used.
	<Other parameters>	Optional	Exclusion from the user data: PhoneNumber Note: Parameters must not have an empty value. For example: Param.TEST_OPTION = "1"
CallSupervisor			
Description	When the agent is on an ACD call, this command initiates a conference leg to the agent's supervisor. For that call leg to be established, the supervisor must push the Answer Agent key to complete the call leg to the supervisor. For more details, see "The CallSupervisor DeviceCommand Object" on page 304.		
stringParam	Not Used		

Table 28: The Basic Voice Feature Device Commands (Continued)

datasetParam	TrackingID	Either PhoneNumber or stringParam must be defined.	If the Siebel Communication Session Manager invokes a particular command with an empty tracking ID, the G*plus* Adapter will try to use the value of the Param.TrackingID defined in the Command Data.
	\<Other parameters\>	Optional	Exclusion from the user data: AttributeExtensions, StringAttributeExtensions, IntAttributeExtensions And corresponding to the AttributeExtensions set of parameters, see the descriptions on Page 245 and Page 246.
EmergencyKey			
Description	Initiates a conference leg to the agent's supervisor and, if installed, to a recording device. The supervisor must push the "Answer Emergency" key to establish the call leg. For more details, see "The EmergencyKey DeviceCommand Object" on page 305.		
stringParam	Not Used		
datasetParam	TrackingID	{@SelectedWork Item:DriverWork TrackID}	If the Siebel Communication Session Manager invokes a particular command with an empty tracking ID, the G*plus* Adapter will try to use the value of Param.TrackingID defined in the Command Data.
	\<Other parameters\>	Optional	Exclusion from the user data: AttributeExtensions, StringAttributeExtensions, IntAttributeExtensions And corresponding to the AttributeExtensions set of parameters, see descriptions on Page 245 and Page 246.

Table 28: The Basic Voice Feature Device Commands (Continued)

QueryAddress			
Description	Sends a query to T-Server to return the information about the DN specified by the command parameters, ThisDN and ACDQueue. If the ThisDN parameter is not defined in the configuration, the default configured DN is used instead.The type of information requested is defined by the command parameters, AddressType and AddressInfoType.		
stringParam	Not Used		
datasetParam	ThisDN	Optional	T-Server inquires about the telephony object specified by the ThisDN parameter. If not present, the default configured DN (Teleset of type **S**) is used instead.
	ACDQueue	Optional	The list of ACD Queue numbers, separated by commas
	AddressType (AddressType Position)	Optional	The type of telephony object in question. The default value is 2.
	AddressInfo Type	Optional	The type of the requested information. The default value is 17.
	Attribute Extensions	Optional	The Extensions attribute is sent with a request to T-Server as described in the following sections: "The AttributeExtensions Parameter" on page 245 and "Sending Extensions Attributes with String and/or Integer Value Types (Method 1)" on page 246.

*The QueryAddress device command together with the set of command data parameters that are defined in a sample GenComm_universal.def file, synchronize the state of the Communications toolbar and the state of the agent on a Lucent Definity G3 switch. For more information about other switches and how to define the corresponding command data parameters for the QueryAddress device command, see the T-Server documentation related to the switch you are using.

SendUserEventOnCommDN	
Description	Sends a user event to the Communication DN.

Table 28: The Basic Voice Feature Device Commands (Continued)

stringParam	Destination number	Optional	
datasetParam	TrackingID	Optional	The work item TrackingID. If not defined, the internally maintained TrackingID is used.
	\<Other parameters\>	Optional	Exclusion from the user data: No exclusions
AlternateCall			
Description	Places the active call on hold and reconnects the previously held call.		
stringParam			
datasetParam	ActiveCallID	Optional	The active work item TrackingID. If not defined, the internally maintained TrackingID is used.
	HeldCallID	Mandatory	The TrackingID of the work item that is currently on hold.
Send DTMF			
Description	Sends the digits that are interpreted by an interactive voice response system.		
stringParam			
datasetParam	TrackingID	Optional	The work item TrackingID. If not defined, the internally maintained TrackingID is used.
	Digits	Mandatory	The digits that are sent.
	ReasonCode	Optional	The value of the ReasonCode that is sent to T-Server as a reasons parameter. The default value is -1.
	Attribute Extensions	Optional	The Extensions attribute is sent with a request to T-Server as described in the following sections: "The AttributeExtensions Parameter" on page 245 and "Sending Extensions Attributes with String and/or Integer Value Types (Method 1)" on page 246.

Table 28: The Basic Voice Feature Device Commands (Continued)

Monitor Next Call			
Description	Enables the supervisor's monitoring of the next call that comes to the agent. The supervisor's DN is either the currently selected extension in the Communications view in the User Preferences screen (the selected DN) or taken from the ThisDN parameter.		
stringParam		Either the stringParam or PhoneNumber parameter must be defined.	
datasetParam	TrackingID	Optional	The work item TrackingID. If not defined, the internally maintained TrackingID is used.
	ThisDN	Optional	The supervisor's DN. If not present, the selected DN is used.
	PhoneNumber	Either PhoneNumber or stringParam must be defined.	The agent's DN that is monitored. If not present, the stringParam is used.
	Monitor Type	Optional	Indicates whether the supervisor wants to monitor one call or subsequent calls. The default value is -1. Other possible values are: 0 = MonitorOneCall 1 = MonitorAllCalls
	ReasonCode	Optional	The value of the ReasonCode that is sent to T-Server as a reasons parameter. The default value is -1.
	Attribute Extensions	Optional	The Extensions attribute is sent with a request to T-Server as described in the following sections: "The AttributeExtensions Parameter" on page 245 and "Sending Extensions Attributes with String and/or Integer Value Types (Method 1)" on page 246.

Table 28: The Basic Voice Feature Device Commands (Continued)

Cancel Monitoring			
Description	Cancels the supervisor's monitoring of the next call that comes to the agent. The supervisor's DN is either the currently selected extension in the Communications view in the User Preferences screen (the selected DN) or taken from the ThisDN parameter.		
stringParam		Either the stringParam or PhoneNumber parameter must be defined.	
datasetParam	TrackingID	Optional	The work item TrackingID. If not defined, the internally maintained TrackingID is used.
	ThisDN	Optional	The supervisor's DN. If not present, the selected DN is used.
	PhoneNumber	Either the stringParam or PhoneNumber parameter must be defined.	The agent's DN that is monitored. If not present, the stringParam parameter is used.
	Monitor Type	Optional	Indicates whether the supervisor wants to monitor one call or subsequent calls. The default value is -1. Other possible values are: 0 = MonitorOneCall 1 = MonitorAllCalls
	ReasonCode	Optional	The value of the ReasonCode that is sent to T-Server as a reasons parameter. The default value is -1.
	Attribute Extensions	Optional	The Extensions attribute is sent with a request to T-Server as described in the following sections: "The AttributeExtensions Parameter" on page 245 and "Sending Extensions Attributes with String and/or Integer Value Types (Method 1)" on page 246.

Table 28: The Basic Voice Feature Device Commands (Continued)

SetActiveCampaign			
Description	Sets the active campaign for the Preview mode.		
stringParam		Either the stringParam or PhoneNumber parameter must be defined.	The name of the campaign in the Preview mode as it is defined in Configuration Manager.
datasetParam	CampaignName	Either the stringParam or PhoneNumber parameter must be defined.	The name of the campaign in Preview mode as it is defined in Configuration Manager.

The Basic Voice Device Commands (Changes for 8.0.1)

Table 29 provides the detailed descriptions of the Basic Voice feature device commands that were changed or added for the 8.0.1 release.

Table 29: The Basic Voice Device Commands (Changes for 8.0.1)

SetActiveCall			
Description	Notifies the driver that a call has been selected in the Communications Toolbar. The driver updates all of the command statuses for the newly selected call.		
stringParam	Not used		
datasetParam	TrackingID	Mandatory	The work item TrackingID.
DeselectActiveCall			
Description	Notifies the driver that the recent call has been deselected in the Communications Toolbar. The driver updates all of the command statuses for the newly selected call.		
stringParam	Not used		
datasetParam	SelectedTrackingID	Mandatory	The TrackingID of the selected workitem.
	DeselectedTrackingID	Mandatory	The TrackingID of the deselected workitem.

The Expert Contact DeviceCommand Objects

This section lists the following Expert Contact feature's DeviceCommand objects:

- KwOnCall
- KwPreviewIntReject
- KwPartyStatusResponse

For more details, contact Genesys Customer Care.

The Outbound Contact DeviceCommands Objects

This section provides the detailed descriptions of the Outbound Campaign feature's DeviceCommand objects:

- PreviewDialingModeStar
- PreviewDialingModeOver
- PreviewRecordRequest
- UpdateCallCompletionStats
- RecordProcessed
- RecordReject
- RequestRecordCancel
- RecordReschedule
- ScheduledRecordReschedule
- DoNotCall
- ChainedRecordRequest
- AddRecord

The mandatory fields GSW_APPLICATION_ID and GSW_CAMPAIGN_NAME are assigned internally by the Outbound Contact Server (OCS) protocol instance, depending on the mode and campaigns that are currently running.

The Outbound Contact Server DeviceCommand objects are described in Table 30.

Table 30: The Outbound Contact Server DeviceCommand Objects

PreviewDialingModeStar	
Command Name	PreviewDialingModeStart
Description	Request to activate a preview session for the agent. (This command is needed if the value of the OCS agent_preview_mode_start configuration option is set to true.)

Table 30: The Outbound Contact Server DeviceCommand Objects (Continued)

OCS Action	Link the agent DN and the campaign ID.
Mandatory Fields	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME
Additional Fields	Not specified
PreviewDialingModeOver	
Command Name	PreviewDialingModeOver
Description	Request to terminate a preview session for the agent.
OCS Action	Remove a link between agent DN and campaign ID.
Mandatory Fields	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME
Additional Fields	Not specified
PreviewRecordRequest	
Command Name	PreviewRecordRequest
Description	Request to send a preview record.
OCS Action	If the value of the OCS agent_preview_mode_start configuration option is set to false, or there is a link between the agent DN and the campaign ID.
Mandatory Fields	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME
Additional Fields	Not specified
UpdateCallCompletionStats	
Command Name	UpdateCallCompletionStats
Description	Desktop sends to update record details. Intermediate update.
OCS Action	Update the record fields internally and wait for the next requests.

Table 30: The Outbound Contact Server DeviceCommand Objects (Continued)

Mandatory Fields	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME GSW_RECORD_HANDLE
Additional Fields	Can be both specified updatable Genesys fields and user defined fields.
RecordProcessed	
Command Name	RecordProcessed
Description	Desktop sends an event to indicate that the record is processed. OCS should update the record information, if it is provided.
OCS Action	Update a record and its chain in the database; use all the changes made by the previous requests regarding the records in the chain. If a RecordProcessed event has a GSW_TREATMENT field correctly specified, OCS applies a treatment to the record.
Mandatory Fields	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME GSW_CALLING_LIST GSW_RECORD_HANDLE
Additional Fields	Can be both specified updatable Genesys fields and user-defined fields.
RecordReject	
Command Name	RecordReject
Description	Desktop sends a request to indicate that the preview record or the scheduled callback is not dialed by this agent. The record should be re-sent to another agent.
OCS Action	Mark a record as Ready and the rest of a chain if GSW_CHAIN_ATTR = = "AllChain".
Mandatory Fields	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME GSW_CALLING_LIST GSW_RECORD_HANDLE

Table 30: The Outbound Contact Server DeviceCommand Objects (Continued)

Additional Fields	Not specified
RequestRecordCancel	
Command Name	RequestRecordCancel
Description	Desktop sends a request to indicate that a preview record or a scheduled callback should not be dialed. Record should not be re-sent to another agent. Should be marked in the database as Canceled.
OCS Action	Change the record

Mandatory Fields	GSW_APPLICATION_ID
	
	if (GSW_RECORD_HANDLE is specified){
	GSW_CAMPAIGN_NAME
	GSW_CALLING_LIST
	}
	else
	{
	GSW_PHONE
	GSW_CAMPAIGN_NAME (optional)
	}

Additional Fields	GSW_CHAIN_ATTR = "AllChain" Or GSW_CHAIN_ATTR = "RecordOnly"
RecordReschedule	
Command Name	RecordReschedule
Description	Request a rescheduled preview record, predictive call, or scheduled call.
OCS Action	Update a record chain and reschedule the record.
Mandatory Fields	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME GSW_RECORD_HANDLE GSW_DATE_TIME GSW_CALLBACK_TYPE

Table 30: The Outbound Contact Server DeviceCommand Objects (Continued)

Additional Fields	Not specified
ScheduledRecordReschedule	
Command name	ScheduledRecordReschedule
Description	Request a reschedule preview record, predictive call, or scheduled call.
OCS Action	Update a record chain and reschedule the record.
Mandatory Fields	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME GSW_RECORD_HANDLE GSW_DATE_TIME GSW_CALLBACK_TYPE
Additional Fields	Not specified
DoNotCall	
Command Name	DoNotCall
Description	Agent requests this number not to be called again.
OCS Action	If GSW_RECORD_HANDLE is specified, update a record or a whole chain as DoNotCall if GSW_CHAIN_ATTR == "AllChain". Add phone(s) (GSW_CHAIN_ATTR == "AllChain") in a DoNotCall list.

Table 30: The Outbound Contact Server DeviceCommand Objects (Continued)

Mandatory Fields	GSW_APPLICATION_ID
	
	1) if (GSW_RECORD_HANDLE is specified){
	GSW_CAMPAIGN_NAME
	GSW_CALLING_LIST
	}
	else
	{
	2) GSW_PHONE
	}
	else
	{
	3) GSW_CUSTOMER_ID
	}

Additional Fields	Not specified
Comments	You can specify these attributes in the Siebel.def file as command parameters. A specific set of mandatory fields is sent to Outbound Contact Server: 1. USE_RECORD_HANDLE 2. USE_PHONE 3. USE_CUSTOMER_ID Tells the Adapter which set of mandatory fields are sent to Outbound Contact Server.
ChainedRecordRequest	
Command Name	ChainedRecordRequest
Description	Request to send all records from the chain defined by record handle.
OCS Action	Send the rest of a chain to the desktop.
Mandatory Fields	GSW_AGENT_REQ_TYPE GSW_APPLICATION_ID GSW_CAMPAIGN_NAME GSW_RECORD_HANDLE
Additional Fields	Not specified

Table 30: The Outbound Contact Server DeviceCommand Objects (Continued)

AddRecord	
Command Name	AddRecord
Description	Request to add record to the particular list.
OCS Action	Verify data and create new record in the list.
Mandatory Fields	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME GSW_CALLING_LIST GSW_PHONE GSW_TZ_NAME
Additional Fields	Any user-defined fields. **Note:** The key in a key-value pair should be equal to the send_attribute of the field that you are required to add. The default field values (if not specified in a request) are as follows: GSW_PHONE_TYPE: 2, "DirectBusinessPhone" GSW_RECORD_TYPE: 2, "General" GSW_RECORD_TYPE: 2, "General" GSW_RECORD_STATUS: 1, "Ready" GSW_CALL_RESULT: 28, "Unknown" GSW_ATTEMPTS: 0 GSW_DATE_TIME: 0 GSW_CALL_TIME: 0 GSW_FROM: 28800 (8 AM) GSW_UNTIL: 64800 (6 PM) GSW_AGENT_ID: 0

The Voice Callback DeviceCommands Objects

This section provides the detailed descriptions of the Voice Callback feature DeviceCommand objects (see Table 31).

Table 31: The VCB Commands

Gplus Adapter Name	Genesys VCB Server Name
VCBServiceStatus	RequestCallbackServiceStatus
VCBQuery	RequestCallbackQuery
VCBQueryResult	RequestCallbackQueryResult
VCBCancel	RequestCallbackCancel
VCBReschedule	RequestCallbackReschedule
VCBAdd	RequestCallbackAdd
VCBPreview	RequestCallbackPreview
VCBProcessed	RequestCallbackProcessed
VCBReject	RequestCallbackReject

Table 32: The Voice Callback DeviceCommands Objects

VCBServiceStatus	
Command Name	VCBServiceStatus (RequestCallbackServiceStatus)
Description	Client sends this request to the VCB server to determine the availability of the Voice Callback service.
Callback Server Action	Sends back the corresponding response with the requested information.
Mandatory Fields	VCB_ORIGIN_APPLICATION_ID VCB_REFERENCE_ID VCB_STATUS: 0
Additional Fields	VCB_ROUTING_POINT VCB_LOCATION
VCBQuery	

Table 32: The Voice Callback DeviceCommands Objects (Continued)

Command Name	VCBQuery (RequestCallbackQuery)
Description	Client sends this request to the VCB server to find the callback requests that satisfy the search conditions.
Callback Server Action	Sends back the corresponding UserEvent (CallbackQueryAcknowledge) with the query result (number of callback requests found).
Mandatory Fields	VCB_ORIGIN_APPLICATION_ID VCB_TARGET_APPLICATION_ID VCB_REFERENCE_ID At least one of the following: VCB_CONTACT VCB_TYPE VCB_ROUTING_POINT VCB_LOCATION
Additional Fields	Any fields that have to be included in the search conditions.
VCBQueryResult	
Command Name	VCBQueryResult (RequestCallbackQueryResult)
Description	Client sends this request to the VCB server to retrieve the callback request by indexing in the query result set.
Callback Server Action	Sends the callback request back to the client (CallbackQueryResult).
Mandatory Fields	VCB_ORIGIN_APPLICATION_ID VCB_TARGET_APPLICATION_ID VCB_REFERENCE_ID VCB_QUERY_INDEX
Additional Fields	Not specified
VCBCancel	

Table 32: The Voice Callback DeviceCommands Objects (Continued)

Command Name	VCBCancel (RequestCallbackCancel)
Description	Client sends this request to the VCB server to cancel the callback request.
Callback Server Action	Deletes the callback request and sends back a confirmation (CallbackCancelAcknowledge).
Mandatory Fields	VCB_ORIGIN_APPLICATION_ID VCB_TARGET_APPLICATION_ID VCB_REFERENCE_ID VCB_RECORD_HANDLE (or VCB_CONTACT)
Additional Fields	Not specified
VCBReschedule	
Command Name	VCBReschedule (RequestCallbackReschedule)
Description	Client sends this request to the VCB server to reschedule a callback request.
Callback Server Action	Reschedules a callback request and sends back confirmation (CallbackRescheduleAcknowledge). The callback is rescheduled to ASAP (as soon as possible) if the VCB_DATE_TIME key does not exist in the UserData attribute.
Mandatory Fields	VCB_ORIGIN_APPLICATION_ID VCB_TARGET_APPLICATION_ID VCB_REFERENCE_ID VCB_RECORD_HANDLE VCB_CALL_RESULT
Additional Fields	VCB_DATE_TIME (if scheduled) VCB_TZ_NAME (if scheduled)
VCBAdd	
Command Name	VCBAdd (RequestCallbackAdd)
Description	Client sends this request to add a callback request.

Table 32: The Voice Callback DeviceCommands Objects (Continued)

Callback Server Action	Creates a callback request object. Processes as a scheduled callback request (immediately, if set to ASAP).
Mandatory Fields	VCB_ORIGIN_APPLICATION_ID VCB_TARGET_APPLICATION_ID VCB_REFERENCE_ID VCB_CONTACT VCB_TYPE: 1, "ASAP" VCB_ORIGIN VCB_ROUTING_POINT VCB_LOCATION
Additional Fields	Any user defined fields and default field values (if not specified in a request): VCB_DATE_TIME (if scheduled) VCB_TZ_NAME (if scheduled)
VCBPreview	
Command Name	VCBPreview (RequestCallbackPreview)
Description	The VCB server sends a callback request to the agent's desktop.
Desktop Action	Sends back acknowledgement.
Mandatory Fields	VCB_ORIGIN_APPLICATION_ID VCB_TARGET_APPLICATION_ID = 0 (if Desktop) VCB_REFERENCE_ID VCB_RECORD_HANDLE VCB_CONTACT VCB_CALL_RESULT All UserData attributes from Callback Request
Additional Fields	Any user defined fields
VCBProcessed	
Command Name	VCBProcessed (RequestCallbackProcessed)

Table 32: The Voice Callback DeviceCommands Objects (Continued)

Description	The agent's desktop submits the record processing results to the VCB server.
Callback Server Action	Sends back acknowledgement. Applies treatment if specified in the configuration.
Mandatory Fields	VCB_ORIGIN_APPLICATION_ID = 0 (if Desktop) VCB_TARGET_APPLICATION_ID VCB_REFERENCE_ID VCB_RECORD_HANDLE VCB_CALL_RESULT
Additional Fields	Any user defined fields
VCBReject	
Command Name	VCBReject (RequestCallbackReject)
Description	An agent's desktop rejects the record.
Callback Server Action	Sends back acknowledgement. Submits a callback request to another agent.
Mandatory Fields	VCB_ORIGIN_APPLICATION_ID = 0 (if Desktop) VCB_TARGET_APPLICATION_ID VCB_REFERENCE_ID VCB_RECORD_HANDLE
Additional Fields	Any user defined fields

The Voice Component Device Events

This section provides the detailed descriptions of the device events for the Voice Component including the Basic Voice feature and the Outbound Campaign feature.

The Basic Voice Device Events

Table 33 on provides descriptions of the device events for the Basic Voice feature.

Table 33: The Basic Voice Device Events

EventDialing	
Siebel CRM DeviceEvent Name	EventDialing
Desktop Action	Displays dialed call information.
TEvent Name	EventDialing
Description	An attempt to make a call on behalf of the telephony object specified by the ThisDN parameter is in progress.
TrackingID	TrackingID of the work item.
Attributes	See Table 26, "The Parameters Used with the Device Events of the Voice Component," on page 310.
EventRinging	
Siebel CRM DeviceEvent Name	EventRinging
Desktop Action	Displays ringing call information.
TEvent Name	EventRinging
Description	An attempt to make a call on behalf of the telephony object specified by the ThisDN parameter is in progress.
TrackingID	TrackingID of the work item.
Attributes	See Table 26, "The Parameters Used with the Device Events of the Voice Component," on page 310.
EventEstablished	
Siebel CRM DeviceEvent Name	EventEstablished
Desktop Action	Displays received call information
TEvent Name	EventEstablished
Description	The connection has been established for an incoming or an outbound call.
TrackingID	TrackingID of the work item.

Table 33: The Basic Voice Device Events (Continued)

Attributes	See Table 26, "The Parameters Used with the Device Events of the Voice Component," on page 310.
EventReleased	
Siebel CRM DeviceEvent Name	EventReleased
Desktop Action	Removes the related information from the work item or removes the related work item.
TEvent Name	EventReleased
Description	A call was released.
TrackingID	TrackingID of the work item.
Attributes	See Table 26, "The Parameters Used with the Device Events of the Voice Component," on page 310.
EventHeld	
Siebel CRM DeviceEvent Name	EventHeld
Desktop Action	Enables the Resume Work Item button and disables the Pause Work Item button (command)
TEvent Name	EventHeld
Description	A call was placed on hold.
TrackingID	TrackingID of the work item.
Attributes	See Table 26, "The Parameters Used with the Device Events of the Voice Component," on page 310.
EventRetrieved	
Siebel CRM DeviceEvent Name	EventRetrieved
Desktop Action	Disables the Resume Work Item button and enables the Pause Work Item button (command)
TEvent Name	EventRetrieved
Description	A call that was on hold has been retrieved.
TrackingID	Not applicable.

Table 33: The Basic Voice Device Events (Continued)

Attributes	See Table 26, "The Parameters Used with the Device Events of the Voice Component," on page 310.
EventUserDataChanged	
Siebel CRM DeviceEvent Name	EventUserDataChanged
Desktop Action	N/A
TEvent Name	EventAttachedDataChanged
Description	The call's user data has been changed.
TrackingID	TrackingID of the work item.
Attributes	See Table 26, "The Parameters Used with the Device Events of the Voice Component," on page 310.
EventRegistered	
Siebel CRM DeviceEvent Name	EventRegistered
Desktop Action	N/A
TEvent Name	EventRegistered
Description	The agent's DN was registered with T-Server.
TrackingID	Not applicable.
Attributes	ThisDN, Mandatory, RegisteredDN
EventUnregistered	
Siebel CRM DeviceEvent Name	EventUnregistered
Desktop Action	N/A
TEvent Name	EventUnregistered
Description	The agent's DN was unregistered with T-Server.
TrackingID	Not applicable.
Attributes	ThisDN, Mandatory, UnregisteredDN
EventCallForwardSet	
Siebel CRM DeviceEvent Name	EventCallForwardSet

Table 33: The Basic Voice Device Events (Continued)

Desktop Action	Disables the Set Forward button and enables the Cancel Forward button
TEvent Name	EventForwardSet
Description	Call forwarding has been set to on.
TrackingID	Not applicable.
Attributes	ThisDN, Mandatory. DN on which the call forwarding has been set.
EventCallForwardCancel	
Siebel CRM DeviceEvent Name	EventCallForwardCancel
Desktop Action	Disables the Cancel Forward button and enables the Set Forward button
TEvent Name	EventForwardCancel
Description	Call forwarding has been set to off.
TrackingID	Not applicable.
Attributes	ThisDN, Mandatory, DN on which the call forwarding has been canceled.
EventAgentBusy	
Siebel CRM DeviceEvent Name	EventAgentBusy
Desktop Action	N/A
TEvent Name	EventDNDOn
Description	The Do Not Disturb mode has been set to on.
TrackingID	Not applicable.
Attributes	ThisDN, Mandatory. DN on which the DND has been set.
EventAgentNotBusy	
Siebel CRM DeviceEvent Name	EventAgentNotBusy
Desktop Action	N/A
TEvent Name	EventDNDOff
Description	The Do Not Disturb mode has been set to off.

Table 33: The Basic Voice Device Events (Continued)

TrackingID	Not applicable.
Attributes	ThisDN, Mandatory. DN on which the DND has been canceled.
EventAgentLogin	
Siebel CRM DeviceEvent Name	EventAgentLogin
Desktop Action	Disables the Login to Voice button and enables the Logout from Voice button (command)
TEvent Name	EventAgentLogin
Description	Agent has been logged in.
TrackingID	Not applicable.
Attributes	ThisDN, Mandatory. DN on which agent was logged in AgentID, Mandatory, AgentID
EventAgentLogout	
Siebel CRM DeviceEvent Name	EventAgentLogout
Desktop Action	Disables the Logout from Voice button and enables the Login to Voice button (command)
TEvent Name	EventAgentLogout
Description	Agent has been logged out.
TrackingID	Not applicable.
Attributes	ThisDN, Mandatory. DN on which agent was logged out AgentID, Mandatory, AgentID
EventAgentReady	
Siebel CRM DeviceEvent Name	EventAgentReady
Desktop Action	Disables the Set Ready for Voice button (command).
TEvent Name	EventAgentReady
Description	Agent has been set to the Ready state.
TrackingID	Not applicable.

Table 33: The Basic Voice Device Events (Continued)

Attributes	ThisDN, Mandatory. DN on which the agent became ready.
	AgentID, Mandatory, AgentID
	Workmode, Mandatory, Agent's work mode
EventAgentNotReady	
Siebel CRM DeviceEvent Name	EventAgentNotReady
Desktop Action	Enables the Set Ready for Voice button (command).
TEvent Name	EventAgentNotReady
Description	Agent has been set to the Not Ready state.
TrackingID	Not applicable.
Attributes	ThisDN, Mandatory. DN on which the agent became not ready.
	AgentID, Mandatory, AgentID
	Workmode, Mandatory, Agent's work mode
EventServerConnected	
Siebel CRM DeviceEvent Name	EventServerConnected
Desktop Action	The Communications toolbar becomes active.
TEvent Name	EventLinkConnected
Description	The connection with T-Server has been established.
TrackingID	Not applicable.
Attributes	None
EventServerDisconnected	
Siebel CRM DeviceEvent Name	EventServerDisconnected
Desktop Action	The Communications toolbar becomes inactive.
TEvent Name	EventLinkDisconnected
Description	The connection with T-Server has been terminated.
TrackingID	Not applicable.
Attributes	None

Table 33: The Basic Voice Device Events (Continued)

EventError	
Siebel CRM DeviceEvent Name	EventError
Desktop Action	Displays an error in the status text area for a few seconds.
TEvent Name	EventError
Description	The EventError message has been received from T-Server.
TrackingID	TrackingID of the work item, if applicable.
Attributes	ErrorCode
EventUserEvent	
Siebel CRM DeviceEvent Name	EventUserEvent
Desktop Action	N/A
TEvent Name	EventUserEvent
Description	A user event from another client application has been received.
TrackingID	Not applicable
Attributes	The T-Event UserData attribute is received in Siebel as ISC_KVParamList.
EventPartyChanged	
Siebel CRM DeviceEvent Name	EventPartyChanged
Desktop Action	Shows the changes in the displayed work item.
TEvent Name	EventPartyChanged
Description	The telephony object specified by the OtherDN attribute has replaced the telephony object specified by the OtherDN attribute in the previously received T-Server event or the previous ConnID of the call has been given a new ConnID value.
TrackingID	TrackingID of the work item, if applicable.
Attributes	See Table 26, "The Parameters Used with the Device Events of the Voice Component," on page 310.

Table 33: The Basic Voice Device Events (Continued)

EventPartyAdded	
Siebel CRM DeviceEvent Name	EventPartyAdded
Desktop Action	Shows changes in the displayed work item
TEvent Name	EventPartyAdded
Description	The telephony object specified by the OtherDN attribute has been added to the conference call.
TrackingID	TrackingID of the work item, if applicable.
Attributes	See Table 26, "The Parameters Used with the Device Events of the Voice Component," on page 310.
EventPartyDeleted	
Siebel CRM DeviceEvent Name	EventPartyDeleted
Desktop Action	Shows changes in the displayed work item
TEvent Name	EventPartyDeleted
Description	The telephony object specified by the OtherDN attribute has been deleted from the conference call.
TrackingID	TrackingID of the work item, if applicable.
Attributes	See Table 26, "The Parameters Used with the Device Events of the Voice Component," on page 310.

The Expert Contact Device Events

This section lists the device events for the Expert Contact feature, as follows:

- EventKwPreviewIntError
- EventKwPreviewRequest
- EventKwPreviewIntCancel
- EventKwPreviewIntAccepted
- EventKwPreviewIntRejected
- EventKwError
- EventKwAck
- EventKwOnCallResponse
- EventKwPartyStatusRequest

- EventKwPartyStatusAck

For more details about these events, contact Genesys Customer Care.

The Outbound Campaign/Outbound Contact Server Device Events

This section provides detailed descriptions of device events for the Outbound Campaign feature, which is associated with the Genesys Outbound Contact Server (OCS) product.

The following list is the Outbound Contact Server (OCS) device events for the G*plus* Outbound Campaign feature. This section provides the details for each of these events in Table 34 on below.

- EventAddRecordAck
- EventCampaignGroupAssigned
- EventCampaignLoaded
- EventCampaignModeChanged
- EventCampaignStarted
- EventCampaignStopped
- EventCampaignUnloaded
- EventChainedRecord
- EventChainedRecordDataEnd
- EventChainedWorkItemChanged
- EventCurrentWorkItemChanged
- EventOCSError
- EventRecordCancel
- EventScheduledCall
- EventPreviewModeOverAck
- EventPreviewModeStartAck
- EventPreviewRecord
- EventPreviewRecordEmpty
- EventRecordCancelAck
- EventRecordRejectAck
- EventRecordCancelAck
- EventRecordRemove
- EventRecordProcessedAck
- EventRecordRejectAck
- EventRecordRescheduleAck
- EventSchRecordRescheduleAck

- EventUpdCallComplStatsAckn
- EventLogOutTime
- EventLogOutAck

Table 34: Outbound Campaign/Outbound Contact Server Device Events

EventAddRecordAck	
Siebel CRM DeviceEvent Name:	EventAddRecordAck
OCS Protocol User Event:	AddRecordAcknowledge
Description	OCS sent insert request to the database (DB).
TrackingID	TrackingID of the work item
Desktop Action	Continue session.
Attributes	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME GSW_CALLING_LIST

EventCampaignGroupAssigned	
Siebel CRM DeviceEvent Name:	EventCampaignGroupAssigned
OCS Protocol User Event:	CampaignGroupAssigned
Description	Notification that an agent assignment was changed to a new campaign in auto dial mode.
TrackingID	TrackingID of the work item
Desktop Action	Displays Campaign Name as it is defined in Configuration Manager
Attributes	GSW_APPLICATION_ID GSW_CAMPAIGN_GROUP_NAME GSW_CAMPAIGN_GROUP_DESCRIPTION GSW_CAMPAIGN_NAME GSW_CAMPAIGN_DESCRIPTION

EventCampaignLoaded	
Siebel CRM DeviceEvent Name:	EventCampaignLoaded
OCS Protocol User Event:	CampaignLoaded
Description	Sent by OCS when a campaign is loaded. **Note:** This event is specific to OCS 6.5, Desktop Protocol 6.5 only.

Table 34: Outbound Campaign/Outbound Contact Server Device Events (Continued)

TrackingID	TrackingID of the work item
Desktop Action	Store application ID and campaign name.
Attributes	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME GSW_CAMPAIGN_DESCRIPTION
EventCampaignModeChanged	
Siebel CRM DeviceEvent Name:	EventCampaignModeChanged
OCS Protocol User Event:	CampaignModeChanged
Description	Should be sent when the mode of the running campaign is changed.
TrackingID	TrackingID of the work item
Desktop Action	In Preview Mode, the outbound traffic is usually more intensive.
Attributes	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME GSW_CAMPAIGN_MODE
EventCampaignStarted	
Siebel CRM DeviceEvent Name:	EventCampaignStarted
OCS Protocol User Event:	CampaignStarted
Description	Should be sent when the dialing of the campaign is started or resumed, or as a response of an event agent login, if the campaign is started.
TrackingID	TrackingID of the work item
Desktop Action	Store a campaign name and an application ID.
Attributes	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME GSW_CAMPAIGN_MODE GSW_CAMPAIGN_DESCRIPTION

Table 34: Outbound Campaign/Outbound Contact Server Device Events (Continued)

EventCampaignStopped	
Siebel CRM DeviceEvent Name:	EventCampaignStopped
OCS Protocol User Event:	CampaignStopped
Description	Should be sent when the dialing for a campaign is started or resumed, or as a response of an event agent login, if the campaign is started.
TrackingID	TrackingID of the work item
Desktop Action	Stop sending requests to a campaign.
Attributes	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME
EventCampaignUnloaded	
Siebel CRM DeviceEvent Name:	EventCampaignUnloaded
OCS Protocol User Event:	CampaignUnloaded
Description	Sent by OCS when a campaign is unloaded. **Note**: This event is specific to OCS 6.5, Desktop Protocol 6.5 only.
TrackingID	TrackingID of the work item
Desktop Action	Consider the campaign as not active.
Attributes	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME GSW_CAMPAIGN_DESCRIPTION
EventChainedRecord	
Siebel CRM DeviceEvent Name:	EventChainedRecord
OCS Protocol User Event:	ChainedRecord
Description	Chain record delivered.
TrackingID	TrackingID of the work item
Desktop Action	Continue the call work.

Table 34: Outbound Campaign/Outbound Contact Server Device Events (Continued)

Attributes	GSW_USER_EVENT
	GSW_APPLICATION_ID
	GSW_CAMPAIGN_NAME
	GSW_CALLING_LIST
	GSW_RECORD_HANDLE
	GSW_PHONE
	GSW_CALL_RESULT GSW_CHAIN_ID
	Genesys and user-defined fields having send attributes.
EventChainedRecordDataEnd	
Siebel CRM DeviceEvent Name:	EventChainedRecordDataEnd
OCS Protocol User Event:	ChainedRecordDataEnd
Description	All chain has been delivered.
TrackingID	TrackingID of the work item
Desktop Action	Continue the call work.
Attributes	GSW_APPLICATION_ID
	GSW_CAMPAIGN_NAME
	GSW_CALLING_LIST
	GSW_RECORD_HANDLE
	GSW_CHAIN_ID
EventChainedWorkItemChanged	
Siebel CRM DeviceEvent Name:	EventChainedWorkItemChanged
OCS Protocol User Event:	N/A
Description	G*plus* Adapter sends notification that the current work item has changed, and that the previous and the current work items belong to the same chain. Sent in response to the command, SetActiveCall.
TrackingID	TrackingID of the work item
Desktop Action	
Attributes	N/A

Table 34: Outbound Campaign/Outbound Contact Server Device Events (Continued)

EventCurrentWorkItemChanged	
Siebel CRM DeviceEvent Name:	EventCurrentWorkItemChanged
OCS Protocol User Event:	N/A
Description	*Gplus* Adapter sends notification that the current work item has changed. Sent in response to the command, SetActiveCall.
TrackingID	TrackingID of the work item
Desktop Action	
Attributes	N/A
EventOCSError	
Siebel CRM DeviceEvent Name:	EventOCSError
OCS Protocol User Event:	Error
Description	OCS sends notification about error on the server side.
TrackingID	TrackingID of the work item
Desktop Action	Continue the session.
Attributes	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME GSW_CAMPAIGN_NAME GSW_ERROR_NAME
EventRecordCancel	
Siebel CRM DeviceEvent Name:	EventRecordCancel
OCS Protocol User Event:	RecordCancel
Description	OCS sends an event to desktop to indicate that this record should not be dialed. Applicable to preview records and scheduled callbacks.
TrackingID	TrackingID of the work item
Desktop Action	Kill the record and the chain if RecordCancel contains GSW_CHAIN_ATTR = "AllChain".

Table 34: Outbound Campaign/Outbound Contact Server Device Events (Continued)

Attributes	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME GSW_CALLING_LIST GSW_RECORD_HANDLE	
EventScheduledCall		
Siebel CRM DeviceEvent Name:	EventScheduledCall	
OCS Protocol User Event:	ScheduledCall	
Description	OCS sends to agent to indicate that scheduled callback should be executed.	
TrackingID	TrackingID of the work item	
Desktop Action	Perform call work.	
Attributes	GSW_USER_EVENT GSW_APPLICATION_ID GSW_CAMPAIGN_NAME GSW_CALLING_LIST GSW_RECORD_HANDLE GSW_PHONE GSW_CALL_RESULT GSW_CALLBACK_TYPE Genesys and user-defined fields having send attributes.	
EventPreviewModeOverAck		
Siebel CRM DeviceEvent Name:	EventPreviewModeOverAck	
OCS Protocol User Event:	PreviewDialingModeOverAcknowledge	
Description	OCS accepts a desktop request to close the preview session.	
TrackingID	TrackingID of the work item	
Desktop Action	Desktop is unable to send requests to OCS and receive callbacks (see the OCS configuration option, agent_preview_mode_start).	
Attributes	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME	

Table 34: Outbound Campaign/Outbound Contact Server Device Events (Continued)

EventPreviewModeStartAck	
Siebel CRM DeviceEvent Name:	EventPreviewModeStartAck
OCS Protocol User Event:	PreviewDialingModeStartAcknowledge
Description	OCS accepts a desktop request to initiate the preview session.
TrackingID	TrackingID of the work item
Desktop Action	Desktop can send requests to OCS and receive callbacks (see the agent_preview_mode_start configuration option).
Attributes	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME
EventPreviewRecord	
Siebel CRM DeviceEvent Name:	EventPreviewRecord
OCS Protocol User Event:	PreviewRecord
Description	Preview record to dial
TrackingID	TrackingID of the work item
Desktop Action	Perform call work
Attributes	GSW_USER_EVENT GSW_APPLICATION_ID GSW_CAMPAIGN_NAME GSW_CALLING_LIST GSW_RECORD_HANDLE GSW_PHONE GSW_CALL_RESULT Genesys and user-defined fields having send attributes.
EventPreviewRecordEmpty	
Siebel CRM DeviceEvent Name:	EventPreviewRecordEmpty
OCS Protocol User Event:	PreviewRecordEmpty
Description	No more records in the OCS internal buffer.
TrackingID	TrackingID of the work item
Desktop Action	Try to send a request later.

Table 34: Outbound Campaign/Outbound Contact Server Device Events (Continued)

Attributes	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME
EventRecordCancelAck	
Siebel CRM DeviceEvent Name:	EventRecordCancelAck
OCS Protocol User Event:	RecordCancelAcknowledge
Description	OCS accepts a desktop request to cancel a record.
TrackingID	TrackingID of the work item
Desktop Action	Kill the record and the chain if RecordCancelAcknowledge contains GSW_CHAIN_ATTR == "AllChain".

Attributes	GSW_APPLICATION_ID
	
	if (GSW_RECORD_HANDLE is specified) {
	GSW_CAMPAIGN_NAME
	GSW_CALLING_LIST
	GSW_RECORD_HANDLE
	}
	else
	{
	GSW_PHONE
	}

EventRecordRejectAck	
Siebel CRM DeviceEvent Name:	EventRecordRejectAck
OCS Protocol User Event:	RecordRejectAcknowledge
Description	OCS accepts the PredictRejectRecord request.
TrackingID	TrackingID of the work item
Desktop Action	Kill the record and the chain if the RecordCancelAcknowledge event contains GSW_CHAIN_ATTR == "AllChain".
Attributes	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME GSW_CALLING_LIST GSW_RECORD_HANDLE

Table 34: Outbound Campaign/Outbound Contact Server Device Events (Continued)

EventRecordCancelAck	
Siebel CRM DeviceEvent Name:	EventRecordCancelAck
OCS Protocol User Event:	RecordCancelAcknowledge
Description	OCS accepts a desktop request to cancel a record.
TrackingID	TrackingID of the work item
Desktop Action	Kill the record and the chain if the RecordCancelAcknowledge event contains GSW_CHAIN_ATTR == "AllChain".

Attributes	GSW_APPLICATION_ID
	
	if (GSW_RECORD_HANDLE is specified) {
	GSW_CAMPAIGN_NAME
	GSW_CALLING_LIST
	GSW_RECORD_HANDLE
	}
	else
	{
	GSW_PHONE
	}

EventRecordRemove	
Siebel CRM DeviceEvent Name:	EventRecordRemove
OCS Protocol User Event:	N/A
Description	G*plus* Adapter sends a request to remove the record from the desktop.
TrackingID	TrackingID of the work item
Desktop Action	Removes the record from the desktop.
Attributes	N/A
EventRecordProcessedAck	
Siebel CRM DeviceEvent Name:	EventRecordProcessedAck
OCS Protocol User Event:	RecordProcessedAcknowledge
Description	OCS confirms that the record has been executed.

Table 34: Outbound Campaign/Outbound Contact Server Device Events (Continued)

TrackingID	TrackingID of the work item
Desktop Action	Kills the record and the chain if requested.
Attributes	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME GSW_CALLING_LIST GSW_RECORD_HANDLE
EventRecordRejectAck	
Siebel CRM DeviceEvent Name:	EventRecordRejectAck
OCS Protocol User Event:	RecordRejectAcknowledge
Description	OCS accepts the PreviewRejectRecord request.
TrackingID	TrackingID of the work item
Desktop Action	Kills the record and the chain if the RecordCancelAcknowledge event contains GSW_CHAIN_ATTR == "AllChain".
Attributes	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME GSW_CALLING_LIST GSW_RECORD_HANDLE
EventRecordRescheduleAck	
Siebel CRM DeviceEvent Name:	EventRecordRescheduleAck
OCS Protocol User Event:	RecordRescheduleAcknowledge
Description	Confirmation that the record was rescheduled.
TrackingID	TrackingID of the work item
Desktop Action	Continue call work.
Attributes	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME GSW_CALLING_LIST GSW_RECORD_HANDLE GSW_DATE_TIME GSW_CALLBACK_TYPE

Table 34: Outbound Campaign/Outbound Contact Server Device Events (Continued)

EventSchRecordRescheduleAck	
Siebel CRM DeviceEvent Name:	EventSchRecordRescheduleAck
OCS Protocol User Event:	ScheduledRecordRescheduleAcknowledge
Description	Confirmation that the record was rescheduled.
TrackingID	TrackingID of the work item
Desktop Action	Continue call work.
Attributes	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME GSW_CALLING_LIST GSW_RECORD_HANDLE GSW_DATE_TIME GSW_CALLBACK_TYPE
EventUpdCallComplStatsAckn	
Siebel CRM DeviceEvent Name:	EventUpdCallComplStatsAck
OCS Protocol User Event:	UpdateCallCompletionStatsAcknowledge
Description	OCS accepts a desktop request to update a record's fields.
TrackingID	TrackingID of the work item
Desktop Action	Continue call work.
Attributes	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME GSW_CALLING_LIST GSW_RECORD_HANDLE
EventLogOutTime	
Siebel CRM DeviceEvent Name:	EventLogOutTime
OCS Protocol User Event:	UpdateEventLogOutTime
Description	Response for a logout request.
TrackingID	TrackingID of the work item
Desktop Action	Displays time left until an automatic logout.

Table 34: Outbound Campaign/Outbound Contact Server Device Events (Continued)

Attributes	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME GSW_LOGOUT_TIME
EventLogOutAck	
Siebel CRM DeviceEvent Name:	EventLogOutAck
OCS Protocol User Event:	UpdateEventLogOutAcknowledge
Description	Automatic logout acknowledge.
TrackingID	TrackingID of the work item
Desktop Action	Logs out an agent. Displays an agent's status change.
Attributes	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME

The Voice Callback Device Events

This section provides the detailed descriptions of the device events for the Voice Callback configuration. The following list is the device events for the G*plus* Voice Callback configuration (see Table 35).

Table 35:   VCB Events

Gplus Adapter Name	Genesys VCB Server Name
EventAddVCBRequestt	RequestCallbackAdd
EventVCBServiceStatus	RequestCallbackServiceStatus
EventVCBQuery	RequestCallbackQuery
The Voice Callback Error Codes and Descriptions	RequestCallbackQueryResult
EventVCBCancel	RequestCallbackCancel
EventVCBReschedule	RequestCallbackReschedule
EventVCBPreview	RequestCallbackPreview
EventVCBProcessed	RequestCallbackProcessed

Table 35: VCB Events (Continued)

Gplus Adapter Name	Genesys VCB Server Name
EventVCBReject	RequestCallbackReject
The Voice Callback Error Codes and Descriptions	RequestCallbackError

Table 36: The Voice Callback Device Events

EventAddVCBRequest	
Gplus Adapter Event Name:	EventAddVCBRequest
VCB Protocol User Event:	AddCallbackRequest
Description	The VCB Server sends this response to the VCB Client to confirm that the request has been processed.
Client Action	Not specified
Attributes	VCB_ORIGIN_APPLICATION_ID VCB_TARGET_APPLICATION_ID VCB_REFERENCE_ID VCB_RESPONSE_TYPE = 0 "Acknowledge" VCB_ERROR_CODE VCB_ERROR_NAME
EventVCBServiceStatus	
Gplus Adapter Event Name:	EventVCBServiceStatus
VCB Protocol User Event:	RequestCallbackServiceStatus
Description	The VCB Server sends this response to the VCB Client to confirm the availability of the Voice Callback service.
Client Action	Perform the callback request processing.

Table 36: The Voice Callback Device Events (Continued)

Attributes	VCB_ORIGIN_APPLICATION_ID VCB_TARGET_APPLICATION_ID VCB_REFERENCE_ID VCB_STATUS: 0 or 1 If Routing Point was specified in request: VCB_RESPONSE_TYPE = 1 "SingleResult" If Routing Point was not specified: VCB_RESPONSE_TYPE = 2 "MultipleResult" or VCB_RESPONSE_TYPE = 3 "MultipleResultEnd" VCB_LOCATION VCB_ROUTING_POINT VCB_ROUTING_POINT_DESCR VCB_ERROR_CODE VCB_ERROR_NAME
EventVCBQuery	
G*plus* Adapter Event Name:	EventVCBQuery
VCB Protocol User Event:	RequestCallbackQuery
Description	The VCB Server sends this response back to the Client Server to acknowledge the query execution and to provide the number of callback requests in a result set.
Client Action	Sends back the UserEvent (CallbackQueryGetResult) to request the query result by index.
Attributes	VCB_ORIGIN_APPLICATION_ID VCB_TARGET_APPLICATION_ID VCB_REFERENCE_ID VCB_RESPONSE_TYPE = 0 "Acknowledge" VCB_QUERY_COUNT VCB_ERROR_CODE VCB_ERROR_NAME
EventVCBQueryResult	
G*plus* Adapter Event Name:	EventVCBQueryResult
VCB Protocol User Event:	RequestCallbackQueryResult

Table 36: The Voice Callback Device Events (Continued)

Description	The VCB Server sends to the Client Server the callback request for the callback query.
Client Action	Collect data.
Attributes	VCB_ORIGIN_APPLICATION_ID
	VCB_TARGET_APPLICATION_ID
	VCB_REFERENCE_ID
	VCB_RESPONSE_TYPE = 1 "SingleResult"
	VCB_RECORD_HANDLE
	VCB_CURRENT_DN
	VCB_QUERY_INDEX
	VCB_ERROR_CODE
	VCB_ERROR_NAME
	UserData
EventVCBCancel	
G*plus* Adapter Event Name:	EventVCBCancel
VCB Protocol User Event:	RequestCallbackCancel
Description	The VCB Server sends this request to the VCB Client to confirm the cancellation of the callback request.
Client Action	Wait for next the callback request.
Attributes	VCB_ORIGIN_APPLICATION_ID
	VCB_TARGET_APPLICATION_ID
	VCB_REFERENCE_ID
	VCB_RESPONSE_TYPE = 0 "Acknowledge"
	VCB_RECORD_HANDLE (or VCB_CONTACT)
	VCB_ERROR_CODE
	VCB_ERROR_NAME
EventVCBReschedule	
G*plus* Adapter Event Name:	EventVCBReschedule
VCB Protocol User Event:	RequestCallbackReschedule
Description	The VCB Server sends this request to the VCB Client to confirm the rescheduling of the callback request.
Client Action	Wait for the next callback request.

Table 36: The Voice Callback Device Events (Continued)

Attributes	VCB_ORIGIN_APPLICATION_ID VCB_TARGET_APPLICATION_ID VCB_REFERENCE_ID VCB_RESPONSE_TYPE = 0 "Acknowledge" VCB_RECORD_HANDLE VCB_ERROR_CODE VCB_ERROR_NAME
EventVCBPreview	
G*plus* Adapter Event Name:	EventVCBPreview
VCB Protocol User Event:	RequestCallbackPreview
Description	Desktop confirms that the callback request has been received and accepted.
Callback Server Action	Not specified
Attributes	VCB_ORIGIN_APPLICATION_ID VCB_TARGET_APPLICATION_ID VCB_REFERENCE_ID VCB_RESPONSE_TYPE = 0 "Acknowledge" VCB_RECORD_HANDLE
EventVCBProcessed	
G*plus* Adapter Event Name:	EventVCBProcessed
VCB Protocol User Event:	RequestCallbackProcessed
Description	The VCB Server sends this request to the desktop to confirm that the corresponding request has been processed.
Desktop Action	Not specified
Attributes	VCB_ORIGIN_APPLICATION_ID VCB_TARGET_APPLICATION_ID VCB_REFERENCE_ID VCB_RESPONSE_TYPE = 0 "Acknowledge" VCB_RECORD_HANDLE
EventVCBReject	
G*plus* Adapter Event Name:	EventVCBReject

Table 36: The Voice Callback Device Events (Continued)

VCB Protocol User Event:	RequestCallbackReject
Description	The VCB Server sends this request to the desktop to confirm that the corresponding request has been rejected.
Desktop Action	Not specified
Attributes	VCB_ORIGIN_APPLICATION_ID VCB_TARGET_APPLICATION_ID VCB_REFERENCE_ID VCB_RESPONSE_TYPE = 0 "Acknowledge" VCB_RECORD_HANDLE
EventVCBError	
G*plus* Adapter Event Name:	EventVCBError
VCB Protocol User Event:	RequestCallbackError
Description	The VCB Server sends this request to the desktop to inform about an error condition.
Desktop Action	Removes the work item from the desktop when the value of the ERR# is {100, 102, 103, 104, 108} and keeps it when the value of the ERR# is {101, 105, 106 }.
Attributes	VCB_ERROR VCB_ ERROR_ NUMBER VCB_ORIGIN_APPLICATION_ID VCB_TARGET_APPLICATION_ID = 0 VCB_REFERENCE_ID VCB_ RESPONSE _TYPE= 4 "Error"

The Voice Callback Error Codes and Descriptions

This section describes the Voice Callback error codes, error numbers, and descriptions (see Table 37).

Table 37: VCB Error Codes and Descriptions

VCB_ERROR	VCB_ERROR_ NUMBER	Description
Invalid Request	100	Invalid request type

Table 37: VCB Error Codes and Descriptions (Continued)

VCB_ERROR	VCB_ERROR_ NUMBER	Description
Invalid Request Data	101	Some mandatory keys are missing
Call not found	102	Call not found on the server
Record not found	103	VCB received request regarding non-existing or already-processed record
Request Already Processed	104	Received request refers to Callback Request already processed
DB Error	105	Can not execute the request due to DB error
AddRecordError	106	Can not add the record
ScheduleRecordError	108	Error rescheduling a record

The Predefined Attribute Names

Table 38 represents the reserved attribute names.

Table 38: Predefined Attributes

Key	Value	Type	Description
VCB_USER_EVENT_REQUEST	Request identifier for any request	String	Request (procedure) name
VCB_USER_EVENT_RESPONSE	Response identifier for any response	String	Name of corresponding request (procedure)
VCB_ RESPONSE _TYPE	0, "Acknowledge" 1, "SingleResult" 2, "MultipleResult" 3, "MultipleResultEnd" 4, "Error"	Int	Type of response
VCB_TARGET_APPLICATION_ID	Application ID	Int	
VCB_ORIGIN_APPLICATION_ID	Application ID	Int	
VCB_REFERENCE_ID	Session-wide unique request identifier	Int	Maintained by requestor

Table 38: Predefined Attributes (Continued)

Key	Value	Type	Description
VCB_CALL_RESULT	0, "Ok" 3, "General Error" 4, "System Error" 6, "Busy" 7, "No Answer" 8, "SIT Detected" 9, "Answering Machine" 10, "All Trunks Busy" 11, "SIT Invalid Num" 12, "SIT Vacant" 13, "SIT Oper Intercept" 14, "SIT Unknown" 15, "SIT No Circuit" 16, "SIT Reorder" 17, "Fax Detected" 21, "Abandoned" 26, "Dropped" 27, "Dropped No Answer" 28, "Unknown" 32, "Silence" 33, "Answer" 34, "NuTone" 35, "NoDialTone" 36, "NoProgress" 37, "NoRingBack" 38, "NoEstablishe" 39, "Pager Detected" 40, "Wrong Party" 41, "Dial Error" 42, "Call Drop Error" 43, "Switch Error" 44, "No Free Port Error" 45, "Transfer Error" 46, "Stale" 47, "Agent CallBack Error" 48, "Group CallBack Error	Int	Call result from the previous dial attempt or the call result that the agent sends to automatically change the detected one.
VCB_ERROR_CODE	Error Code, see Error Codes table.	Int	Must be present when VCB_RESPONSE_TYPE = 4 "Error"

Table 38:   Predefined Attributes (Continued)

Key	Value	Type	Description
VCB_ERROR_NAME	Error Name	String	Must be present when VCB_RESPONSE_TYPE = 4 "Error"
VCB_CURRENT_DN	DN Number	String	DN that the callback request currently resides on
VCB_CONTACT	String of digits	String	Contact number (for example, phone number)
VCB_TYPE	0, "Unknown" 1, "ASAP" 2, "Scheduled"	String	Callback Type
VCB_QUEUE		String	To define the search condition by the queue
VCB_STAGE	0, "Unknown" 1, "Queued" 2, "NotQueued"	Int	To define the search condition by the stage of the callback request processing. "Queued" means that the callback request is in queue awaiting distribution.
VCB_ROUTING_POINT	Name of Routing Point associated with Voice Callback Service	String	To specify the Routing Point in the request to the VCB Server.
VCB_ROUTING_POINT_DESCR	Description of Routing Point associated with Voice Callback Service	String	Description which the agent may use to assign the agent-created callback request to the most appropriate queue.
VCB_LOCATION	Name of Switch	String	To specify the location in a request to the VCB Server.
VCB_DATE_TIME	"MMDDYYYYHHMM"	String	The date/time to schedule the callback
VCB_EWT	"HHMM"	String	EWT estimated during the Callback offering.
VCB_EWT_TIME	"MMDDYYYYHHMM"	String	The date/time when the EWT was estimated.
VCB_ATTEMPTS	0...	Int	The number of attempts

Table 38:   Predefined Attributes (Continued)

Key	Value	Type	Description
VCB_ORIGIN	0, "Unknown" 1, "IVR" 2, "WEB" 3, "Desktop"	String	Callback origination
VCB_TZ_NAME		String	Configuration Server TZ name (usually a standard three letter abbreviation such as PST).
VCB_STATUS	0, "Unknown" 1, "Not Available" 2, "Available"	String	The availability of the Callback service
VCB_QUERY_COUNT	0...	Int	The Number of callback requests returned by the query.
VCB_QUERY_INDEX	1...	Int	Index of the callback request in the query result set.
CRM_RECORD_MODE	"Preview" "Auto"	String	VCB delivery mode, "Auto dial" or "Manual"
CRM_REFERENCE_ID		String	
CRM_IS_VCB_VIEW	"TRUE" "FALSE"	String	Indicates that the current Siebel view is the VCB view.
CRM_PROTOCOL_EXT	"OCS" "VCB"	String	The Genesys protocols shortcuts for Outbound Contact Server and Voice Callback Solution.

GENESYS

7

Deploying the UCS Gateway Server

This chapter describes how to configure and install the G*plus* UCS Gateway Server for Siebel CRM and consists of the following sections:

Overview

The G*plus* UCS Gateway Server for Siebel CRM is a back-end component used by the G*plus* Adapter for Siebel CRM Multimedia Component to access the data stored in the Genesys Universal Contact Server.

The process of configuring and installing the UCS Gateway Server includes the following general procedures:

- Configuring Genesys
- Installation of the G*plus* UCS Gateway Server for Siebel CRM
- Configuring Siebel

New in This Release

This section provides information about new features or functionality in the G*plus* UCS Gateway Component.

Release 8.0.210   The following new features or functionality are included in this release:

- HTTP Authentication. See "Deploying Basic HTTP Authentication" on page 557 for details.

Release 8.0.2   The following new features or functionality are included in this release:

- Support for Siebel versions 8.1.1.14/8.2.2.14 (IP2014) for both HI and Open UI modes.
- Support for Genesys Framework 8.5.
- Support for Red Hat Enterprise Linux 6.
- Support for Red Hat Enterprise Linux 7.
- Support for AIX 7.1.
- Support for Solaris 11.

Release 8.0.1   The following new features or functionality are included in this release:

- Support for the API (application programming interface) based on the PSDK (Platform SDK).

Release 8.0.0   The following new features or functionality are included in this release:

- Support for the Red Hat Enterprise Linux (RHEL) operating system.

Configuring Genesys

This chapter describes how to configure and install the G*plus* UCS Gateway Server for Siebel CRM.

Configuring the Genesys side of the G*plus* UCS Gateway Server for Siebel CRM consists of the following sections:

Prestart Information

Before starting the configuration process you should have the following Genesys Framework applications running:

*   Configuration Database
*   Configuration Server
*   Configuration Manager

As a Configuration Manager user, you should have sufficient privileges to make changes to Configuration Layer objects.

Importing the UCS Gateway Server for Siebel CRM Application Template

This section describes how to import the G*plus* UCS Gateway Server for Siebel CRM Application Template.

Recommendations

Genesys recommends using an Application Template when you are configuring your Adapter. The Application Template for your Adapter contains the most important configuration options set to the values recommended for the majority of environments. When modifying configuration options for your Adapter later in the process, you can change the values inherited from the template rather than create all the options by yourself.

Procedure:
USC Gateway Server: Importing the USC Gateway Server Application Template

Purpose: To import the G*plus* UCS Gateway Server for Siebel CRM Application Template.

Start of procedure

1.   In Configuration Manager, under Environment, right-click the Application Templates folder.

2.   Select Import Application Template.

3.   Browse to select the Application Template for the G*plus* UCS Gateway Server for Siebel CRM.

4.   Select the following template, as follows:
     *   Release 8.0.0: Gplus_UCS_Gateway_for_SiebelCRM_800.apd
     *   Release 8.0.1: Gplus_UCS_Gateway_for_SiebelCRM_801.apd

- Release 8.0.2: Gplus_UCS_Gateway_for_SiebelCRM_802.apd

5. Click Open.

The Properties dialog box for the Application Template object displays.

6. Optional: Edit the Application Template name.

7. Click OK to accept the default values.

The Application Template object has been imported to Genesys Configuration Layer.

End of procedure

Next Steps

- Create the Configuration Layer Application object for the G*plus* UCS Gateway Server for Siebel CRM. See,

Creating the UCS Gateway Server for Siebel CRM Application Object

This section describes how to create the G*plus* UCS Gateway Server for Siebel CRM Application object.

Procedure:
UCS Gateway Server: Creating the UCS Gateway Server Application object

Purpose: To create the Application object for the G*plus* UCS Gateway Server for Siebel CRM.

Start of procedure

1. In Configuration Manager, under Environment, right-click the Applications folder.

2. Select New > Application.

3. Select the Application template that you just created:
- Release 8.0.0: Gplus_UCS_Gateway_for_SiebelCRM_800.apd
- Release 8.0.1: Gplus_UCS_Gateway_for_SiebelCRM_801.apd
- Release 8.0.2: Gplus_UCS_Gateway_for_SiebelCRM_802.apd

4. Click OK.

The Properties dialog box for the Application object appears.

End of procedure

Next Steps

- Configure the tabs in the Properties dialog box for the UCS Gateway Server. See, "Configuring the Tabs in the Properties Dialog Box".

Configuring the Tabs in the Properties Dialog Box

This section describes how to configure the tabs in the Properties dialog box, arranged in the order in which they display.

Procedure:
UCS Gateway Server: Configuring the tabs in the Properties dialog box

Purpose: To configure the tabs in the Properties dialog box for the G*plus* UCS Gateway Server for Siebel CRM.

Start of procedure

General Tab **1.** Click the General tab in the Properties dialog box (see Figure 32 on page 384).
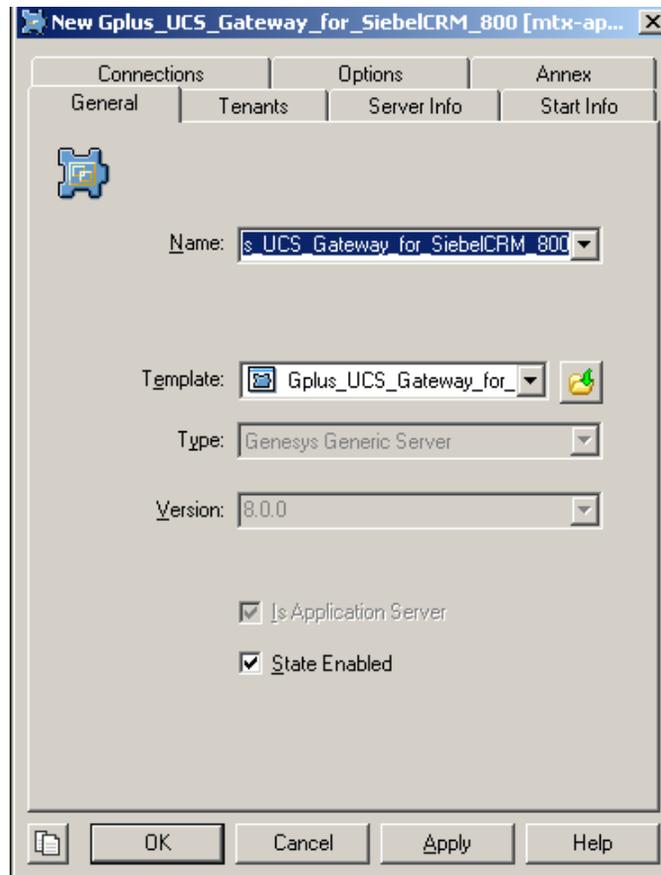
Figure 32: View of General Tab in Single-Tenant Environment

2. In the Name list, click the name of the Application object that you are configuring.

3. In the Template list, click the name of the template that you are configuring or use the Browse button to select the template that you are configuring.

4. Click Apply.

5. Next, if you are working in a multi-tenant environment, go to the Tenants Tab; otherwise, go to the Server Info Tab.

Note: The Tenants tab only displays, if you are working in a multi-tenant environment.

Tenants Tab   6. Select the Tenants tab.

7. Select the Genesys Tenants under which the objects that are exported from Siebel are created.

8. Click Add.

Server Info Tab   9. Select the Server Info tab.

10. In the Host field, use the Browse button to select the host on which you are installing the UCS Gateway Server, and click OK.

11. In the Port list, add a port with any valid port number.

Note:　This option determines the HTTP port number that is used by the UCS Gateway Server to receive HTTP packets from Siebel. Use the value that you enter for this option when you configure the Siebel part of the UCS Gateway Server.

Start Info Tab　**12.** Select the Start Info tab.

13. In the Working Directory field, enter the full name of the Component installation directory on the host that you specified on the Server Info tab. The value that you enter in this field is used as the default destination folder during installation.

14. Enter any valid value into each of the following fields:
  - Command Line
  - Command Line Arguments

The values that you enter in these fields are overwritten during installation; however, data must be present in these fields during the configuration process.

15. Leave the default values for the remaining fields.

Connections Tab　**16.** Select the Connections tab.

17. Add a connection to the Universal Contact Server

Options Tab　**18.** Select the Options tab (see Figure 33 on).
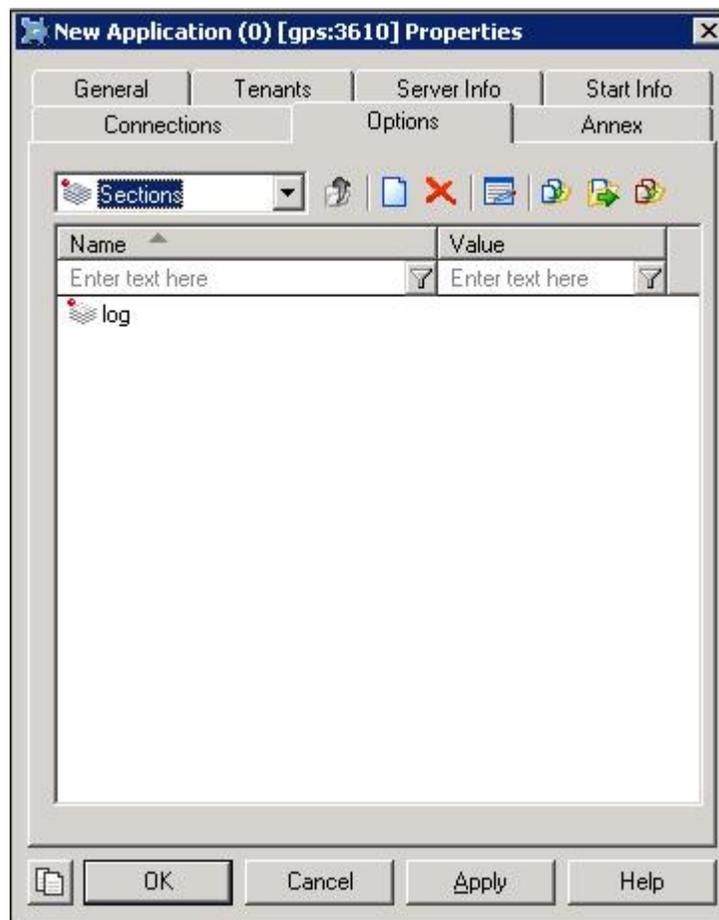
Figure 33: View of Options Tab

In the Sections pane, the following section is listed:

- log

19. Double-click the log section to start.

20. Configure the configuration options that are described in the following section, "Setting the Genesys Configuration Options for the UCS Gateway Server for Siebel CRM" on page 387.

End of procedure

Next Steps

- Set the G*plus* UCS Gateway Server for Siebel CRM's configuration option. See, Procedure: UCS Gateway Server: Configuring the tabs in the Properties dialog box, on page 383.

Setting the Genesys Configuration Options for the UCS Gateway Server for Siebel CRM

Unless specified otherwise, set the G*plus* UCS Gateway Server for Siebel CRM configuration options in the Options tab of the Application object using the following navigation path:

In Genesys Adminstrator—Application object > Options tab > Advanced View (Options).

In Configuration Manager—Application object > Properties dialog box > Options tab.

log Section

The G*plus* UCS Gateway Server for Siebel CRM supports a common set of Log options to allow the precise configuration of the log file output. For a complete list of Common Log options and their descriptions, see the "Common Log Options" chapter of the *Framework 8.0 Configuration Options Reference Manual*.

Note: If you do not specify any log options, the default values apply.

Installation

This section describes the installation process for the G*plus* UCS Gateway Server for Siebel CRM.

UCS Gateway Memory Allocation

If you intend for your environment to handle large e-mails (which usually means working with large e-mail attachments), you may need to increase the amount of memory that you allocate for the UCS Gateway Server. Insufficient memory may cause the UCS Gateway Server to return an HTTP 500 error when the driver makes a request to add attachments to EmailOut.

Use the -Xmx option to set the maximum memory size for the UCS Gateway Server. You can change this option in the starter.ini file on Windows platforms, or in the startup.sh file on UNIX platforms—for example, to set the maximum memory size to 512 MB, set the value of the -Xmx option to 512M.

Installing the UCS Gateway Server in a Windows Environment

This section describes how to install the G*plus* UCS Gateway Server for Siebel CRM on a Windows system.

Procedure:
UCS Gateway Server: Installing the UCS Gateway Server in a Windows environment

Purpose: To install the G*plus* UCS Gateway Server for Siebel CRM in a Windows environment.

Start of procedure

1. To start the installation process, run the setup.exe file from the G*plus* UCS Gateway Server for Siebel CRM installation package.

2. In the Welcome window, click Next.

3. In the Configuration Parameters for the Genesys Configuration Server window, enter the following:
   - The name of the host on which Configuration Server is running.
   - The communication port that client applications must use to connect to Configuration Server.
   - The user name used to access Configuration Server.
   - The password used to access Configuration Server.

4. Click Next.

5. In the Select Application window, select the Application object by its corresponding number that you configured in the previous procedure, Procedure: UCS Gateway Server: Creating the UCS Gateway Server Application object, on page 382.

6. Click Next.

7. In the Choose Destination Location window, click Next to accept the default destination folder or use the Browse button to select a different destination folder.

8. In the Ready to Install window, click Install.

9. In the Setup Complete window, click Finish.

The G*plus* UCS Gateway Server for Siebel CRM is now installed.

In the G*plus* Adapter for Siebel CRM program folder in the Start menu, you can see that the installer created a shortcut for the G*plus* UCS Gateway Server for Siebel CRM.

End of procedure

Next Steps

- Configure the Siebel part of the G*plus* UCS Gateway Server for Siebel CRM. See, "Configuring Siebel" on page 390.

Installing the USC Gateway Server in a UNIX Environment

This section describes how to install the G*plus* UCS Gateway Server for Siebel CRM on an UNIX operating system.

Note: Do not use special symbols in any destination directory name when installing in a UNIX environment.

Procedure:
USC Gateway Server: Installing the USC Gateway Server in a UNIX environment

Purpose: To install the G*plus* UCS Gateway Server for Siebel CRM in a UNIX environment.

Start of procedure

1. In the directory in which the USC Gateway Server installation package was copied, locate a shell script named: install.sh.

2. Run this script from the command prompt by typing sh and the file name:

   sh install.sh

3. When prompted, specify the host name, port, user name, and password of the computer on which the USC Gateway Server will be installed—for example:

   a. Enter the host name of Configuration Server.

   b. Enter the port of Configuration Server.

   c. Enter the user name for Configuration Server.

   d. Enter the password for the user name.

4. Choose the Configuration Server environment by its corresponding number.

5. Specify the Application object you configured in the previous procedure, Procedure: UCS Gateway Server: Creating the UCS Gateway Server Application object, on page 382.

6. Specify the full path to the destination directory in which you want the USC Gateway Server to be installed.

The G*plus* UCS Gateway Server for Siebel CRM is now installed.

End of procedure

Next Steps

• Configure the Siebel side of the environment. See the section, "Configuring Siebel".

Configuring Siebel

For the description of how to configure Siebel for the G*plus* UCS Gateway Server for Siebel CRM, please see "Configuring the UCS Gateway Definition in Siebel" on page 422.

Chapter

8

Deploying the Multimedia Component

This chapter describes how to configure and install the G*plus* Adapter for Siebel CRM Multimedia Component and includes the following sections:

- Overview, page 391
- New in This Release, page 392
- Installing the Multimedia Component, page 392
- Configuring Siebel, page 395
- Configuring the UCS Gateway Definition in Siebel, page 422
- Multimedia Component Driver and Configuration Parameters, page 423
- Interaction Server Device Commands, page 430
- Interaction Server Device Commands (Changes for 8.0.1), page 440
- Interaction Server Device Commands (Changes for 8.0.2), page 441
- Outbound Contact Device Commands, page 442
- Multimedia Component Device Events, page 446
- Multimedia Component Device Events (Changes for 8.0.2), page 454

Overview

The process of configuring and installing the G*plus* Adapter for Siebel CRM Multimedia Component includes the following general procedures:

- Installing the Genesys Open Media Communication Driver (itx_scdrv) on the G*plus* Communication Server for Siebel CRM.
- Compiling and deploying the Siebel repository file.
- Configuring the access permissions for the Siebel business services.
- Deploying the web templates and browser scripts.
- Configuring Siebel.

See the G*plus* *Adapter 8.0 for Siebel CRM User's Guide* for additional information about using this component.

New in This Release

This section provides information about new features or functionality in the G*plus* Adapter for Siebel CRM Multimedia Component:

Release 8.0.210   No new features were introduced in this release.

Release 8.0.2   The following new features or functionality are included in this release:

- Support for Siebel versions 8.1.1.14/8.2.2.14 (IP2014) for both HI and Open UI modes.
- Support for Siebel Chat (now both Genesys and Siebel Chat are supported).
- Support for Genesys Framework 8.5.
- Support for Red Hat Enterprise Linux 6.
- Support for Red Hat Enterprise Linux 7.
- Support for AIX 7.1.
- Support for Solaris 11.

Release 8.0.110   The following new features or functionality are included in this release:

- Support for Siebel versions 8.1.1.11/8.2.2.4 (IP2013) for both HI and Open UI modes.

Release 8.0.1   No new features were introduced in this release.

Release 8.0.0   The following new features or functionality are included in this release:

- Support for the Push Preview dialing mode for Outbound Campaigns.
- Support for the Red Hat Enterprise Linux (RHEL) operating system.
- Support for the Multimedia Session Failover Handling. (See Chapter 11, "Deploying the Multimedia Session Failover Handling," on page 545).

Installing the Multimedia Component

This section describes the installation process for the G*plus* Adapter for Siebel CRM Multimedia Component. The G*plus* Adapter for Siebel CRM Multimedia Component can be installed on either Windows or UNIX platforms.

Installation Prerequisites

The prerequisites for the installation of the G*plus* Adapter for Siebel CRM Multimedia Component are as follows:

- Back up the Siebel database and the Siebel repository (*.srf) file.

- Install the G*plus* Communication Server (see Chapter 3, "Deploying the Communication Server for Siebel CRM," on).

- Install the G*plus* UCS Gateway (see Chapter 7, "Deploying the UCS Gateway Server," on).

Installation and Uninstallation Guidelines

In a Windows environment, the target computer may have only one installation of the G*plus* Adapter for Siebel CRM Multimedia Component. In an UNIX environment, more than one installation of the G*plus* Adapter for Siebel CRM Multimedia Component is allowed, but on any single computer, you can only have one Adapter for every Siebel Server.

If the G*plus* Adapter for Siebel CRM Multimedia Component is already installed on the target computer, a new installation is considered as a reinstallation on that computer. Before you reinstall, you are asked the following, depending on your environment:

- In a Windows environment, you are asked to uninstall the G*plus* Adapter for Siebel CRM Multimedia Component.

- In an UNIX environment, you are asked to confirm the reinstallation.

The installation package contains a set of complementary icons for the Ready/NotReady buttons. During the installation process, the standard Siebel icons for the NotReady button are overwritten with the new ones from the Genesys installation package. To preserve the standard Siebel icons, Genesys recommends that you backup the following Siebel image files in a safe location:

- icon_notready_enabled.gif

- icon_notready_disabled.gif

If you uninstall the G*plus* Adapter, you can manually restore the original buttons with these files.

Note: Genesys recommends backing up the Siebel web template file, CCHtmlType.swf, before starting the installation procedure.

Installation

This section describes how to install the G*plus* Adapter for Siebel CRM Multimedia Component on either a Windows or a Linux operating systems:

Note: Do not use special symbols in any destination directory name when installing G*plus* Adapter for Siebel CRM Multimedia Component in an UNIX environment.

Procedure:
Multimedia Component: Installing the Multimedia Component

Purpose: To install the G*plus* Adapter for Siebel CRM Multimedia Component on either a Windows or a Linux operating system.

Start of procedure

1. To start the installation process, run the setup.exe file (for Windows) or run the install.sh file (for UNIX) from the G*plus* Adapter for Siebel CRM Multimedia Component installation package on the host where the G*plus* Communication Server is installed.

    Note: For the Windows installation, skip to step 3.

Linux 2. Enter the correct path to the G*plus* Communication Server, so that the installation file has the correct folder location to place the files necessary for running the Open Media Communication Driver (itx_scdrv).

Windows **3.** Enter the path to the main installation directory, so that the installation program has the correct folder location to place the Siebel archive (*.sif) files, the Siebel SmartScripts, the Siebel image files, and the general product information.

Note: If the Voice Component is not installed, you must *manually* copy the following files to their proper locations:

Copy the files representing the Siebel icon images from the destination directory, <Destination Directory>/<Siebel Version>/images/, where <Siebel Version> is 7.7/8.0/8.1/8.1_8.2_OUI/IP2014, depending on your version of the Siebel Server, to the following icon images directory: < Web Server Host>/<SWEIconImages directory>.

If your Siebel is configured to use Open UI, copy the images to the folder <SiebSrvr>\WEBMASTER\images\enu, where SiebSrvr is the folder where Siebel server is installed.

For Siebel version 7.8, use the 7.7 directory.

End of procedure

Next Steps

• Configure the Siebel part of the G*plus* Adapter for Siebel CRM Multimedia Component. See the section, "Configuring Siebel".

Configuring Siebel

This section describes how to configure the Siebel part of the G*plus* Adapter for Siebel CRM Multimedia Component.

Configuring the Siebel section of the G*plus* Adapter for Siebel CRM Multimedia Component consists of the following sub-sections:

Prestart Information

Before starting this part of the configuration process, you must make sure to do the following:

• You must have the Siebel Tools application running.

• You should be connected to the local copy of the Siebel Server database.

- You should have sufficient privileges to check-in and check-out projects from the Siebel Server repository.
- You should ensure that all required steps from "Patching and Configuring Siebel CRM" are performed.

Configuring Siebel Using the Siebel Tools

Use the Siebel Tools to compile an updated version of the Siebel repository file (SRF or *.srf file) for one, or more, of the Siebel applications that you use on your Siebel Server, which you will then deploy in the server. For more information about using Siebel Tools, see the Siebel documentation.

Configuring Siebel using the Siebel Tools consists of the following sub-sections:

Overview of the Siebel Repository File Preparation

When compiled, the Siebel repository file integrates information from the various Multimedia Component archives files (the *.sif files) that you import.

For the Multimedia Component, you must import all of the Multimedia Component *.sif files and resolve any data conflicts introduced by the imported file.

Configuring Siebel Tools for the Siebel Repository File Deployment

This section describes how to deploy the Siebel repository file, the browser scripts, the web templates, and how to change the Siebel Server configuration. To deploy the repository file, use the compiled repository file as the Siebel Server repository file.

Procedure:
Multimedia Component: Updating the Siebel repository file

Purpose: To update the Siebel repository file.

Start of procedure

1. Start Siebel Tools.

2. If you did not previously import the *.sif archive files for the Voice Component, then import the following archive files:

   - GenSymbolicStrings.sif (for Siebel 7.8 and higher)
   - GenComm.sif
   - GenesysTools.sif

   These files are located at: <Installation Directory>/<Siebel Version> , where <Siebel Version> equals 7.7, 8.0, 8.1, /8.1_8.2_OUI or IP2014, depending on your version of Siebel Server.

Create the Genesys Multimedia Project

3. Create the Genesys Multimedia project, if it does not exist.

4. Lock the Genesys Multimedia project.

   Note: See the section, "Configuring Siebel Using Siebel Tools" on page 197 for more information on how to import the *.sif archive files and how to create and lock a project.

5. In Siebel Tools, in the Project field of the Object Explorer, select All Projects.

6. Select the Business Component folder.

   The Business Component window displays.

7. Select one of the Action-related Siebel business components (BC) that you use and that corresponds to the following two conditions:

   a. The business component must have the following fields:

   - Attachment Flag
   - Attachment Id
   - Call Id
   - Comment
   - Description
   - Email BCC Line
   - Email Body
   - Email CC Line
   - Email Sender Address
   - Email To Line
   - Parent Activity Id
   - Primary Contact Id
   - Short Comment
   - Started
   - Status
   - Type

   b. The business component should *not* have the following fields:

- Attachment File Name
- Attachment Flag 2
- Interaction Id

For example, you might use the Action Home Page business component or Action business component.

8. Make three copies of the chosen business components and name them as follows:
   - Action - Genesys MCR.
   - Action - Genesys MCR(EmailOut).
   - Genesys Action(Chat).

9. Change the Project field for newly-created business components to Genesys Multimedia.

10. Make a copy of the Action Attachment business component and name it Action Attachment - MCR EmailOut.

11. Change the Project field for newly-created business components to Genesys Multimedia.

12. Select one of the Contact-related Siebel business components that you use and that corresponds to the following two conditions:

    a. The business component must have the following fields:
       - Comment
       - Email Address
       - Employee Flag
       - First Name
       - Job Title
       - Last Name
       - Login Name
       - Status

    b. The business component should *not* have the following fields:
       - IsBCC
       - IsCC
       - IsTo

    For example, you might use the Contact(All) business component.

13. Make three copies of the chosen business component and name them as follows:
    - Contact - Genesys MCR(EmailOut)
    - Inbound Interaction Contact List
    - Outbound Interaction Contact List

14. Change the Project field for newly-created business components to Genesys Multimedia.

15. Make a copy of the EAI HTTP Transport business service and name it Genesys EAI HTTP Transport. Select or change the Project field for a newly-created business service to Genesys Multimedia.

Importing the GplusMCR_Email_ Chat_nested.sif Archive File

16. In Siebel Tools, select Tools > Import from Archive....

17. In the Select Archive to Import window, navigate to the GplusMCR_Email_Chat_nested.sif archive file from the following directory: <InstallationDirectory>/<Siebel Version>/, where <Siebel Version> equals 7.7, 8.0, 8.1, /8.1_8.2_OUI or IP2014, depending on your version of Siebel Server. This file updates the business components that you just created.

18. Click Open.

The Import Wizard-Preview window displays.

19. In the Import Wizard-Preview window, locate the Conflict resolution section, and select Merge the object definition from the archive file with the definition in the repository.

20. Click Next.

The Import Wizard Review Conflicts and Actions window displays.

21. Merge any conflicts, if they correspond to your needs, keeping the following rules in mind:

    a. Tables 39 to 44 contain the relevant objects and attributes for the Genesys Adapter functionality, which are related to the cloned business components. All other objects and attributes *must* be taken from the original business components—for example, right-click on each of the attributes (if they exist) and re-set the resolution to Repository.

    b. For each object or attribute that is listed in the tables, re-set the resolution to File, as long as no conflict exists.

    c. If there are conflicts for a given attribute or object, the conflicts must be manually resolved.

Table 39: Action - Genesys MCR Business Component Objects and Resolutions

Object Type	Object Name	Resolution
BC Attribute	INACTIVE	File
BC Attribute	NO_DELETE	File
BC Attribute	NO_INSERT	File
BC Attribute	NO_MERGE	File
BC Attribute	NO_UPDATE	File
BC Attribute	SCRIPTED	File

Table 39: Action - Genesys MCR Business Component Objects and Resolutions (Continued)

Object Type	Object Name	Resolution
Server Script	BindActivityContacts	File
Server Script	BusComp_PreWriteRecord	File
Server Script	BusComp_SetFieldValue	File
Server Script	UpdateEmailInfo	File
Field	Attachment File Name	File
Field	Interaction Id	File
Multi Value link	Attachment	File

Table 40: Action - Genesys MCR EmailOut BC Objects and Resolutions

Object Type	Object Name	Resolution
BC Attribute	INACTIVE	File
BC Attribute	NO_DELETE	File
BC Attribute	NO_INSERT	File
BC Attribute	NO_MERGE	File
BC Attribute	NO_UPDATE	File
BC Attribute	SCRIPTED	File
Server Script	BindActivityContacts	File
Server Script	BusComp_PreWriteRecord	File
Server Script	BusComp_SetFieldValue	File
Server Script	PrintPropertySet	File
Server Script	SkipMETA	File
Server Script	UpdateEmailInfo	File
Field	Attachment File Name	File
Field	Attachment Flag 2	File

Table 40: Action - Genesys MCR EmailOut BC Objects and Resolutions (Continued)

Object Type	Object Name	Resolution
Field	Interaction Id	File
Multi Value link	Attachment	File

Table 41: Genesys Action(Chat) BC Objects and Resolutions

Object Type	Object Name	Resolution
BC Attribute	INACTIVE	File
BC Attribute	NO_DELETE	File
BC Attribute	NO_INSERT	File
BC Attribute	NO_MERGE	File
BC Attribute	NO_UPDATE	File
Field	Type	File

Table 42: Action Attachment - MCR EmailOut BC Objects and Resolutions

Object Type	Object Name	Resolution
BC Attribute	INACTIVE	File
BC Attribute	NO_DELETE	File
BC Attribute	NO_INSERT	File
BC Attribute	NO_MERGE	File
BC Attribute	NO_UPDATE	File
BC Attribute	SCRIPTED	File
Server Script	BusComp_NewRecord	File
Server Script	BusComp_PreDeleteRecord	File
Server Script	BusComp_PreInvokeRecord	File
Server Script	BusComp_WriteRecord	File
Field	ActivityIsNew	File

Table 43: Contact - Genesys MCR (EmailOut) BC Objects and Resolutions

Object Type	Object Name	Resolution
BC Attribute	INACTIVE	File
BC Attribute	NO_UPDATE	File
BC Attribute	SCRIPTED	File
Server Script	(declarations)	File
Server Script	BusComp_PreGetFieldValue	File
Server Script	BusComp_PreInvokeRecord	File
Server Script	BusComp_PreSetFieldValue	File
Server Script	ClearIntArrays	File
Field	IsBCC	File
Field	IsCC	File
Field	IsTo	File

Table 44: Genesys EAI HTTP Transport Business Service Objects and Resolutions

Object Type	Object Name	Resolution
Server Script	PatchInputWithAuthData	File
Server Script	Service_PreInvokeMethod	File

22. Click Next. The "Do you wish to proceed?" window opens.

23. Click Yes. The objects from the archive are imported into the Siebel repository.

24. Click Finish to complete the import.

25. For the Action - Genesys MCR and Action - Genesys MCR(EmailOut) business components, set the Force Active field attribute to the checked state (or true in the Property window) for the following fields:

- Email Sender Address
- Email To Line
- Email BCC Line
- Email CC Line

- Description
- Email Body
- Interaction Id
- Type
- Started
- Status

26. For the Action Attachment - MCR EmailOut business components, set the Force Active field attribute to the checked state (or true in the Property window) for the following fields:
    - Activity Id
    - ActivityFileExt
    - ActivityFileName
    - ActivityFileSize

27. For the Genesys Action(Chat) business components, set the Force Active field attribute to the checked state (or true in the Property window) for the following fields:
    - Description
    - Interaction ID
    - Type
    - Started
    - Status

28. For the Inbound Interaction Contact List and Outbound Interaction Contact List business components, set the attribute No Update to the unchecked state (or false in the Property window).

29. Import the GplusMCR_Email_Chat.sif archive file from the following directory: <Installation Directory>/<Siebel Version>/, where <Siebel Version> is 7.7, 8.0, 8.1, /8.1_8.2_OUI or IP2014.

30. Click Open.

    The Import Wizard-Preview window displays.

31. In the Import Wizard-Preview window, locate the Conflict resolution section, and select Merge the object definition from the archive file with the definition in the repository.

32. Click Next.

    The Import Wizard Review Conflicts and Actions window displays.

33. In the Conflicting Objects pane, select the Business Component category.

34. In the Object differences pane, select the Attribute differences pane. If there are any attributes in the Attribute differences pane, make sure that the Resolution column for each attribute is set to Repository. To do this: right-click on an attribute and select Repository from the drop-down menu.

35. Click Next.

    The "Do you wish to proceed?" window displays.

36. Click Yes.

The objects from the archive are imported into the Siebel repository.

37. Click Finish to complete the import.

Creating the Web Templates

38. Make sure that the following web templates exist:
- Applet Form Grid Layout
- Applet Popup Form Grid Layout

Siebel web templates include a preset format that Siebel CRM reuses each time it requires a particular layout.

39. If they do not exist, create them by importing the gridlayout.sif file from the following folder: <Siebel Tools root directory>\OBJECTS.

> **Note:** Before importing the gridlayout.sif file, the Siebel Tools and Common Components projects need to be locked.

40. If there is no gridlayout.sif file available, create the following two web templates:

a. <Name = Applet Form Grid Layout, Type = Applet Template - Grid Layout, Project = <Any Locked Project>

then create the following new record on the Web Template Files applet:

filename=CCAppletFormGridLayout.swt, Name = Generic Form Applet (Base)

b. <Name = Applet Popup Form Grid Layout, Type = Applet Template - Grid Layout, Project = <Any Locked Project>

then create the following new record on the Web Template Files applet:

filename=CCAppletPopupFormGridLayout.swt, Name = Generic Form Applet (Base)

Adding the Genesys Tab to the Siebel Application

41. Add the Genesys tab (screen = Genesys) to the Siebel application that is being used—for example, Siebel Universal Agent (for the Siebel Horizontal version) or Siebel Financial Services (for the Siebel Vertical version).

42. When inserting the Genesys tab in addition to adding the Screen column, you also need to define the Text - String Override field or provide the Text - String Reference reference.In the Object differences pane, select the Attribute differences pane. If there are any attributes in the Attribute differences pane, make sure that the Resolution column for each attribute is set to Repository. To do this: right-click on an attribute and select Repository from the drop-down menu.

Creating Symbolic String References

43. If you want to use the Text - String Reference reference, then you must first create a symbolic string for the Genesys display name and then enter it into the text field. Symbolic strings provide a centralized mechanism for storing and managing repository text strings.

a. Click Symbolic String in the Object Explorer view.

b. Click on the Symbolic String view and press Ctrl+N to create a new record.

c. Set the Name field to a X_Genesys value.

     d. Set the Current String Value field to a Genesys name.

     e. Set Project Field to any locked project.

     f. Set the Type field to a Conversion value.

Adding or Modifying the Siebel Server Script Event Handler

44. For Sibel Server versions prior to 8.1.1.11, add or modify (if it already exists) the Siebel Server script event handler for the Application_PreNavigate event of the Siebel application being used.

If you have Siebel Server version 8.1.1.11/8.2.2.4 or later, skip this step.

The content of the script event handler is provided in the <InsDir>/<Siebel Version>/ Application_PreNavigate.es file.

     a. Refer to "Updating the Application's Application_PreNavigate Event Server Script" on page 624 in the Appendix for general instructions.

     b. Select the Application object, open the context menu by clicking the right mouse button, and choose Edit Server Scripts. If there are no server scripts associated with the Application object, you are prompted to select the scripting language. Select eScript in this case.

     c. Double-click the Application_PreNavigate event. Siebel Tools displays the content of the corresponding event handler.

     d. If the content of the corresponding event handler consists only of return (ContinueOperation), replace this information with the content of the <InsDir>/<Siebel Version>/Application_PreNavigate.es file.

     e. If the Universal Callback feature of the Voice Component was deployed, then the Application_PreNavigate event handler has already been modified, and now needs to be merged with the Application_PreNavigate.es file.

     f. If Application_PreNavigate is already customized, you will need to merge the Application_PreNavigate.es file with your customized code, depending on your original customization logic.

     g. Save the changes in the server script and close the script window.

45. For users of Siebel 8.0, and higher:

Allow access to the Genesys Chat and Communications Client business services from the browser script by adding the following user properties to the application you use:

- ClientBusinessServiceN = Genesys Chat
- ClientBusinessServiceM = Communications Client where N and M must be sequential integers, starting at 0 and incrementing by one. These integers should be different from the existing integers.

46. Compile the *.srf file and the browser scripts.

End of procedure

Next Steps

- Configure the Siebel Tools for the repository file deployment. See the section, "Configuring the Siebel Server".

Configuring the Siebel Server

Complete the following procedures, described in this section, to configure the Multimedia Component to work with the Siebel Server:

- Multimedia Component: Configuring Siebel Tools for the repository file deployment, page 406
- Multimedia Component: Deploying the Siebel repository file, browser scripts, web templates, and changing the Siebel Server configuration, page 408

Procedure:
Multimedia Component: Configuring Siebel Tools for the repository file deployment

Purpose: To configure Siebel Tools for the repository file deployment.

Start of procedure

Applying Changes to the Siebel Database Tables

1. Create the Genesys project tables on the Siebel Server Database by completing the following steps:

   Note: The Multimedia Component requires its own tables, which do not exist in the original Siebel database.

   a. In Siebel Tools, in the Table field of the Object Explorer, make a query to select four new Genesys tables: CX_AGNT_WB_CNT, CX_GEN_ACT_CHAT and CX_GEN_ACT_X.

   b. With the three tables selected, right-click Add to Archive. The Export to Archive File window displays the three tables.

   c. Select the appropriate temporary file name and click Save.

   d. Close the Siebel Tools application, but open it again under the Siebel Server database.

   e. Import the newly created archive file.

   f. Click Next, and then click Next again.

   g. Click Finish.

2. In Siebel Tools, in the Object Explorer, navigate to the Table object.

3. Navigate to the List of Tables pane.

4. Locate the following tables:
   - CX_AGNT_WB_CNT
   - CX_GEN_ACT_CHAT
   - CX_GEN_ACT_X

5. Click Apply.

   A pop-up warning window appears.

6. Click OK to accept the warning.

7. In the Apply Schema window, from the Tables drop down list, select Current Row.

8. Enter the correct values for the Database user, Database user password, and ODBC data source fields.

   You must enter the user name and password for a Database user who has administrator's privileges in the Siebel environment.

   See the Siebel documentation for more information about creating custom tables.

9. Enter the correct values for the Database user, Database user password, and ODBC data source fields.

   You must enter the user name and password for a Database user who has administrator's privileges in the Siebel environment.

   See the Siebel documentation for more information about creating custom tables.

10. Click Apply.

    The message Changes successfully applied appears, indicating that the tables were created.

11. Click the Activate button to propagate the database changes and make them available to all users.

End of procedure

Next Steps

- Deploy the Siebel repository file, the browser scripts, the web templates, and change the Siebel Server configuration. See, Procedure: Multimedia Component: Deploying the Siebel repository file, browser scripts, web templates, and changing the Siebel Server configuration.

Procedure:
Multimedia Component: Deploying the Siebel repository file, browser scripts, web templates, and changing the Siebel Server configuration

Purpose: To deploy the Siebel repository file and to change the Siebel Server configuration.

Start of procedure

1. Stop the Siebel Server.

2. Back up the original Siebel repository file.

3. Copy the compiled repository file instead of the original Siebel repository file.

4. Deploy (copy) the browser scripts from the <Browser script compilation folder> as set in Siebel Tools to the corresponding Siebel Web Server extension folder (<SWEApp>/public/language_code).

5. For Siebel Servers prior to 8.1.1.11, deploy (copy) the content of <Installation Directory>/<SiebelVersion>/WEBTEMPL folder into the Siebel Server WEBTEMPL directory (siebsrvr/WEBTEMPL), where <Siebel Version> is 7.7, 8.0, or 8.1.

   For Siebel Servers 8.1.1.11/8.2.2.4 or later, which support High Interactivity mode, deploy (copy) the content of <Installation Directory>/<SiebelVersion>/WEBTEMPL folder into the Siebel Server WEBTEMPL directory (siebsrvr/WEBTEMPL/custom), where <Siebel Version> is 8.1_8.2_OUI or IP2014.

   For Siebel Servers 8.1.1.11/8.2.2.4 or later, which support Open UI mode, deploy (copy) the content of <Installation Directory>/<SiebelVersion>/WEBTEMPL/OUIWEBTEMPL folder into the Siebel Server OUI WEBTEMPL directory (siebsrvr/WEBTEMPL/OUIWEBTEMPL/custom), where <Siebel Version> is 8.1_8.2_OUI or IP2014.

6. Add the content of the <Installation Directory>/<Siebel Version>/WEBTEMPL/ToMerge/GenesysHtmlType.swf file to the CCHtmlType.swf from the Siebel Server WEBTEMPL directory (siebsrvr/WEBTEMPL for High Interactivity mode and siebsrvr/WEBTEMPL/OUIWEBTEMPL for Open UI mode, when applicable).

7. For Siebel Servers that support High Interactivity mode, update the commToolbar_shared.js file located in the corresponding Siebel Web Server extension folder (<SWEApp>/public/language_code/latest system browser folder (the folder with the name = largest number)/scripts) using instructions

from the <Installation Directory>/<Siebel Version>/SCRIPTS/commToolbar_shared.js file, where <Siebel Version> is 7.7, 8.0, 8.1, /8.1_8.2_OUI or IP2014.

For Siebel Servers that support Open UI mode, copy all script files from the <Installation Directory>/<Siebel Version>/SCRIPTS/OUI folder into the Siebel Server Open UI customization folder (siebsrvr\WEBMASTER\siebel_build\scripts\siebel\custom), where <Siebel Version> is 8.1_8.2_OUI or IP2014.

8. For Siebel versions prior to 8.0:

Allow access to the Genesys Chat Business Service from the browser script:

a. Open the application configuration file with a text editor—for example, for the Siebel Horizontal version of the Siebel Universal Agent, use the uagent.cfg file, or for the Siebel Vertical version of the Siebel Financial Services, use the fins.cfg file.

b. Find the [SWE] section.

c. Add the following lines:

ClientBusinessServiceN = Genesys Chat

ClientBusinessServiceM = Communications Client

where N and M must be sequential integers, starting with 0 and incrementing by one. These integers should be different from the integers in the existing ClientBusinessService lines.

d. Save and close the file.

9. Start the Siebel Server.

10. Set up the business service query access for the following business services in the Siebel Server configuration:

- Genesys Chat
- MCR Activity manager
- MCR TopActiveWorkItem
- GplusMediaRoute
- MCR Session manager
- GplusMediaRouteIXN (if the Media Routing Component is deployed)
- GplusMediaRouteiWD (if the iWD Routing Component is deployed)
- Genesys Siebel Chat

To configure the Business Service Query Access List parameter of the Siebel Enterprise Server:

a. Log into the Siebel Server as a Siebel administrator.

b. Navigate to Administration > Server Configuration > Enterprises.

c. In the Enterprise Servers applet, select Enterprise Server.

d. On the Third Level View Bar, switch to the Parameters tab.

e. In the Enterprise Parameters applet, select the Business Service Query Access List parameter.

f. If the value is empty, set it to the following:

Genesys Chat,MCR Activity manager,MCR TopActiveWorkItem,GplusMediaRoute,GplusMediaRouteIXN,GplusMediaRoute iWD,MCR Session manager,Genesys Siebel Chat.

Otherwise, add it to the end, separated by a comma.

Note: There are no empty spaces after the commas in the value.

g. Restart the Siebel Server to make the changes effective.

11. For Siebel Servers that support Open UI mode, register custom scripts:

a. Log into the Siebel Server as a Siebel administrator.

b. Navigate to Site Map > Administration - Application > Manifest Files.

c. Add the following three new records into the Files applet:
- siebel/custom/GplusChatAppletPModel.js
- siebel/custom/GplusChatAppletRenderer.js
- siebel/custom/GplusCommToolbarUpdate.js

d. Navigate to Site Map > Administration - Application > Manifest Administration.

e. Add three new records into the UI Objects and subordinate applets, as follows:

For the GplusCommToolbarUpdate.js, update the parameters for the applet rows as shown in Table 45, Table 46, and Table 47.

Table 45: UI Objects applet row

Parameter name	Value
Inactive Flag	N
Type	Application
Usage Type	Common
Name	PLATFORM INDEPENDENT

Table 46: Object Expression applet row

Parameter name	Value
Inactive Flag	N
Group Name	

Table 46: Object Expression applet row (Continued)

Parameter name	Value
Expression	Desktop
Level	1
Operator	
Web Template Name	

Table 47: Files applet row

Parameter name	Value
Inactive Flag	N
Name	siebel/custom/GplusCommToolbarUpdate.js

For the GplusChatAppletPModel.js, update the parameters for the applet rows as shown in .

Table 48: UI Objects applet row

Parameter name	Value
Inactive Flag	N
Type	Applet
Usage Type	Presentation Model
Name	Genesys Chat Applet

Table 49: Object Expression applet row

Parameter name	Value
Inactive Flag	N
Group Name	
Expression	
Level	1
Operator	
Web Template Name	

Table 50: Files applet row

Parameter name	Value
Inactive Flag	N
Name	siebel/custom/GplusChatAppletPModel.js

For the GplusChatAppletRenderer.js, update the parameters for the applet rows as shown in Table 51, Table 52, and Table 53.

Table 51: UI Objects applet row

Parameter name	Value
Inactive Flag	N
Type	Applet
Usage Type	Physical Renderer
Name	Genesys Chat Applet

Table 52: Object Expression applet row

Parameter name	Value
Inactive Flag	N
Group Name	
Expression	Desktop
Level	1
Operator	
Web Template Name	

Table 53: Files applet row

Parameter name	Value
Inactive Flag	N
Name	siebel/custom/GplusChatAppletRenderer.js

Next Steps

- Configure Siebel using the Web Client. See the section, "Configuring Siebel Using the Siebel Web Client" on page 413.

Configuring Siebel Using the Siebel Web Client

This section describes the specific processes required to complete the configuration for the Multimedia Component.

Note: You must be logged in as a Siebel Administrator before starting.

Complete the following procedures, described in this section, to configure the Siebel Call Center application on the Siebel Server.

Procedure:
Multimedia Component: Updating the associated List of Values (LOV) tables

Purpose: To import the List of Values table for the Multimedia Component.

Start of procedure

1. Log in as Siebel administrator.
2. Open the Site Map, then navigate to Administration - Business Service, Simulator.
3. Specify Genesys Tools as the Service Name.
4. Specify ImportAll as the Method Name.
5. Set Iterations to 1.

6. In the Input Arguments applet, click Load From File, browse for
<InsDir>/<Siebel Version>/GplusMCR_LOV.xml and load this file as shown in
Figure 34 on .
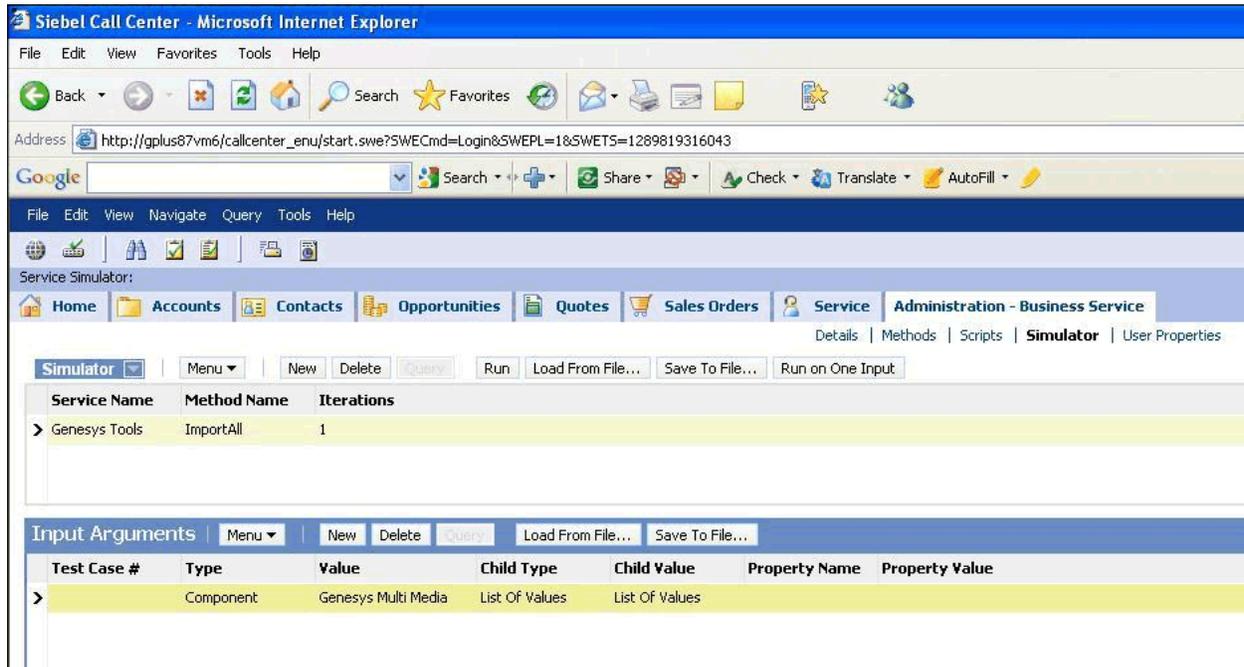
Figure 34: Importing the Lists of Values Table

7. Click Run on the Service Methods applet.

8. Check the import results in the Output Arguments applet.

End of procedure

Next Steps

•   Set up the agent responsibilities. See, Procedure: Multimedia Component:
    Setting up the agent responsibilities.

Procedure:
Multimedia Component: Setting up the agent responsibilities

Purpose: To set up the agent responsibilities.

Start of procedure

1. Log in as Siebel administrator.

2. Navigate to Application Administration, Responsibilities.

3. Find the Genesys Multimedia Agent responsibility.

4. Use the Users applet to add agents who will use Multimedia views.

5. Log out.

End of procedure

Next Steps

- Create a customized configuration file. See, Procedure: Multimedia Component: Creating a customized configuration file, on page 415.

Procedure:
Multimedia Component: Creating a customized configuration file

Purpose: To create a new configuration file.

Start of procedure

1. Log in as Siebel administrator.

2. Navigate to Communication Administration, All Configurations view.

3. Create a new configuration import <Siebel Version>/GenComm_universal.def from the installation directory.

For more information, refer to the section "Universal Definition File" on page 31 in Chapter 5, "System Requirements."

End of procedure

Next Steps

- Configure the CTI connection. See, Procedure: Multimedia Component: Configuring the CTI connection, on page 415.

Procedure:
Multimedia Component: Configuring the CTI connection

Purpose: To configure the CTI connection.

Start of procedure

Using the Siebel Web Client:

1. Log in as a Siebel administrator.

2. Navigate to the Communication Administration, All Configurations view.

3. Find the target configuration.

4. Click the Profiles tab, then click on the Gplus_Universal driver.

5. Specify the Driver:PrimaryGenCommServerURL parameter, by providing the location of the G*plus* Communication Server.

    For example, if the G*plus* Communication Server is installed on the cti host, and port 18000 is specified in the Genesys Configuration Layer as the Server port, then specify the following parameter:

    Driver:PrimaryGenCommServerURL=" http://cti:18000"

    > **Note:** The specified name must be resolvable from the host where the Siebel Communication Server is running.

6. If you have a backup G*plus* Communication Server, enter the information for the Driver:BackupGenCommServerURL parameter, otherwise leave it empty.

7. Specify the Service:ConnectionName parameter. This parameter should contain the Connection name (the Interaction Server Application object) as specified in the Connections tab for the G*plus* Communication Server. For example:

    Service:ConnectionName= InteractionServer_Switch_122

8. Log out.

End of procedure

Next Steps

*   Create agents and agent-related records. See the section, "Multimedia Component Agent Administration".

Multimedia Component Agent Administration

This section explains the processes involved in creating agents and agent-related records.

The Multimedia Component agent administration consists of the following procedures:

Procedure:
Multimedia Component: Assigning the correct responsibility to an agent

Purpose: To assign the correct responsibility to an agent.

Start of procedure

1. Log in as a Siebel administrator.

2. Select Site Map > Administration > Application, Responsibilities.

3. On the Responsibilities applet, select the Genesys Multimedia Agent responsibility.

4. Add the required agents who will use the Multimedia view by using the Users applet on the Responsibilities tab.

End of procedure

Next Steps

• Configure or create a Teleset. See, .

Procedure:
Multimedia Component: Configuring or creating Telesets

Purpose: To configure or create a Teleset, which is a set of DNs that represent a single workplace.

Start of procedure

1. Navigate to Site Map > Administration - Communications > All Telesets.

2. Find the Teleset that the agent already uses, or create a new one for the new agent. For each Teleset, provide the following information:
   • Create a new Teleset in the Telesets list.
   • Add your agent's Extension.
   • For a new agent, add the agent on the Agents tab

3. Give the Teleset a unique name—for example, you may want the Teleset based on the cubicle number, or the location of the Teleset.

4. Select the Extensions tab.

5. Add all the Teleset DNs in the list (one, or more DNs) with a Place name of <Genesys Place>@GP. For each DN, you must you must define the DN type:

S—Standard DN

A—ACD Position

Notes: If you use the G*plus* Configuration Synchronization component, then the Genesys Place name is the same as the name of the Teleset in Siebel—for example:

> Teleset Name= T_CUBE_0001

> then extension should be:

T_CUBE_0001@GP

If a Teleset has only DNs of type ACD Position, then you must configure at least one of these DNs as a DN of type S in Siebel.

Table 54 shows how an example Teleset configuration scenario could be defined.

Table 54: Siebel Telesets

Genesys		Siebel	
Object Type	**Name**	**Object Type**	**Name**
Place	MyTeleset	Teleset	MyTeleset
		S	MyTeleset@GP
Extension	1001	S	1001
ACD Position	2001	A	2001

6. Follow Steps 2 to 5 above for each Teleset.

End of procedure

Next Steps

- Assign an agent to a Multimedia configuration. See, Procedure: Multimedia Component: Assigning agents to a Multimedia configuration, on page 419.

Procedure:
Multimedia Component: Assigning agents to a Multimedia configuration

Purpose: To assign an agent to a Multimedia configuration.

Start of procedure

1. Navigate to the Agent General Profile view.
2. Find the agent, then add the multimedia configuration as the primary configuration for this agent.
3. If Siebel-Genesys synchronization is not used, then it is necessary to manually define the Agent Login value the same as the "Employee ID" value from the Genesys Configuration layer.

End of procedure

Next Steps

• There are no further steps.

Workbin Parameters

Workbins are available for use by an agent in Siebel defined as a List of Value table of type MCR_WORKBIN_TYPE, where the Language-Independent Code field defines the Siebel name of the workbin. The display value should correspond to the workbin name as defined in Genesys Interaction Workflow Designer. It may be necessary to change the display value to correlate with a pre-existing Genesys Multimedia configuration.

Configuring Chat

This section describes how to configure the Genesys or Siebel Chat User Interface (UI) for Genesys chat media.

Notes: Siebel Chat UI is supported starting from Patchset17 for Siebel IP2013 and Patchset7 for Siebel IP2014.

The 8.0.2 version of Gplus Adapter for Siebel CRM Multimedia component supports both Genesys (legacy) and Siebel (native Siebel) Chat UIs for Genesys chat media.

Genesys Chat UI is configured by default. No additional setup actions are required.

Configuring the Siebel Chat UI consists of the following procedures:

Procedure:
Multimedia Component: Changing the accepting chat workflow process

Purpose: To adjust the Siebel chat accepting process to the Genesys chat protocol.

Start of procedure

1. Connect to a server database using Siebel Tools.

2. Select the Workflow Process folder in the Object Explorer pane.

3. Choose the Chat Standard Accept Process - v3 workflow process.

4. Press the Revise toolbar button.

   A new WF record with In Progress status will appear.

5. Start editing the new WF process.

6. Skip checking for transferred chat. Either remove or disconnect the IsTransferChat? step.

7. For the step Get Service Request Id, change the Business Service Method property to GetChatDataFromDBByInteractionId.

   The workflow graph should look like the following:
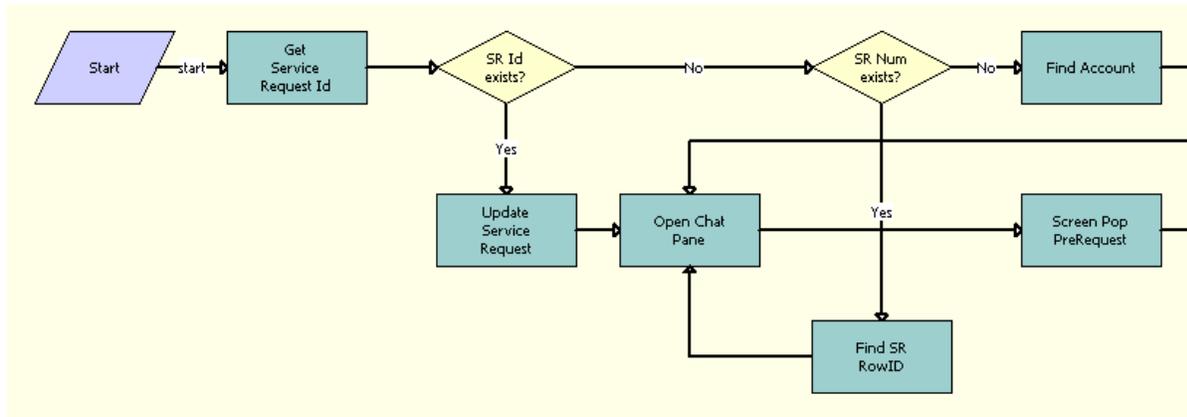
Figure 35: The Workflow Graph

8. Publish and activate the workflow process by pressing the Publish/Activate toolbar button.

End of procedure

Next Steps

- Changing the CTI configuration (.def). See Procedure: Multimedia Component: Changing the CTI configuration (.def)

Procedure:
Multimedia Component: Changing the CTI configuration (.def)

Purpose: To select CTI configuration parameters and commands for Siebel Chat UI support.

1. Set the CTI Configuration's IsChatEnabled parameter value to TRUE.

2. Replace the chat media with the SiebelChat in the Channel String CTI Driver parameter value, or comment/uncomment the corresponding lines in the out-of-the-box GenComm_universal.def file. You do not have to setup SiebelChat media within the Genesys framework as it works as an alias for standard Genesys chat media.

3. Replace the chat media with the SiebelChat in the DeviceCommand and FilterSpec parameter values of the NotReadyForGChatGroup CTI command, or comment/uncomment the corresponding lines in the out-of-the-box GenComm_universal.def file.

End of procedure

Next Steps

- Updating the Business Service Query Access List. See Procedure: Multimedia Component: Updating the Business Service Query Access List

Procedure:
Multimedia Component: Updating the Business Service Query Access List

Purpose: To adjust the Siebel chat accepting process to the Genesys chat protocol.

Start of procedure

1. Confirm that Genesys Siebel Chat is included in the BusinessServiceQueryAccessList Siebel enterprise parameter value.

   See

End of procedure

Next Steps

• No further steps are required.

Configuring the UCS Gateway Definition in Siebel

This section describes how to configure the UCS Gateway definition in Siebel.

Procedure: Multimedia Component: Configuring the UCS Gateway definition in Siebel

Purpose: To configure the UCS Gateway definition in Siebel.

Start of procedure

1. Log in as a Siebel administrator.
2. Navigate to the Outbound Web Services configuration view (Web Services administration).
3. Import the <Siebel Version>/Gplus.xml file.

   This import creates the Genesys Gplus WebService outbound web service.
4. Configure the Genesys Gplus WebService by changing the URL in the Address field of the Gplus Service port. The URL should point to the installed UCS Gateway—for example:

   http://cti-ucs:2006/gateway

End of procedure

Next Steps

• There are no further steps.

Multimedia Component Driver and Configuration Parameters

Table 55 contains a list of the driver parameters for the Multimedia Component, which includes the Genesys Chat and Genesys E-mail drivers.

Table 55: Multimedia Component Driver Parameters

Parameter name	Default value	Must be defined in profile?	Comment
Driver:BackupGenCommServerURL		No	Specifies the Universal Resource Locator (URL) of the backup G*plus* Communication Server.
Driver:LibraryName	UniComm	No	Specifies the name of the remote driver module that is loaded by the G*plus* Communication Server to handle requests for the Siebel Communication Driver.
Driver:PrimaryGenCommServerURL	CHANGE_ME	Yes	Specifies the Universal Resource Locator (URL) of the primary G*plus* Communication Server
Name	Gplus_Universal	No	Specifies the driver name.
Service:AgentId	{@AgentID}	No	Uses the {@AgentID} macro to obtain the Agent Login of the current agent.

Table 55: Multimedia Component Driver Parameters (Continued)

Parameter name	Default value	Must be defined in profile?	Comment
Service:ConnectionName	CHANGE_ME	Yes	Specifies the name of the Interaction Server that is used.
Service:DNList	{@DNList}	Yes	Uses the {@DNList} macro to obtain a list of DNs with the standard Siebel extensions of type S associated with the current agent.

Table 56 contains a list of the communications drivers properties for the Multimedia Component.

Table 56: Multimedia Component Communications Drivers Properties

Property name	Default value	Required	Comment
Channel Type	Gplus_Universal	Yes	Specifies the driver's channel type; should not be changed.
Inbound Flag	True	Yes	Specifies the driver capability: can work with inbound interactions; should not be changed.
Outbound Flag	True	Yes	Specifies the driver capability: can work with outbound interactions; should not be changed.
Interactive	True	Yes	Specifies the driver capability: designed for interactive work; should not be changed.
Channel String	voice, outboundpreview, email, chat, SiebelEmail, BackgroundEmail	Yes	Specifies a comma-separated list of supported Genesys media types.
Library Name	GenCommDrv	Yes	Specifies the name of the local driver module.

Table 57 contains a list of the configuration parameters for the Multimedia Component.

Table 57: The Multimedia Component Configuration Parameters

Parameter Name	Default Value	Required	Comment
ChatTranscriptDirection	TopToBottom	No	Determines the direction of the chat transcript. The valid values are TopToBottom and BottomToTop.
CheckPopupBeforeExecute	FALSE	Yes	Checks for the presence of the popup window before navigation. Genesys recommend leaving this parameter set to FALSE.
EmailInboundDoneQueue	__STOP__	Yes	Multi-channel and routing (MCR) Queue where the inbound e-mails are placed after the Mark done or Reply operations are finished. __STOP__ means that the processing of the interaction is completely stopped. If post-processing for the inbound e-mails is needed, then this value should be changed to the post-processing queue as defined in your business process workflow. (For details on post-processing, see the Genesys eServices documentation).

Table 57: The Multimedia Component Configuration Parameters (Continued)

Parameter Name	Default Value	Required	Comment
EmailOutboundDoneQueue	__STOP__	Yes	Multi-channel and routing (MCR) Queue where the outbound e-mails are placed after the Delete operation is finished. __STOP__ means that the processing of the interaction is completely stopped. If post-processing for the deleted outbound e-mails is needed, then this value should be changed to the post-processing queue as defined in your business process workflow. (For details on post-processing, see the Genesys eServices documentation).
ExceptionLogPrefix	exclog_	No	Determines the prefix of a log file—for example, the full name is formulated as "< ExceptionLogPrefix >+<Agent Name>.log") and is used for storing error messages that may appear during CTI calls to the business service methods.
FromAddress	CHANGE_ME	Yes	This value must be modified. The value of this option is used to fill out the From field in the outbound and reply e-mails—for example, support@mycompany.com.
MediaRoutingDefaultQueue	E-mails for agent processing	No	If the queue is not explicitly specified in the submit request, then the Siebel eMails and other open media work items are submitted into this queue.

Table 57: The Multimedia Component Configuration Parameters (Continued)

Parameter Name	Default Value	Required	Comment
MediaRoutingDoneQueue	__STOP__	No	Multi-channel and routing (MCR) Queue where the Siebel eMails and/or other open media interactions are placed after an agent completes the handling of the interaction. __STOP__ means that the processing of the interaction is completely stopped. If post-processing for the Siebel eMails and/or open media interactions is needed, then this value should be changed to the post-processing queue as defined in your business process workflow. (For details on post-processing, see the Genesys eServices documentation).
NewOutboundEmailQueue	E-mails for processing by agents	Yes	Multi-channel and routing (MCR) Queue where the new outbound e-mail is created.
NoteEmailBodyTruncated	...Message is too large... whole content is attached as <EmailBodyAttachmentName>	Yes	The text that is added to the end of an inbound e-mail, if the size of the e-mail body exceeds the limit (16008 bytes). <EmailBodyAttachmentName> is a special tag which is replaced by the name of the attachment containing the whole original message.
NoteReplyBodyTruncated	...Source message is too large... skipped.	Yes	The text that is added (instead of the body of the original e-mail) to the end of the reply e-mail, if the content of the reply e-mail generated by default is greater than 16008 bytes.

Table 57: The Multimedia Component Configuration Parameters (Continued)

Parameter Name	Default Value	Required	Comment
OpenMediaSessionRecovery	TRUE	Yes	Turns the failure-recovery mechanism on or off. The default value is TRUE.
ReplyOutboundEmailQueue	E-mails for processing by agents	Yes	Multi-channel and routing (MCR) Queue where the outbound reply e-mail is created.
RecoveryPullViewUserDataKey	<user data key name>	Yes	Defines the interaction's user data key name where the View Id (<queue name>/<view name>) for pulling an interaction is taken from.
			The default value is RecoveryPullViewSystem. Make sure that the value of this parameter corresponds to the used data key set in the Prepare_for_Hold strategy.
			The key RecoveryPullViewSystem is reserved by the Adapter and must be used to specify the View Id in the workflow.

Table 57: The Multimedia Component Configuration Parameters (Continued)

Parameter Name	Default Value	Required	Comment
			Possible values of the RecoveryPullViewUserData Key are:
			1. Any string: points to any key/value pair that contains the View Id from which the interaction needs to be pulled on recovery.
			2. If a specified key does not exist: the Adapter pulls the interaction without any View Id specifications, so the interaction is pulled independently in the queue/view/strategy that it is located.
			3. If a specified key exists, but the value of it is defined as a non-existent View Id: the interaction is not pulled.
			4. Empty string (""): the Adapter pulls an interaction without a View Id specification.
			5. If the configuration parameter is not present in the def.file: the default value RecoveryPullViewSystem is used.
SendEmailNewDefaultQueue	Outbound queue	Yes	Multi-channel and routing (MCR) where the outbound e-mail is placed as a result of the Send operation.
SendEmailReplyDefaultQueue	Outbound queue	Yes	Multi-channel and routing (MCR) Queue where the reply e-mail is placed as a result of the Send operation.

Interaction Server Device Commands

Table 58 on page 430 displays the Interaction Server driver supports the following device commands:

- OpenMediaLogin
- OpenMediaLogout
- @some-media@OpenMediaReady
- @some-media@OpenMediaNotReady
- @some-media@OpenMediaChangeMediaStateReason
- OpenMediaAccept
- OpenMediaReject
- OpenMediaPlaceInQueue
- OpenMediaLeave
- OpenMediaSelect
- OpenMediaDeselect
- OpenMediaSubmit
- OpenMediaReply
- OpenMediaChangeProperties
- OpenMediaPlaceInWorkbin
- OpenMediaPullInteractionById
- OpenMediaTransfer
- ChatSend
- ChatReleaseParty

Notes: • This set of Interaction Server device commands is different than those that the Voice driver supports.
- Changes for the 8.0.1 release are described in the following subsection Interaction Server Device Commands for 8.0.1.

Table 58: The Interaction Server Driver Device Commands

OpenMediaLogin		
Description	Performs a Login request.	
stringParam	Not Used	

Table 58: The Interaction Server Driver Device Commands (Continued)

datasetParam	Reason (ReasonCode)	Optional	The reason for the login.
	Description (ReasonValue)	Optional	The description of the reason.
OpenMediaLogout			
Description	Performs a Logout operation.		
stringParam	Not Used		
datasetParam	Reason (ReasonCode)	Optional	The reason for the logout.
	Description (ReasonValue)	Optional	The description of the reason.
@some-media@OpenMediaReady			
Description	Sets the agent status to Ready for some-media media types.		
stringParam	Not Used		
datasetParam	Reason (ReasonCode)	Optional	The reason for becoming Ready, if any reason is needed.
	Description (ReasonValue)	Optional	The human-readable description of the specified reason.
@some-media@OpenMediaNotReady			
Description	Sets the agent status to Not-Ready for some-media media types.		
stringParam	Not Used		
datasetParam	Reason (ReasonCode)	Optional	The reason for becoming Not-Ready, if any reason is needed.
	Description (ReasonValue)	Optional	The human-readable description of specified reason

Table 58: The Interaction Server Driver Device Commands (Continued)

@some-media@OpenMediaChangeMediaStateReason			
Description	Changes the agent media state reason for some-media media types at any time while the agent is logged in.		
stringParam	Not Used		
datasetParam	Reason (ReasonCode)	Optional	The reason for the media state, if any is needed.
	Description (ReasonValue)	Optional	The human-readable description of the specified reason.
OpenMediaAccept			
Description	Accepts the interaction (inbound or outbound) offered to the agent.		
stringParam	Not Used		
datasetParam	TrackingID	Optional	The driver identifier of the item to be accepted. If the value is not specified, the driver chooses an 'offered' interaction based on the built-in list of priorities.
	ChatUsrNickName	Optional	The nickname of the agent that is used for chat (this value is ignored for other interactions). If the value is not provided, the Service:AgentId value is used instead.
	TimeZoneOffset	Optional	The offset of the agent time zone that is used only for chat interactions. The default value is 0.
OpenMediaReject			
Description	Rejects (declines) the offered, but not accepted, interaction back to the router.		
stringParam	Not Used		

Table 58: The Interaction Server Driver Device Commands (Continued)

datasetParam	TrackingID	Optional	The driver identifier of the item that is rejected. If the value is not specified, the driver chooses an 'offered' interaction based on the built-in list of priorities.
	Reason (ReasonCode)	Optional	The reason for the decline.
	Description (ReasonValue)	Optional	Human-readable description of specified reason
OpenMediaPlaceInQueue			
Description	Places the interaction handled by the agent into the specified queue and removes it from the desktop. If the queue name is __STOP__, then this command will completely stop the processing of the interaction.		
stringParam	Not Used		
datasetParam	TrackingID	Optional	The driver identifier of the interaction. If the value is not specified, the driver uses, by default, the currently selected interaction.
	Reason (ReasonCode)	Optional	The reason for placing the interaction in the queue or for stopping the processing.
	Description (ReasonValue)	Optional	The human-readable description of the specified reason.
OpenMediaLeave			
Description	Declines the interaction currently handled by the agent. Similar to OpenMediaReject but is only for accepted interactions.		
stringParam	Not Used		

Table 58: The Interaction Server Driver Device Commands (Continued)

datasetParam	TrackingID	Optional	The driver identifier of the interaction. If the value is not specified, the driver uses, by default, the currently selected interaction.
	Reason (ReasonCode)	Optional	The reason for declining the interaction.
	Description (ReasonValue)	Optional	The human-readable description of the specified reason.
OpenMediaSelect			
Description	Marks the specified interaction as selected (such as the default interaction if the command does not contain the interaction id). This command should be used to synchronize the state of the Siebel communication toolbar which has one active (selected) interaction and the driver. If this command changes the selected interaction, the driver sends the corresponding OpenMediaSelected event with the interaction information to Siebel.		
stringParam	Not Used		
datasetParam	TrackingID	Mandatory	The driver identifier of the item to be selected.
OpenMediaDeselect			
Description	Deselects the specified interaction and sends an OpenMediaDeselected event to Siebel. If the specified interaction was not selected, then no action is taken.		
stringParam	Not Used		
datasetParam	TrackingID	Mandatory	The driver identifier of the item to be deselected.

Table 58: The Interaction Server Driver Device Commands (Continued)

OpenMediaSubmit			
Description	Creates a new interaction with the specified attributes in the specified queue.		
stringParam	Not Used		
datasetParam	InteractionId	Mandatory	The interaction ID of the interaction to be created.
	InteractionSubtype	Mandatory	The interaction subtype, such as OutboundNew, OutboundReply. A list of possible attribute values is defined in Genesys Configuration under the corresponding business attribute.
	InteractionType	Mandatory	An interaction type, such as Outbound or Inbound. A list of possible attribute values is defined in Genesys Configuration under the corresponding business attribute.
	MediaType	Mandatory	The media type of the new interaction—for example, email or chat.
	SubmitQueue	Mandatory	The queue where the new interaction is created.
	Properties	Optional	The additional attributes (user data). Note that all the unknown key-value pairs in datasetParam are also treated as user data.
	ParentId	Optional	The ID of the parent interaction.

Table 58: The Interaction Server Driver Device Commands (Continued)

OpenMediaReply			
Description	Places in queue or stops the processing of the specified interaction (parent) and creates a child interaction with the specified ID. This command combines OpenMediaPlaceInQueue and OpenMediaSubmit requests for convenience purposes. If more advanced logic is required, the base commands should be used instead.		
stringParam	Not Used		
datasetParam	TrackingID	Mandatory	The driver ID of the source item.
	Queue	Mandatory	The target queue for the source interaction (can be __STOP__).
	Reason (ReasonCode)	Optional	The reason for the reply.
	Description (ReasonValue)	Optional	The human-readable description of the specified reason.
	InteractionId	Mandatory	The Interaction Server Interaction ID of the interaction to be created.
	InteractionSubtype	Mandatory	The interaction subtype—for example, OutboundNew or OutboundReply. A list of possible attribute values is defined in Genesys Configuration under the corresponding business attribute.
	InteractionType	Mandatory	The Interaction type—for example, Outbound or Inbound. A list of possible attribute values is defined in Genesys Configuration under the corresponding business attribute.

Table 58: The Interaction Server Driver Device Commands (Continued)

datasetParam (continued)	MediaType	Mandatory	The media type of the new interaction—for example, email or chat.
	SubmitQueue	Mandatory	The queue where the new interaction is created.
	Properties	Optional	The additional attributes (user data). Note that all of the unknown key-value pairs in datasetParam are also treated as user data.
	ParentId	Optional	The ID of the parent interaction.
OpenMediaChangeProperties			
Description	Changes the user properties associated with the specified interaction.		
stringParam	Not Used		
datasetParam	TrackingID	Optional	The driver identifier of the interaction. If the value is not specified, the driver uses, by default, the currently selected interaction.
	Properties	Optional	The properties to be updated.
	DeletedProperties	Optional	The properties to be removed from the interaction.
	<Unrecognized>	Optional	All of the pairs from the provided dataset that are not recognized as predefined attributes are treated similarly to the value of the Properties parameter, - updated.

Table 58: The Interaction Server Driver Device Commands (Continued)

OpenMediaPlaceInWorkbin			
Description	Places the specified interaction in the specified workbin. Upon successful completion, the interaction is removed from the CTI toolbar.		
stringParam	Not Used		
datasetParam	TrackingID	Optional	The driver identifier of the interaction. If the value is not specified, the driver uses, by default, the currently selected interaction.
	Workbin	Mandatory	The name of the workbin as defined in the Genesys Interaction Workflow Designer.
	WorkbinAgentId WorkbinPlaceId WorkbinPlaceGroupId WorkbinAgentGroupId	Depending on the type of workbin, one of these attributes will be mandatory.	The properties to be removed from the interaction.
	Reason (ReasonCode)	Optional	The operation reason.
	Description (ReasonValue)	Optional	The description of the reason.
OpenMediaPullInteractionById			
Description	Pulls the specified interaction.		
stringParam	Not Used		
datasetParam	InteractionId	Mandatory	The ID of the interaction to be pulled.
	Reason (ReasonCode)	Optional	The operation reason.
	Description (ReasonValue)	Optional	The description of the reason.

Table 58: The Interaction Server Driver Device Commands (Continued)

OpenMediaTransfer			
Description	Transfers the interaction to another agent.		
stringParam	Not Used		
datasetParam	TrackingID	Optional	The driver identifier of the item. If the value is not specified, the driver uses, by default, the currently selected item.
	WorkinAgentId	Mandatory	The target agent login ID (employee ID in Genesys terms).
ChatSend			
Description	Sends a chat message. Applicable only for chat interactions.		
stringParam	Not Used		
datasetParam	TrackingID	Optional	The driver identifier of the item. If the value is not specified, the driver uses, by default, the currently selected item.
	ChatMsgText	Mandatory	The text to be sent.
ChatReleaseParty			
Description	Removes the specified party from the chat session.		
stringParam	Not Used		
datasetParam	TrackingID	Optional	The driver identifier of the item. If the value is not specified, the driver uses, by default, the currently selected item.

Table 58: The Interaction Server Driver Device Commands (Continued)

datasetParam (continued)	ChatAfterAction	Optional	The action to be performed with the chat session after removing the specified party. These actions can be: KEEP_ALIVE CLOSE_IF_NO_AGENTS FORCE_CLOSE See the *Multi-Channel Routing Chat Development Guide* for more information. The default value is KEEP_ALIVE.
	ChatUsrId	Optional	Assumes self, if this parameter value is not provided.
	ChatMsgText	Optional	Text to be sent (if any).

Interaction Server Device Commands (Changes for 8.0.1)

Table 59 on page 440 displays the Interaction Server driver device commands changes for the 8.0.1 release:

-

Note: This set of Interaction Server device commands is different than those that the Voice driver supports.

Table 59: The Interaction Server Driver Device Command Changes for 8.0.1

OpenMediaDeselect	
Description	Deselects the specified interaction and sends an OpenMediaDeselected event to Siebel. If the specified interaction was not selected, then no action is taken.
stringParam	Not Used

Table 59: The Interaction Server Driver Device Command Changes for 8.0.1 (Continued)

OpenMediaDeselect			
datasetParam	SelectedTrackingID	Mandatory	The driver identifier of the item to be selected.
	DeselectedTrackingID	Mandatory	The driver identifier of the item to be deselected.

Interaction Server Device Commands (Changes for 8.0.2)

Table 60 displays the Interaction Server driver device commands changes for the 8.0.2 release:

- ChatHold
- ChatResume

Table 60: The Interaction Server Driver Device Command Changes for 8.0.2

ChatHold			
Description	Places a chat interaction on hold and sends a ChatSuspended event. If no corresponding interaction found, then no action is taken.		
stringParam	Not Used		
datasetParam	TrackingID	Optional	The driver identifier of the interaction to be held.
	SelectedTrackingID	Optional	The driver identifier of the currently selected interaction. If the TrackingID parameter is not provided, then a chat interaction that is not the current one is held.
ChatResume			
Description	Resumes a chat interaction and sends a ChatResumed event. If no corresponding interaction found, then no action is taken.		
stringParam	Not Used		

Table 60: The Interaction Server Driver Device Command Changes for 8.0.2 (Continued)

datasetParam	TrackingID	Optional	The driver identifier of the interaction to be held. If none is provided, then the current interation is taken.

Outbound Contact Device Commands

This section provides the detailed descriptions of the following Outbound Campaign feature device commands:

- OpenMediaOcsCallCompletionStats
- OpenMediaOcsRecordProcessed
- OpenMediaOcsRecordReject
- OpenMediaOcsRecordCancel
- OpenMediaOcsRecordReschedule
- OpenMediaOcsRecordDoNotCall
- OpenMediaOcsChainedRecordReqst

Table 61 provides the details for each of these device commands:

Table 61: The Outbound Contact Device Commands

OpenMediaOcsCallCompletionStats	
Command Name	OpenMediaOcsCallCompletionStats
Description	The desktop sends an event to update the record details. Intermediate update.
OCS Action	Updates the record fields internally, waits for the next requests.
Mandatory Fields	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME GSW_RECORD_HANDLE
Additional Fields	Can be specified, updatable Genesys fields and user defined fields.

Table 61: The Outbound Contact Device Commands (Continued)

OpenMediaOcsRecordProcessed	
Command Name	OpenMediaOcsRecordProcessed
Description	The desktop sends an event to indicate that the record is processed. Outbound Contact Server should update the record information, if it is provided.
OCS Action	Updates a record and its chain in the database; uses all changes made by previous requests regarding the records in the chain. If a RecordProcessed event has the GSW_TREATMENT field correctly specified, Outbound Contact Server applies a treatment to the record.
Mandatory Fields	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME GSW_CALLING_LIST GSW_RECORD_HANDLE
OpenMediaOcsRecordReject	
Command Name	OpenMediaOcsRecordReject
Description	The desktop sends a request to indicate that the preview record or scheduled callback is not dialed by this agent. The record should be re-sent to another agent.
OCS Action	Places the push preview interaction back to the originating queue.
Mandatory Fields	TrackingID
Additional Fields	Not specified
OpenMediaOcsRecordCancel	
Command Name	OpenMediaOcsRecordCancel
Description	The desktop sends a request to indicate that a preview record or a scheduled callback should not be dialed. The record should not be re-sent to another agent. Should be marked in the database as Canceled.
OCS Action	Changes the record.

Table 61: The Outbound Contact Device Commands (Continued)

Mandatory Fields	GSW_APPLICATION_ID
	if (GSW_RECORD_HANDLE is specified){
	GSW_CAMPAIGN_NAME
	GSW_CALLING_LIST
	}
	else
	{
	GSW_PHONE
	GSW_CAMPAIGN_NAME (optional)
	}
	TrackingID

Additional Fields	GSW_CHAIN_ATTR = "AllChain" Or GSW_CHAIN_ATTR = "RecordOnly"
OpenMediaOcsRecordReschedule	
Command Name	OpenMediaOcsRecordReschedule
Description	Request a reschedule for a preview record, predictive call, or scheduled call.
OCS Action	Updates a record chain and reschedules the record
Mandatory Fields	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME GSW_RECORD_HANDLE GSW_DATE_TIME GSW_CALLBACK_TYPE TrackingID
Additional Fields	Not specified
OpenMediaOcsRecordDoNotCall	
Command Name	OpenMediaOcsRecordDoNotCall
Description	An agent requests that this number is not to be called again.
OCS Action	Specifies GSW_RECORD_HANDLE to update a record or a whole chain as DoNotCall, if GSW_CHAIN_ATTR == "AllChain". Adds phone(s) (GSW_CHAIN_ATTR == "AllChain") in a DoNotCall list.

Table 61: The Outbound Contact Device Commands (Continued)

Mandatory Fields	GSW_APPLICATION_ID
	1) if (GSW_RECORD_HANDLE is specified){
	GSW_CAMPAIGN_NAME
	GSW_CALLING_LIST
	}
	else
	{
	2) GSW_PHONE
	}
	else
	{
	3) GSW_CUSTOMER_ID
	}
	TrackingID

Additional Fields	Not specified
Comments	In the Siebel.def file, you can specify these attributes as command parameters: • USE_RECORD_HANDLE • USE_PHONE • USE_CUSTOMER_ID Tells the Adapter which specific set of mandatory fields is sent to Outbound Contact Server.
OpenMediaOcsChainedRecordReqst	
Command Name	OpenMediaOcsChainedRecordReqst
Description	A request to send all records from the chain that are defined by record handle.
OCS Action	Send the rest of a chain to the desktop.
Mandatory Fields	GSW_AGENT_REQ_TYPE GSW_APPLICATION_ID GSW_CAMPAIGN_NAME GSW_RECORD_HANDLE TrackingID
Additional Fields	Not specified

Multimedia Component Device Events

This section provides the detailed descriptions of the following Multimedia Component device events:

- Logged in
- Logged out
- Invited
- OpenMedia in service
- OpenMedia out of service
- Selected
- Deselected
- Released
- Accepted
- Submitted
- Pulled
- Workbin Content
- Workbin Content Changed
- Chat New Part
- Chat Re Enter
- Chat Party Left
- Chat Message
- Command Status

For information about the use and the syntax of these device events, refer to the *Siebel Communications Server Administration Guide* for your version of the Siebel Server.

Table 62 provides the details for each of these device commands:

Table 62: The Multimedia Component Device Events

Logged in		
Device Event	OpenMediaLoggedIn	
Description	Agent has been logged in	
TrackingID	Not applicable	
Attributes	TenantId	Tenant identifier to which agent belongs

Table 62: The Multimedia Component Device Events (Continued)

Logged out	
Device Event	OpenMediaLoggedOut
Description	Agent has been logged out from Open Media
TrackingID	Not applicable
Attributes	Not applicable
Invited	
Device Event	OpenMediaInvited
Description	Agent is invited to handle the specified interaction
TrackingID	Driver ID of the offered interaction
Attributes	Interaction properties
OpenMedia in service	
Device Event	OpenMediaInService
Description	OpenMedia capability of the Adapter is back in service (it is possible to execute OpenMedia requests)
TrackingID	Not applicable
Attributes	Not applicable
OpenMedia out of service	
Device Event	OpenMediaOutOfService
Description	The OpenMedia capability of the Adapter appears to be out of service, and it is not possible to execute any OpenMedia requests—possible reason: connection to Interaction Server has been lost.
TrackingID	Not applicable
Attributes	Not applicable
Selected	
Device Event	OpenMediaSelected
Description	Event sent when the driver changes the internally-selected work item, and is usually sent as a result of a successfully completed OpenMediaSelect command.
Attributes	Interaction properties of the selected item.

Table 62: The Multimedia Component Device Events (Continued)

Deselected	
Device Event	OpenMediaDeselected
Description	Event sent when the driver deselects a work item when another interaction is selected, and is usually sent as a result of successfully completed OpenMediaSelect and OpenMediaDeselect commands.
Attributes	Interaction properties of the selected item.
Released	
Device Event	OpenMediaReleased
Description	The interaction is released from the agent—for example, usually as a result of a transfer, a place in queue operation.
Attributes	Interaction properties
Accepted	
Device Event	OpenMediaAccepted
Description	The interaction is successfully accepted.
Attributes	Interaction properties of the accepted work item.
Submitted	
Device Event	OpenMediaSubmitted
Description	The interaction is created.
Attributes	Interaction properties of the submitted work item.
Pulled	
Device Event	OpenMediaPulled
Description	The interaction is pulled to the agent.
Attributes	Interaction properties of the pulled work item.
Workbin Content	
Device Event	OpenMediaWorkbinContent
Description	Deliver the content of the specified workbin.

Attributes	WorkbinId	Workbin identifier
	Content	Content of workbin

Table 62: The Multimedia Component Device Events (Continued)

Workbin Content Changed		
Device Event	OpenMediaWorkbinChanged	
Description	The content of the workbin is changed.	
Attributes	WorkbinId	Workbin identifier
	InteractionId	ID of the interaction
	InteractionType	Type of the interaction
	InteractionSubtype	Subtype of the interaction
	MediaType	Media type of the interaction
	Parent Id	ID of the parent interaction (if any)
	Operation	Type of operation if content changes
	UserData	User data
Chat New Part		
Device Event	ChatNewParty	
Description	New party was added to the chat session.	
Attributes	ChatUsrId	Chat user ID
	ChatUsrType	Chat user type (AGENT or CLIENT)
	ChatMsgType	Chat message type (such as TEXT)
	ChatUsrNick	Nickname of the user who joined the chat session
	ChatSerialNumber	Sequential number of the chat event
	ChatTimeshift	Time shift of the chat session
	ChatVisibility	Chat visibility (reserved)
	ChatStartedAt	Chat start time
Chat Re Enter		
Device Event	ChatReEnter	
Description	Party re-enters the chat session.	

Table 62: The Multimedia Component Device Events (Continued)

Attributes	ChatUsrId	Chat user ID
	ChatUsrType	Chat user type (AGENT or CLIENT)
	ChatMsgType	Chat message type (such as TEXT)
	ChatUsrNick	Nickname of the user who re-entered the chat session
	ChatSerialNumber	Sequential number of the chat event
	ChatTimeshift	Time shift of the chat session
	ChatVisibility	Chat visibility (reserved)
	ChatStartedAt	Chat start time
Chat Party Left		
Device Event	ChatPartyLeft	
Description	Chat user left the chat session	
Attributes	ChatUsrId	Chat user ID
	ChatUsrType	Chat user type (AGENT or CLIENT)
	ChatMsgType	Chat message type (such as TEXT)
	ChatUsrNick	Nickname of the user who left the chat session
	ChatSerialNumber	Sequential number of the chat event
	ChatTimeshift	Time shift of the chat session
	ChatVisibility	Chat visibility (reserved)
	ChatStartedAt	Chat start time
Chat Message		
Device Event	ChatMessage	
Description	Someone enters a new chat message.	

Table 62: The Multimedia Component Device Events (Continued)

Attributes	ChatUsrId	Chat user ID of message author
	ChatUsrType	Chat user type (AGENT or CLIENT) of message author
	ChatMsgType	Chat message type (such as TEXT)
	ChatMsgText	Content of the chat message
	ChatUsrNick	Nickname of the message author
	ChatSerialNumber	Sequential number of the chat event
	ChatTimeshift	Time shift of the chat session
	ChatVisibility	Chat visibility (reserved)
	ChatStartedAt	Chat start time
Command Status		
Device Event	OpenMediaCommandStatus	
Description	The command status is changed.	
Commands	The string describing the difference between the current and the previous command status.	

Outbound Campaign/Outbound Contact Server Device Events

This section provides the detailed descriptions of the Outbound Campaign device events, which is associated with the Genesys Outbound Contact Server (OCS) product.

The available Outbound Contact Server (OCS) device events for the G*plus* Outbound Campaign feature are:

- OpenMediaOcsRecordDoNotCallAck
- OpenMediaOcsRecordCancelAck
- OpenMediaOcsRecordProcessedAck
- OpenMediaOcsRecordRejectAck
- OpenMediaOcsRecordReschdulAck

Table 63 describes the available OCS device events for the G*plus* Outbound Campaign feature:

Table 63: Outbound Campaign/Outbound Contact Server Device Commands

OpenMediaOcsRecordDoNotCallAck	
Siebel CRM DeviceEvent Name	OpenMediaOcsRecordDoNotCallAck
OCS Protocol User Event	DoNotCallAcknowledge
Description	Confirmation that DoNotCall was accepted.
TrackingID	TrackingID of the work item
Desktop Action	Kills the record and the chain if DoNotCallAcknowledge contains the following: GSW_CHAIN_ATTR != "RecordOnly".

Attributes	GSW_APPLICATION_ID
	if (GSW_RECORD_HANDLE is specified){
	GSW_CAMPAIGN_NAME
	GSW_CALLING_LIST
	GSW_RECORD_HANDLE
	}
	else
	{
	GSW_PHONE
	else
	{
	GSW_CONTCT_ID
	}

OpenMediaOcsRecordCancelAck	
Siebel CRM DeviceEvent Name:	OpenMediaOcsRecordCancelAck
OCS Protocol User Event:	RecordCancelAcknowledge
Description	OCS accepts a desktop request to cancel a record.
TrackingID	TrackingID of the work item
Desktop Action	Kills the record and the chain if RecordCancelAcknowledge contains the following: GSW_CHAIN_ATTR == "AllChain".

Table 63: Outbound Campaign/Outbound Contact Server Device Commands (Continued)

Attributes	GSW_APPLICATION_ID
	if (GSW_RECORD_HANDLE is specified) {
	GSW_CAMPAIGN_NAME
	GSW_CALLING_LIST
	GSW_RECORD_HANDLE
	}
	else
	{
	GSW_PHONE
	}

OpenMediaOcsRecordProcessedAck	
Siebel CRM DeviceEvent Name:	OpenMediaOcsRecordProcessedAck
OCS Protocol User Event:	RecordProcessedAcknowledge
Description	OCS confirms that record has been executed.
TrackingID	TrackingID of the work item
Desktop Action	Kills the record and the chain, if requested.
Attributes	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME GSW_CALLING_LIST GSW_RECORD_HANDLE
OpenMediaOcsRecordRejectAck	
Siebel CRM DeviceEvent Name:	OpenMediaOcsRecordRejectAck
OCS Protocol User Event:	N/A
Description	Push preview interaction is placed into the originating queue.
TrackingID	TrackingID of the work item
Desktop Action	Kills the record and the chain if RecordCancelAcknowledge contains the following: GSW_CHAIN_ATTR == "AllChain".
Attributes	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME GSW_CALLING_LIST GSW_RECORD_HANDLE

Table 63: Outbound Campaign/Outbound Contact Server Device Commands (Continued)

OpenMediaOcsRecordReschdulAck	
Siebel CRM DeviceEvent Name:	OpenMediaOcsRecordRescheduIAck
OCS Protocol User Event:	RecordRescheduleAcknowledge
Description	Confirmation that the record was rescheduled.
TrackingID	TrackingID of the work item
Desktop Action	Continue call work.
Attributes	GSW_APPLICATION_ID GSW_CAMPAIGN_NAME GSW_CALLING_LIST GSW_RECORD_HANDLE GSW_DATE_TIME GSW_CALLBACK_TYPE

Multimedia Component Device Events (Changes for 8.0.2)

This section provides the detailed descriptions of the following Multimedia Component device events:

- ChatSuspended
- ChatResumed

Note: These events are intended only for Siebel Chat UI support. They have no purpose for other medias.

Table 64: Interaction Server Driver Device Events Introduced in 8.0.2

Chat is held or suspended	
Device Event	ChatSuspended
Description	A chat interaction is suspended.

**Table 64: Interaction Server Driver Device Events Introduced in 8.0.2
(Continued)**

Attributes	InteractionId	ID of the interaction.
	InteractionType	Type of interaction.
	ParentId	ID of the parent interaction (if any).
	InteractionSubtype	Subtype of the interaction.
	MediaType	Media type of the interaction.
	SubmittedAt	Interaction submission time.
	ChatStartedAt	Chat start.
	UserData	User data.
Chat is resumed		
Device Event	ChatResumed	
Description	A chat interaction is resumed.	
Attributes	InteractionId	ID of the interaction.
	InteractionType	Type of interaction.
	ParentId	ID of the parent interaction (if any).
	InteractionSubtype	Subtype of the interaction.
	MediaType	Media type of the interaction.
	SubmittedAt	Interaction submission time.
	ChatStartedAt	Chat start.
	UserData	User data.

9 Deploying the Media Routing Component

This chapter describes how to configure and install the G*plus* Adapter for Siebel CRM Media Routing Component. The chapter includes the following sections:

Overview

Installation and configuration of the G*plus* Adapter for Siebel CRM Media Routing Component includes:

- Compiling and deploying the Siebel repository
- Configuring and activating business workflows
- Configuring Siebel eMail Response
- Configuring Siebel

New in This Release

This section provides information about new features or functionality in the G*plus* Adapter for Siebel CRM Media Routing Component.

Release 8.0.210 The following new features or functionality are included in this release:

- HTTP Authentication. See "Deploying Basic HTTP Authentication" on page 557 for details.

Release 8.0.2 The following new features or functionality are included in this release:

- Support for Siebel versions 8.1.1.14/8.2.2.14 (IP2014) for both HI and Open UI modes.

- Support for Genesys Framework 8.5.

- Support for Red Hat Enterprise Linux 6.

- Support for Red Hat Enterprise Linux 7.

- Support for AIX 7.1.

- Support for Solaris 11.

Release 8.0.110 The following new features or functionality are included in this release:

- Support for Siebel versions 8.1.1.11/8.2.2.4 (IP2013) for both HI and Open UI modes.

Release 8.0.1 No new features were introduced in this release.

Release 8.0.0 The following new features or functionality are included in this release:

- The G*plus* Adapter for Siebel CRM Media Routing Component no longer uses the G*plus* Communication Server to send routing requests to the Genesys environment, but rather uses the G*plus* Open Media Server.

- The G*plus* Adapter for Siebel CRM Media Routing Component contains a dedicated server for routing requests—the G*plus* Open Media Server for Siebel CRM.

- Support for the Red Hat Enterprise Linux (RHEL) operating system.

Installing the Media Routing Component

The G*plus* Adapter for Siebel CRM Media Routing Component can be installed on either Windows or UNIX platforms.

Note: The G*plus* Adapter for Siebel CRM Media Routing Component uses some of the same functionality that the G*plus* Adapter for Siebel CRM Multimedia Component does, so the G*plus* Adapter for Siebel CRM Media Routing Component should be installed *after* the G*plus* Adapter for Siebel CRM Multimedia Component. The G*plus* Adapter for Siebel CRM Media Routing Component will not function without the G*plus* Adapter for Siebel CRM Media Routing Component.

The G*plus* Adapter for Siebel CRM Media Routing Component uses the same communications driver names and profiles as the G*plus* Adapter for Siebel CRM Multimedia Component. However, if you plan to use the Media Routing Component for routing Siebel work items in addition to Siebel eMail, you will need to make some changes in the communication configuration. For more information, see the section, "The Media Routing Component uses the same driver as the Multimedia Component, so the device commands and events are the same as for the Multimedia Component. For a list of the device commands and events, refer to Chapter 8, "Deploying the Multimedia Component," on page 391." on page 502.

Installation Prerequisites

The following are prerequisites for installation of the G*plus* Adapter for Siebel CRM Media Routing Component:

- Back up the Siebel database and the Siebel repository (*.srf) file.

- Install the G*plus* Multimedia Component (see Chapter 8, "Deploying the Multimedia Component," on page 391).

Installation

This section describes how to install the G*plus* Adapter for Siebel CRM Media Routing Component on either a Windows or a Linux operating systems:

Note: Do not use special symbols in any destination directory name when installing G*plus* Adapter for Siebel CRM Media Routing Component in an UNIX environment.

Procedure:
Media Routing Component: Installing the Media Routing Component

Purpose: To install the G*plus* Adapter for Siebel CRM Media Routing Component on either a Windows or a Linux operating system.

Start of procedure

1. To start the installation process, run the setup.exe file (for Windows) or run the install.sh file (for UNIX) from the G*plus* Adapter for Siebel CRM Media Routing Component installation package.

2. Enter the path to the destination directory for installation, so that the installation program installs the G*plus* Open Media Server for Siebel CRM (follow the steps in Procedure: Media Routing Component: Installing the Open Media Server, on page 470) and has the correct folder location to place the Siebel archive (*.sif) files, the Siebel SmartScripts, the Siebel image files, and the general product information.

End of procedure

Next Steps

• Configure the Genesys part of the G*plus* Adapter for Siebel CRM Media Routing Component. See the section, "Configuring Genesys" on page 460.

Configuring Genesys

This chapter describes how to configure and install the G*plus* Open Media Server for Siebel CRM and consists of the following sections:

Prestart Information

Before starting the configuration process you should have the following Genesys Framework applications running:

- Configuration Database
- Configuration Server
- Configuration Manager

As a Configuration Manager user, you should have sufficient privileges to make changes to Configuration Layer objects.

Overview

The G*plus* Open Media Server for Siebel CRM implements open media routing functionality. It acts as a dedicated server that redirects requests from Siebel CRM to the underlying Genesys communication middleware—Open Media Interaction Server. The G*plus* Open Media Server for Siebel CRM may run on different hosts with different platforms and operating systems.

Note: The process of configuring and installing the G*plus* Open Media Server for Siebel CRM consists of:

- Configuring Genesys
- Installing Open Media Server

The G*plus* Open Media Server for Siebel CRM is implemented as a Genesys server, and is represented by an Application object in Genesys Configuration Manager.

The following procedures describe how to configure the Genesys section of the G*plus* Open Media Server for Siebel CRM and create the Application object in Genesys Configuration Manager.

Importing the Open Media Server Application Template

This section describes the procedures for importing the G*plus* Open Media Server for Siebel CRM Application template.

Recommendations

Genesys recommends using an Application Template when you are configuring your Adapter. The Application Template for your Adapter contains the most important configuration options set to the values recommended for the majority of environments. When modifying configuration options for your Adapter later in the process, you can change the values inherited from the template rather than create all the options by yourself.

Procedure:
Media Routing Component: Importing the Open Media Server for Siebel CRM Application template

Purpose: To import the G*plus* Open Media Server for Siebel CRM Application template.

Start of procedure

1. In Configuration Manager, under Environment, right-click the Application Templates folder.

2. Select the Import Application template.

3. Browse to and select the Application template for the G*plus* Open Media Server.

4. Select the following template, as follows:
   - Release 8.0.0: Gplus_OpenMedia_Server_for_SiebelCRM_800.apd
   - Release 8.0.1: Gplus_OpenMedia_Server_for_SiebelCRM_801.apd
   - Release 8.0.2: Gplus_OpenMedia_Server_for_SiebelCRM_802.apd

5. Click Open.

   The Properties dialog box for the Application Template object displays.

6. Click OK to accept the default values.

   The Application Template object has been imported to the Genesys Configuration Layer.

End of procedure

Next Steps

- Create the Configuration Layer Application object for the G*plus* Open Media Server for Siebel CRM. See the section, "Creating the Open Media Server for Siebel CRM Application Object".

Creating the Open Media Server for Siebel CRM Application Object

This section describes how to create the G*plus* Open Media Server for Siebel CRM Application object.

Procedure:
Media Routing Component: Creating the Open Media Server for Siebel CRM Application object

Purpose: To create the Application object for the G*plus* Open Media Server for Siebel CRM.

Start of procedure

1. In Configuration Manager, under Environment, right-click the Applications folder.

2. Select New > Application.

3. Select the Application template that you just created:
    *   Release 8.0.0: Gplus_OpenMedia_Server_for_SiebelCRM_800.apd
    *   Release 8.0.1: Gplus_OpenMedia_Server_for_SiebelCRM_801.apd
    *   Release 8.0.2: Gplus_OpenMedia_Server_for_SiebelCRM_802.apd

4. Click OK.

    The Properties dialog box for the Application appears.

End of procedure

Next Steps

*   Configure the tabs in the Properties dialog box for the G*plus* Open Media Server for Siebel CRM. See the section, "Configuring the Tabs in the Properties Dialog Box".

Configuring the Tabs in the Properties Dialog Box

This section describes how to configure the tabs in the Properties dialog box, arranged in the order in which they display. The first tab is the General tab (see Figure 36 on).

Procedure:
Media Routing Component: Configuring the tabs in the Properties dialog box

Purpose: To configure the tabs in the Properties dialog box for the G*plus* Open Media Server for Siebel CRM.

Start of procedure

General Tab   1.   Click the General tab in the Properties dialog box (see Figure 36).
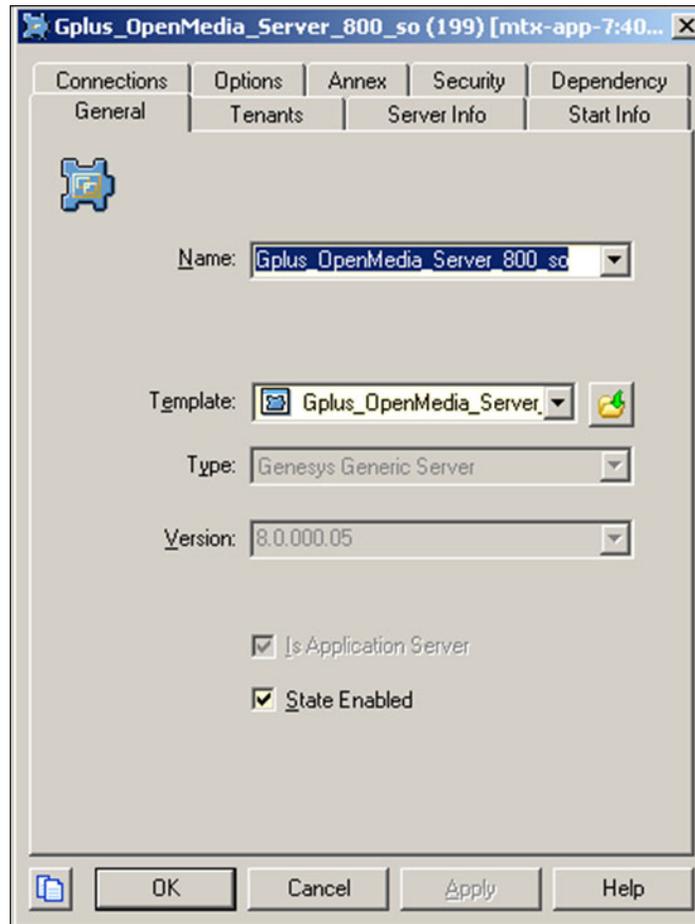
Figure 36: View of the General Tab in a Single-Tenant Environment

2.   In the Name list, click the name of the Application object you are configuring.

3.   In the Template list, click the name of the template you are configuring or use the Browse button to select the template you are configuring.

4.   Click Apply.

5.   Next, if you are working in a multi-tenant environment, go to the Tenants Tab; otherwise, go to the Server Info Tab.

Note: The Tenants tab only displays, if you are working in a multi-tenant environment.

Tenants Tab   6. Select the Tenants tab (see Figure 37).
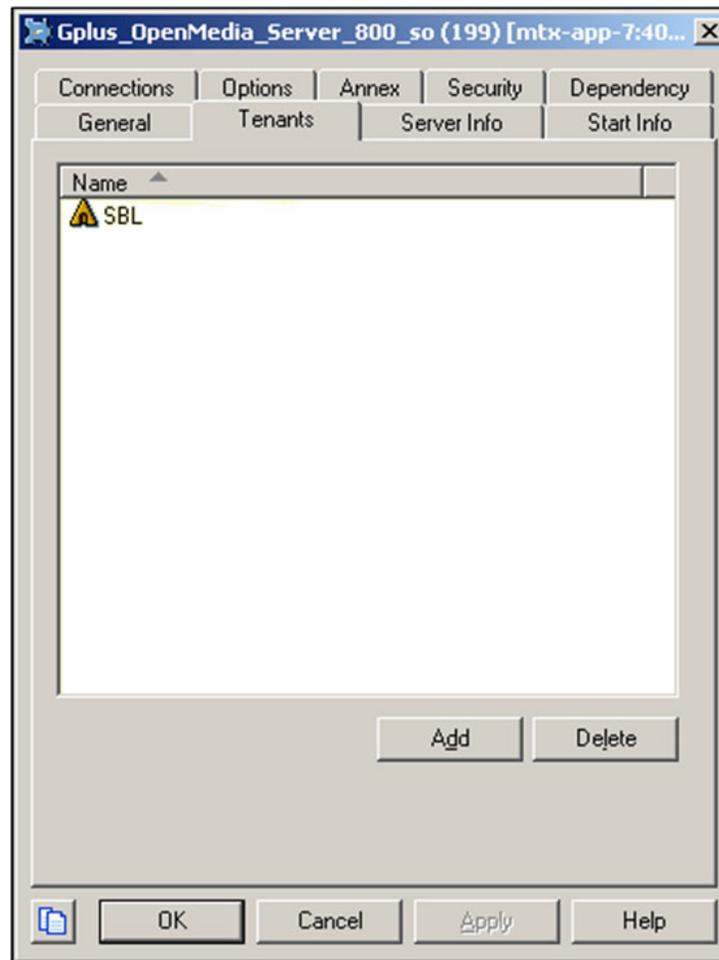
Figure 37: View of the Tenants Tab

7. Select the Genesys Tenants under which the objects that are exported from Siebel are created.

8. Click Add.

Server Info Tab   9. Select the Server Info tab.

10. In the Host field, use the Browse button to select the host where you are installing the G*plus* Open Media Server and click OK.

11. In the Communications Port field, enter any valid port number for the port with the ID of default. Genesys recommends setting this value to 18101.

Note: This option determines the HTTP port number used by the G*plus* Open Media Server to receive HTTP packets from Siebel. Use the value that you enter for this option when you configure the Siebel part of the G*plus* Open Media Server.

Start Info Tab   12. Select the Start Info tab.

13. In the Working Directory field, enter the full name of the Component installation directory on the host you specified on the Server Info tab. The value you enter in this field will be used as the default destination folder during installation.

14. Enter any valid value into each of the following fields:

- Command Line
- Command Line Arguments

The values that you enter in these fields are overwritten during installation; however, data must be present in these fields during the configuration process.

15. Leave the default values for the remaining fields.

Connections Tab   **16.** Select the Connections tab.

Notes: • This step is *mandatory* for configuring the G*plus* Open Media Server to work with the Media Routing Component. You may also configure a connection to the Genesys Message Server.

- The content of the Connections tab depends on the Adapter components being used. For Media Routing components, it is necessary to add a connection to the Interaction Server.

---

17. On the Connections tab, click Add to attach a connection to the Interaction Server (see Figure 38 on page 467).

- Optionally, you can add a connection to the Genesys Message Server.
- Optionally, you can add a connection to the Genesys Configuration Server.
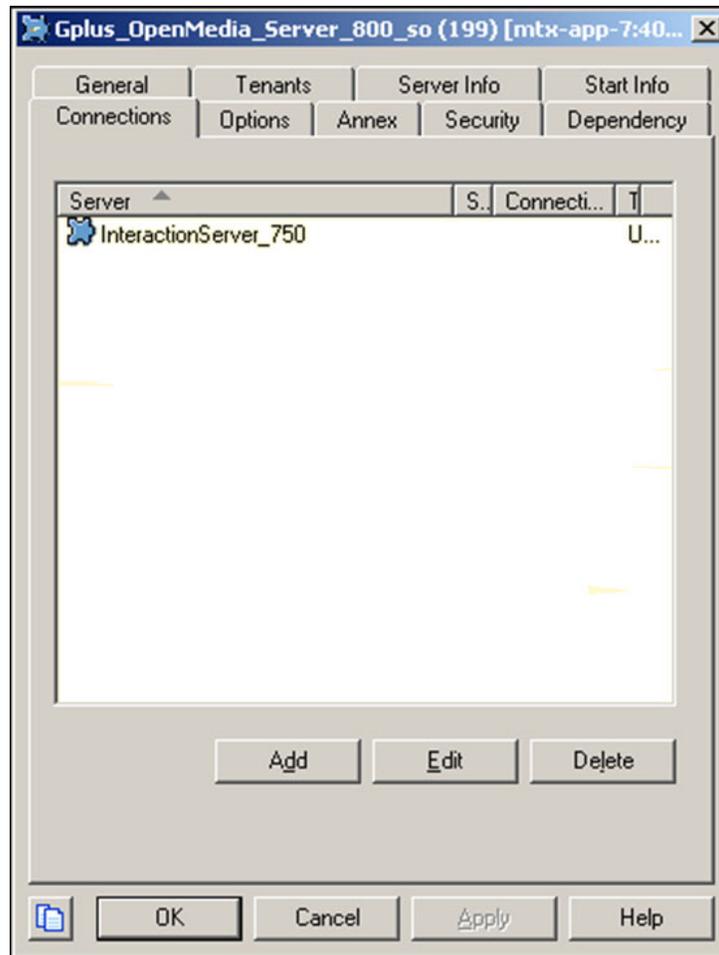
Figure 38: View of Connections Tab in a Multi-Tenant Environment

Options Tab   **18.** Select the Options tab (see Figure 39 on).
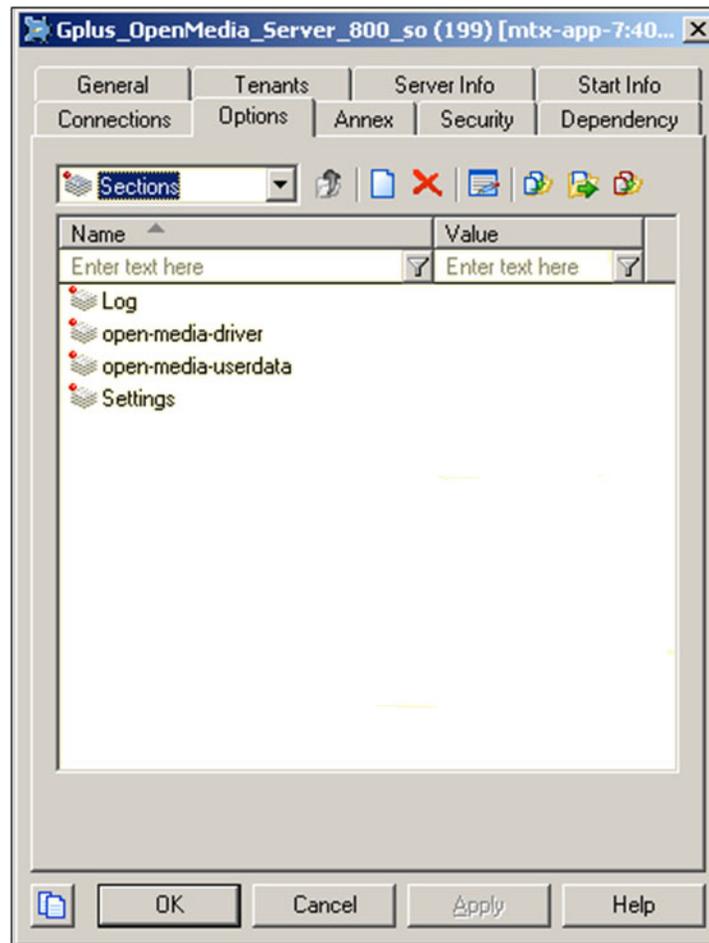
Figure 39: View of Options Tab

In the Sections pane, the following sections are listed:

- Log
- open-media-driver
- open-media-userdata
- Settings

19. Double-click the Log section to start.

20. Configure the configuration options described in the following section, "Setting the Genesys Configuration Options for the Open Media Server for Siebel CRM" on page 469.

End of procedure

Next Steps

- Set the G*plus* Open Media Server's configuration options. See, "Setting the Genesys Configuration Options for the Open Media Server for Siebel CRM".

Setting the Genesys Configuration Options for the Open Media Server for Siebel CRM

Unless specified otherwise, set the G*plus* Open Media Server for Siebel CRM configuration options in the Options tab of the Application object using the following navigation path:

- In Genesys Adminstrator—Application object > Options tab > Advanced View (Options).

- In Configuration Manager—Application object > Properties dialog box > Options tab.

For ease of reference, the options have been arranged in alphabetical order within their corresponding sections:

Changes to configuration options take effect after the Open Media Server restarts.

Log Section

The G*plus* Open Media Server for Siebel CRM supports a common set of log options to allow the precise configuration of the log file output. For a complete list of Common Log options and their descriptions, see the "Common Log Options" chapter of the *Framework 8.0 Configuration Options Reference Manual*.

Note: If you do not specify any log options, the default values apply.

Selective Protection of Sensitive Data in Logs

To protect against displaying the *private* parameters in the UserData, Extensions attributes, and the Reason_Code parameters in the Adapter's log, a number of options can be configured in the Options section of the G*plus* Open Media Server Application object in Configuration Manager.

These options are described in the Log-Filter and Log-Filter-Data sections of the *Framework Configuration Options Reference Manual*.

For more information, see the section, Chapter 3, "Log Section," on page 57.

open-media-driver and open-media-userdata Sections

See the sections, "open-media-driver Section" on page 59 and "open-media-userdata Section" on page 60 for examples of how to configure the options from the open-media-driver and open-media-userdata sections.

Note: The open-media-driver and open-media-userdata configuration options must not be changed.

Settings Section

The Settings section has the following option that can be configured:

MaxProcessingThreads Option

Default Value: 5
Valid Value: Any positive integer greater than or equal to (>=) 4

Provides the value for the maximum number of allowed threads.

Installing the Open Media Server

This section describes the installation process for the G*plus* Adapter 8.0 Open Media Server for Siebel CRM and is divided into the following sub-sections:

- Installing the Open Media Server in a Windows and a UNIX Environment, page 470
- The Target Directory Structure and File Locations, page 471
- Uninstalling the Open Media Server, page 471

Installing the Open Media Server in a Windows and a UNIX Environment

This section describes how to install the G*plus* Open Media Server on a Windows or a UNIX operating system.

Note: Do not use any special symbols in the destination directory name when installing in a UNIX environment.

Procedure:
Media Routing Component: Installing the Open Media Server

Purpose: To install the G*plus* Adapter Open Media Server for Siebel CRM on a Windows or a UNIX operating system.

Start of procedure

1. To start the installation process, run the setup.exe file (for Windows), or the install.sh file (for UNIX) from the G*plus* Open Media Server installation package.

2. Answer the installation questions according to your selected deployment scenario.

End of procedure

The Target Directory Structure and File Locations

The following generic directory names are used in the description of the directories structure:

- <Destination Directory> is the destination directory for installation used by the installation script to copy the Adapter component files.

- <Application Name> is the subdirectory in the <Destination Directory> where all files copied by installation are placed. It is the name of the Application object of the G*plus* Open Media Server for Siebel CRM.

- Files in subdirectories <Destination Directory>/<Application Name>/7.7 are related to Siebel versions 7.7/7.8.

- Files in subdirectories <Destination Directory>/<Application Name>/8.0 are related to Siebel version 8.0.

- Files in subdirectories <Destination Directory>/<Application Name>/8.1 are related to Siebel version 8.1.

- Files in subdirectories <Destination Directory>/<Application Name>/8.1_8.2_OUI are related to Siebel version 8.1.1.11/8.2.2.4 (IP2013).

- Files in subdirectory <Destination Directory>/<Application Name>/IP2014 are related to Siebel 8.1.1.14/8.2.2.14 (IP2014).

Uninstalling the Open Media Server

Complete information about uninstalling the G*plus* Open Media Server and its components is provided in Chapter 14, "Uninstallation Instructions," on page 565.

Configuring Siebel

This section describes how to configure the Siebel part of the G*plus* Open Media Server.

Configuring the Siebel section of the G*plus* Open Media Server consists of the following sub-sections:

Prestart Information

Before starting this part of the configuration process, you must make sure to do the following:

- You must have the Siebel Tools application running.
- You should be connected to the local copy of the Siebel Server database.
- You should have sufficient privileges to check-in and check-out projects from the Siebel Server repository.
- You should ensure that all required steps from "Patching and Configuring Siebel CRM" are performed.

Configuring Siebel Using the Siebel Tools

Use the Siebel Tools to compile an updated version of the Siebel repository file (SRF or *.srf file) for one, or more, of the Siebel applications that you use on your Siebel Server, which you will then deploy on the server. For more information about using Siebel Tools, see the Siebel documentation.

Configuring Siebel using the Siebel Tools consists of the following sub-sections:

Checking Out Existing Projects from the Siebel Repository for the Media Routing Component

The Siebel implementation of the Media Routing Component makes use of a number of objects provided by Siebel. Use the procedure to enable modifications of these objects. To do so, you will need to check-out the corresponding projects from the Siebel repository.

Procedure:
Media Routing Component: Checking out existing projects

Purpose: To check-out existing projects from the Siebel repository.

Start of procedure

1. In Object Explorer, start Siebel Tools against the local database.

2. In Siebel Tools, select Tools > Check Out...

    The Check Out dialog box displays.

3. In the Projects list, select the GplusMediaRouting project to be checked-out.

End of procedure

Next Steps

*   Import the GplusMediaRouting.sif archive file. See, Procedure: Media Routing Component: Importing the GplusMediaRouting.sif archive file.

Procedure:
Media Routing Component: Importing the GplusMediaRouting.sif archive file

Purpose: To import the GplusMediaRouting.sif archive file.

Start of procedure

1. Start Siebel Tools.

2. Lock the Genesys Multimedia project and lock the GplusMediaRouting project, if it already exists.

3. In Siebel Tools, select Tools > Import from Archive... .

4. In the Select Archive to Import window, browse to one of the following <Installation Directory> paths, depending on your version of Siebel:
    *   <Installation Directory>/7.7 for Siebel Server versions 7.7/7.8
    *   <Installation Directory>/8.0 for Siebel Server version 8.0
    *   <Installation Directory>/8.1 for Siebel Server version 8.1
    *   <Installation Directory>/8.1_8.2_OUI for Siebel Server version 8.1.1.11/8.2.2.4 (IP2013)
    *   <Installation Directory>/IP2014 for Siebel Server version 8.1.1.14/8.2.2.14 (IP2014)

5. Select the GplusMediaRouting.sif archive file.

6. Click Open.

   The Import Wizard–Preview window displays.

7. In the Import Wizard–Preview window, in the Conflict resolution section, select Merge the object definition from the archive file with the definition in the repository.

8. Click Next.

   The Import Wizard–Review Conflicts and Actions window displays.

9. Click Next.

   The "Do you wish to proceed?" window displays.

10. Merge any conflicts, if they correspond to your needs.

11. Click Yes.

   The objects from the archive are imported into the Siebel repository.

12. Click Finish to complete the import.

End of procedure

Next Steps

• Compile the Siebel repository file. See the section, "Compiling, Updating, and Deploying the Siebel Repository File".

Compiling, Updating, and Deploying the Siebel Repository File

You can either compile all of the projects, or you can only compile the locked projects.

Procedure:
Media Routing Component: Compiling the Siebel Repository File

Purpose: To compile the Siebel repository file.

Start of procedure

1. In Siebel Tools, select Tools > Compile Projects...

2. Select Locked projects.

3. In the Siebel Repository File edit box, enter the name of the repository file.

4. Click Compile.

   The status bar at the bottom of the Object Compiler window indicates when the compilation is finished.

5. When the compilation is finished, close Siebel Tools.

End of procedure

Next Steps

- Set up the business service query access for the GplusMediaRoute and the GplusMediaRouteIXN business services. See, Procedure: Media Routing Component: Setting the business service query access for the GplusMediaRoute and the GplusMediaRouteIXN business services.

Procedure:
Media Routing Component: Setting the business service query access for the GplusMediaRoute and the GplusMediaRouteIXN business services

Purpose: To set up the business service query access for the GplusMediaRoute and the GplusMediaRouteIXN business services.

Start of procedure

1. Log into the Siebel Server as a Siebel administrator.

2. Navigate to Administration > Server Configuration > Enterprises.

3. On the Component Group tab, select the required component—for example, Siebel Financial Services.

4. Switch to the Parameters tab.

5. Navigate to the Business Service Query Access List record.

6. Add GplusMediaRoute and GplusMediaRouteIXN business services to the end of the record value, separated by a comma.

7. Restart the Siebel Server to make the changes effective.

End of procedure

Next Steps

- Deploy the Siebel repository file. See, Procedure: Media Routing Component: Deploying the Siebel repository file, on page 476.

Procedure:
Media Routing Component: Deploying the Siebel repository file

Purpose: To deploy the Siebel repository file.

Start of procedure

1. Stop the Siebel Server.
2. Back up the original Siebel repository file.
3. Copy the compiled repository file into the proper location, so that you can use it instead of the original Siebel repository file.
4. Start the Siebel Server.

End of procedure

Configuring Siebel Using the Siebel Web Client

This section describes how to configure the Siebel application using the Siebel Web Client and consists of the following sub-sections:

- Siebel prior to 8.1.1.11/8.2.2.4: Configuring and Importing a Siebel Workflow Process, page 477
- Siebel prior to 8.1.1.11/8.2.2.4: Deploying the Workflow Processes on the Siebel Server, page 481
- Siebel 8.1.1.11/8.2.2.4 or later: Configuring and Deploying Siebel Workflow Processes, page 482
- Creating a Siebel Enterprise Server Named Subsystem, page 488
- Configuring the Multimedia Component Agents and Communications, page 490
- Configuring the Siebel eMail Response, page 492

Note: You must connect to your Siebel Server using the Siebel Web Client.

Siebel prior to 8.1.1.11/8.2.2.4: Configuring and Importing a Siebel Workflow Process

This section describes how to configure and import a Siebel workflow process. A Siebel workflow is a series of Siebel business services that are linked to complete a task.

Notes:
- This procedure is required only if you are using the G*plus* Adapter for Siebel CRM Media Routing Component for the routing of Siebel eMail.
- Use the following steps to make any changes to the Siebel repository file that are relevant to the Siebel business process:
  - Use Siebel Tools to make the changes to the repository file.
  - Export the updated repository file into an *.XML file.
  - Use the Siebel Client to import the updated repository file.
- This procedure only applies to Siebel Server versions prior to 8.1.1.11.

Procedure:
Media Routing Component: Configuring and importing a Siebel workflow process

Purpose: To import and configure a Siebel workflow process for a series of Siebel business services.

Note: When importing a workflow process, if you encounter the following error: This operation is not allowed when there are no records displayed. Please execute a query that returns at least one record or add a new record, refer to the following Siebel error code—Alert 594.

Start of procedure

Siebel Tools usage:

1. Lock the EMR Workflow project.
2. Using the Object Explorer, choose the Workflow Process tab.

3. Import the GplusMediaRouting-ProcessMessage.xml file, as shown below in Figure 40.

Figure 40: Siebel Tools Usage

4. Define the string values for the following properties according to your environment (see Figure 41):

- ConnectionName—The ConnectionName value is the Application name of a Genesys Interaction Server as specified on the Connections tab for the G*plus* Open Media Server.
- SubmitQueue—The SubmitQueue value is the name of the queue to which the interaction should be submitted.

5. If you use the nonreal-time (background) mode for e-mail processing, set the value of the RoutingMediaType property to BackgroundEmail. This property is used as a parameter for the GplusMediaRouteIXN business

service. For more information about these properties, see the "Media Routing Component Customization" section of the *Gplus Adapter 8.0 for Siebel CRM Developer's Guide*.
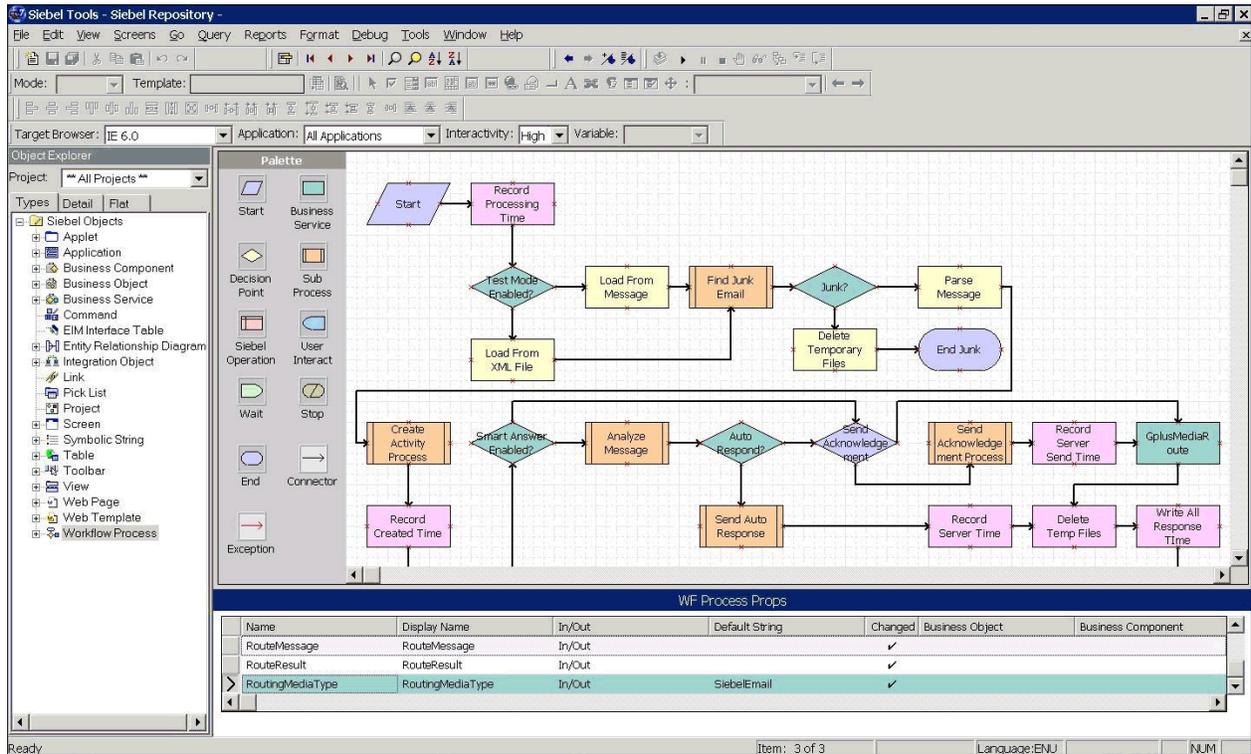
Figure 41: Defining Workflow Properties

> **6.** To add AttachedUserData values to a routing request, you must add custom input arguments for the GplusMediaRouting workflow step. To add these values, do the following:
>
> **a.** In Design view, select the GplusMediaRouting box (see Figure 41).
>
> **b.** Right-click on GplusMediaRouting.
>
> **c.** From the menu, select Show Input Arguments (see Figure 42 on page 480).

d. In the Input Arguments view, add a new record.

All input arguments, except the predefined arguments, are attached to a routing request as AttachedUserData.
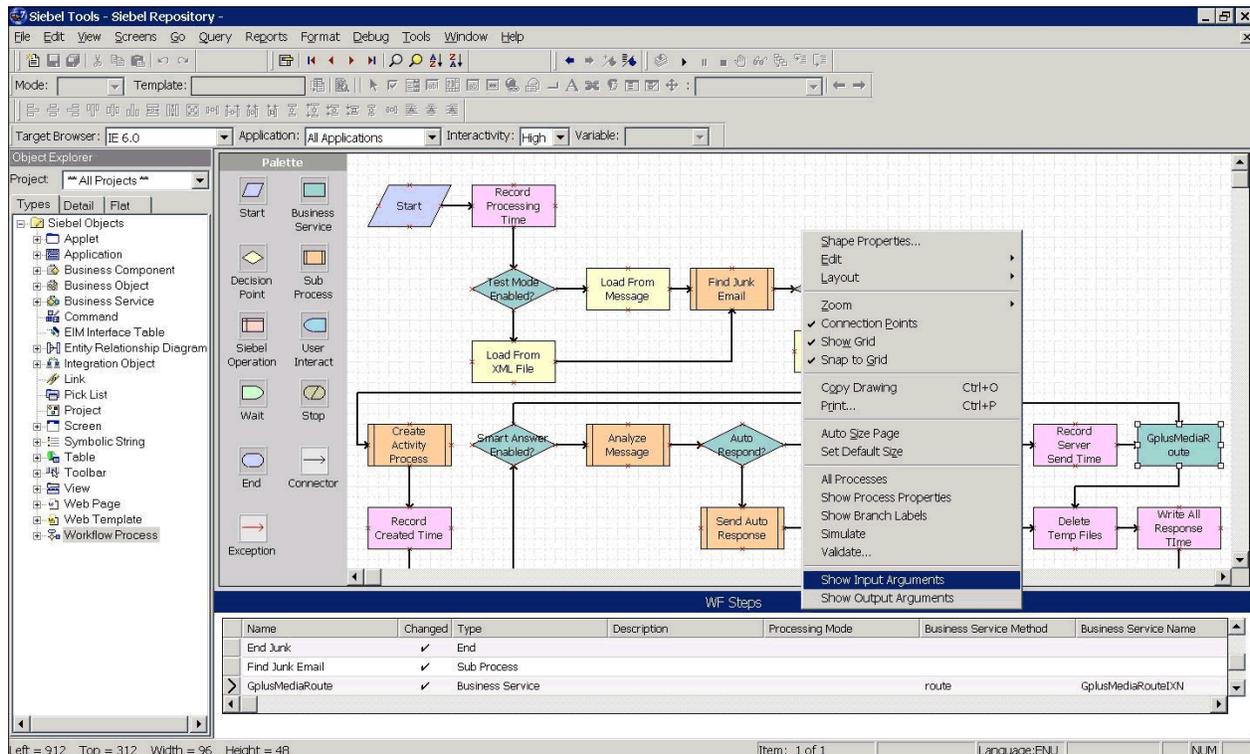
Figure 42: Editing Input Arguments for GplusMediaRouting Workflow Step

7. Deploy the workflow process.

8. Export the workflow process into a file. For example, you could refer to this process as <SEWorkflow.XML>.

9. Compile and apply the *.srf file.

End of procedure

Next Steps

- Deploy the workflow processes on the Siebel Server. See the section, "Siebel prior to 8.1.1.11/8.2.2.4: Deploying the Workflow Processes on the Siebel Server" on .

Siebel prior to 8.1.1.11/8.2.2.4: Deploying the Workflow Processes on the Siebel Server

The workflow processes help automate the business processes and they are defined within Siebel to emit or to receive Siebel *. XML messages.

Note: This procedure is required only if you are using the *Gplus* Adapter for Siebel CRM Media Routing Component for the routing of Siebel eMail.

Note: This procedure only applies to Siebel Server versions prior to 8.1.1.11.

Procedure:
Media Routing Component: Activating the Siebel Workflow Process

Purpose: To activate the Siebel workflow process.

Start of procedure

1. Log into the Siebel Client as a Siebel Administrator.

2. Import the <SEWorkflow.XML> file exported previously from the Siebel Tools into the Active Workflow Processes list/set, as shown in Figure 43 on .
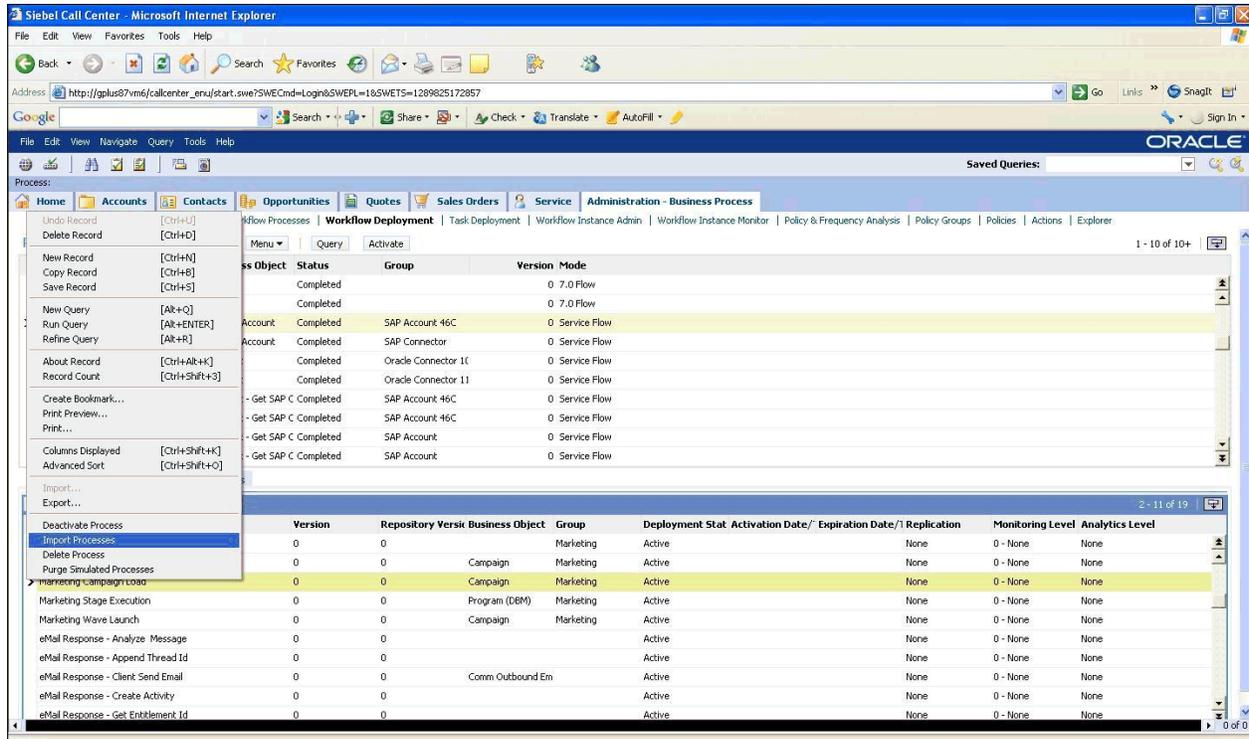
Figure 43: Business Process - Workflow Process Import

3. Make sure that the workflow is active. If not, select the workflow process you want to activate and click Activate.

End of procedure

Next Steps

• Create a Siebel Enterprise Server Named Subsystem. See the section, "Creating a Siebel Enterprise Server Named Subsystem".

Siebel 8.1.1.11/8.2.2.4 or later: Configuring and Deploying Siebel Workflow Processes

This section describes how to configure and deploy two Siebel workflow processes required to apply Gplus Media Routing to Siebel email processing.

Notes: • This procedure is required only if you are using the G*plus* Adapter for Siebel CRM Media Routing Component for the routing of Siebel eMail.

• This procedure only applies to Siebel Servers version 8.1.1.11/8.2.2.4 or later.

Procedure:
Media Routing Component: Updating or cloning the Siebel email processing workflow process

Purpose: To configure and deploy Siebel workflow processes.

Start of procedure

1. Open the Siebel Tool against the Siebel Server DB and log in as a Siebel Administrator.

2. Lock the EMR Workflow project.

3. Using the Object Explorer, choose the Workflow Process tab.

4. In the Workflow Processes list, query for a process named eMail Response - Process Message.

5. You can modify the selected Siebel workflow process directly or create and modify a copy. For the purpose of this procedure, describe the modified workflow as <Gplus Email Process Message>.

6. If you want to modify the vanilla Siebel workflow process, select eMail Response - Process Message from the Workflow Processes list and click Revise.

7. In the Workflow Processes list, right-click <Gplus Email Process Message> and choose Edit Workflow Process. The Workflow Process Designer appears.

8. Replace the Route Message sub-process block with a new business service step block, with the following properties:

Table 65: Business Service Step Properties

Parameter	Value
Name	GplusMediaRoute
Business Service Name	GplusMediaRouteIXN
Business Service Method	route

Configure the block connections with the same settings as the previous block (see Figure 44 on).
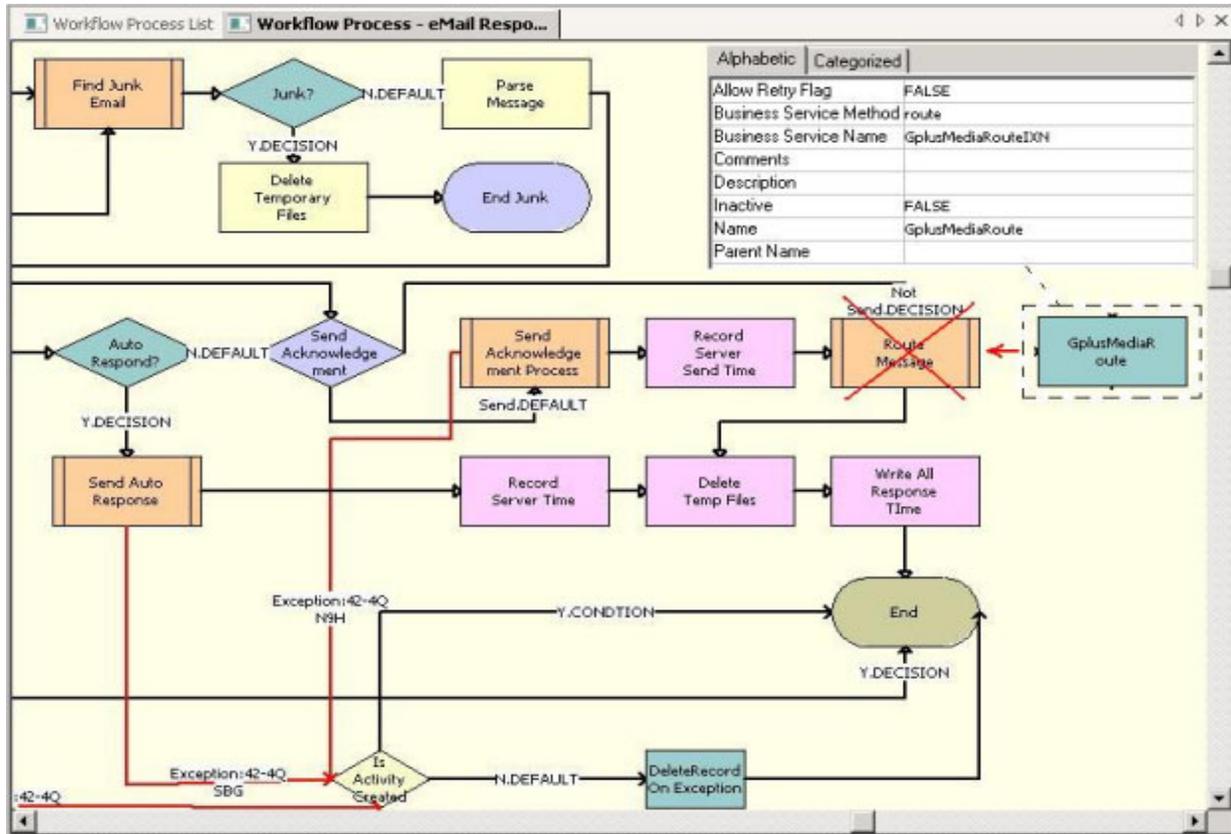
Figure 44: Editing the eMail Response - Process Message

> **9.** Add the following workflow process properties:

Table 66: Workflow Process Properties

Name	Display Name	In/Out	Data Type
RouteMessage	RouteMessage	In/Out	string
RouteResult	RouteResult	In/Out	string

> **10.** Select the GplusMediaRoute business service step and from the Multi Value Property window, select the Input Arguments tab.
>
> **11.** Add the following arguments:

Table 67: Input Arguments

Input Argument	Type	Value	Property Name
BackupGOMSConnectionSubsystem	Literal	GplusOpenMediaServer Backup	
BusComp	Literal	Action	

Table 67: Input Arguments (Continued)

Input Argument	Type	Value	Property Name
BusObject	Literal	Action	
ConnectionName	Literal	<Interaction Server Name>	
DebugLogFile	Literal		
FailedStatus	Literal	NotQueued	
FromAddress	Process Property		MsgSenderAddress
FromPersonal	Process Property		MsgSenderName
InteractionField	Literal	Call Id	
InteractionSubtype	Literal	InboundNew	
InteractionType	Literal	Inbound	
ReceivedAt	Process Property		MsgReceivedTime
PrimaryGOMSConnectionSubsystem	Literal	GplusOpenMediaServer Primary	
RoutingMediaType	Literal	SiebelEmail (or BackgroundEmail)	
StatusField	Literal	Status	
Subject	Process Property		MsgSubject
SubmitQueue	Literal	<Routing Queue Name>	
SuccessStatus	Literal	Queued	
ThirdPartyId	Process Property		Activity Id

You can change the values, if required; however, you must provide the field values in the angled brackets "<>".

12. Select the GplusMediaRoute business service step and from the Multi Value Property window, select the Output Arguments tab.

13. Add the following output arguments:

Table 68: Output Arguments

Property Name	Type	Output Argument
RouteMessage	Literal	RouteMessage
RouteResult	Literal	RouteResult

14. Save the workflow process.

15. In the History Toolbar, click Back to return to Siebel Tools.

16. In the Workflow Processes list, select <Gplus Email Process Message> and choose Validate.

17. In the Workflow Processes list, select <Gplus Email Process Message> and click Publish/Activate.

End of procedure

Procedure:
Media Routing Component: Updating the Siebel workflow process for sending email

Purpose: To update the Siebel workflow process for sending emails.

Start of procedure

1. Using the Object Explorer, select the Workflow Process tab.

2. In the Workflow Processes list, query for a process named eMail Response - Client Send Email.

3. In the Workflow Processes list, select eMail Response - Client Send Email and click Revise.

4. In the Workflow Processes list, right-click eMail Response - Client Send Email, and choose Edit Workflow Process. The Workflow Process Designer appears.

5. Add a new business service step block between the Update Reply Activity
and the Feedback Categories blocks (or any other location you prefer).
Configure the following Business Service step properties:

Table 69: Business Service Step Properties

Parameter	Value
Name	GplusMarkWorkItemDone
Business Service Name	GplusMediaRoute
Business Service Method	MarkWorkItemDone

Update the corresponding connections (see Figure 45).
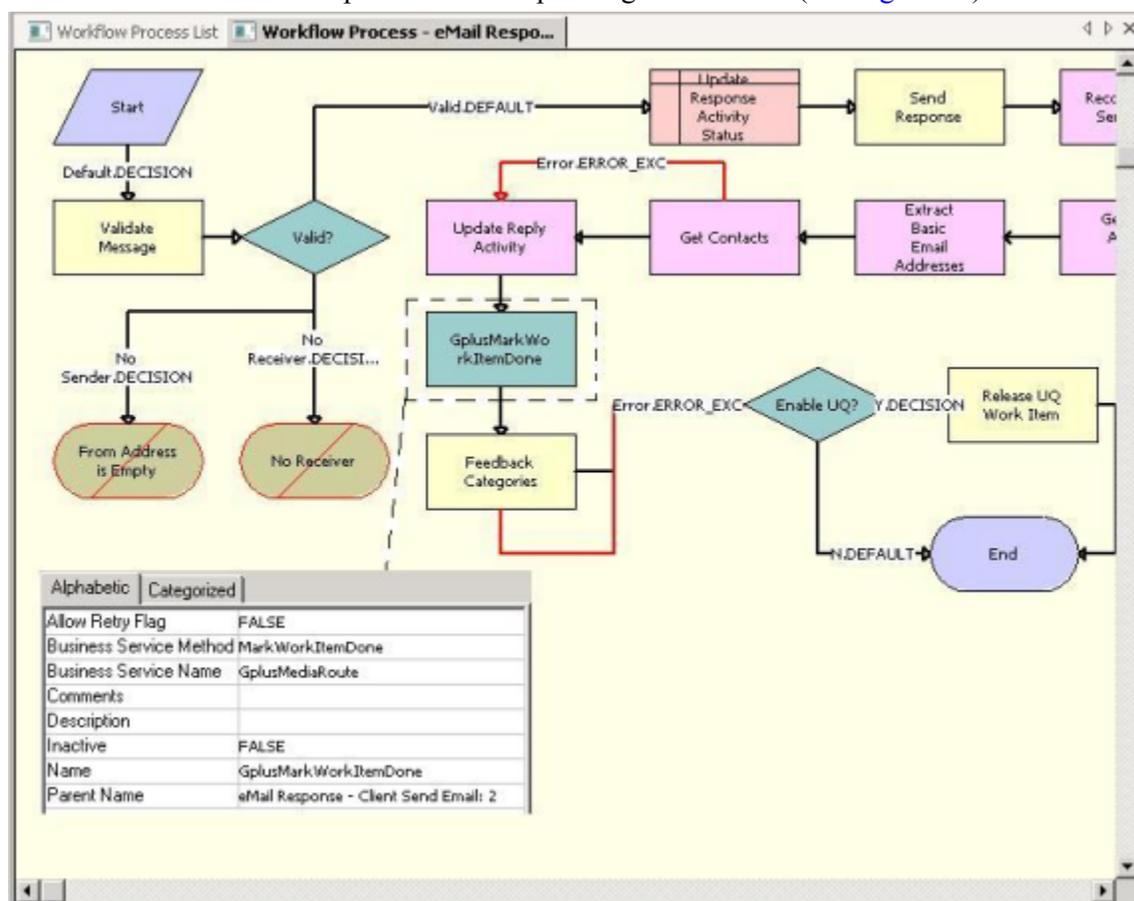
Figure 45: Workflow Process-eMail Response - Client Send Email

6. Select the GplusMarkWorkItemDone business service step and from the Multi
Value Property window, select the Input Arguments tab.

7. Add the following input arguments:

Table 70: Input Arguments

Input Argument	Type	Value	Property Name
BusObject	Literal	Action	
BusComp	Literal	Action	
ChildActivityId	Process Property		Activity Id
DebugLogFile	Literal		
FailedStatus	Literal		
InteractionField	Literal	Call Id	
RecIdField	Literal	Activity Id	
ParentRecIdField	Literal	Parent Activity Id	
StatusField	Literal	Status	
SuccessStatus	Literal	Done	

8. Save the workflow process.

9. In the History Toolbar, click Back to return to Siebel Tools.

10. In the Workflow Processes list, select eMail Response - Client Send Email and choose Validate.

11. In the Workflow Processes list, select eMail Response - Client Send Email and click Publish/Activate.

End of procedure

Next Steps

- Create a Siebel Enterprise Server Named Subsystem. See, "Creating a Siebel Enterprise Server Named Subsystem" on .

Creating a Siebel Enterprise Server Named Subsystem

This section describes how to configure a connection subsystem for the G*plus* Open Media Server that is used by the Media Routing Component. There are slight variations in the navigation that is required by this process, depending on the version of Siebel with which you are working.

Procedure:
Media Routing Component: Creating a connection subsystem

Purpose: To configure a connection subsystem for the G*plus* Open Media.

Start of procedure

1. Log in to Siebel Client as a Siebel Administrator.

2. Navigate through the Site Map to Siebel Server configuration:
    *   Select Site Map > Administration - Server Configuration.

Creating a Connection Subsystem

3. Click the Profile Configuration applet.

4. Click the New Record button.

5. Specify the following parameters for the new record (see Figure 46):
    *   Set the Profile field and Alias field to GplusOpenMediaServerPrimary.
    *   Set the Subsystem Type field to HTTPSubSys.
    *   Set the Description field to GplusOpenMediaServerPrimary.
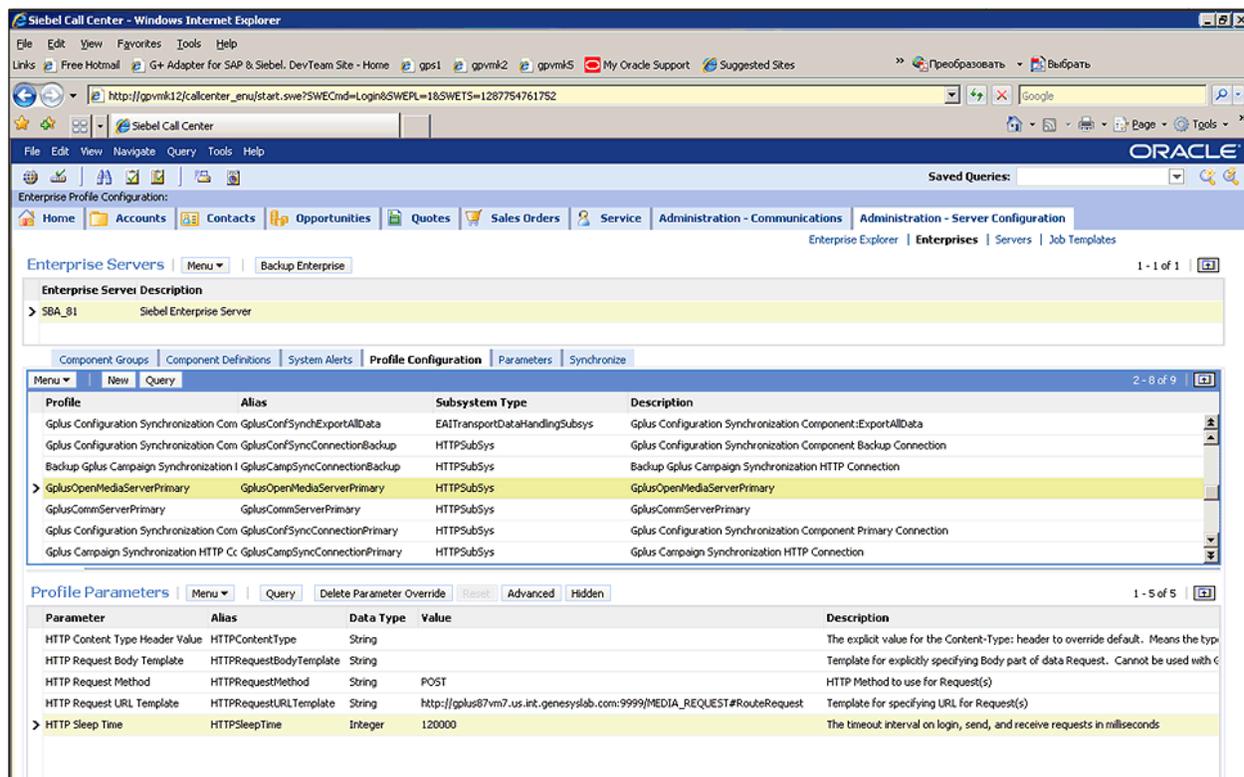
Figure 46: Creating a Connection Subsystem

6. Save the newly created record.

7. For the Named Subsystem record that you just created, enter the following parameters in the Profile Parameter applet:
   * Set the HTTP Request Method parameter to POST.
   * Set the HTTP Request URL Template to http://<host>:<port>/MEDIA_REQUEST#RouteRequest,

   where <host> and <port> are the host and the port of the G*plus* Open Media Server being used.

Creating a Backup Connection Subsystem

8. In the Profile Configuration applet, create a new record.

9. Specify the following parameters for the new record (see Figure 46 on page 489):
   * Set the Profile field and Alias field to GplusOpenMediaServerBackup.
   * Set the Subsystem Type field to HTTPSubSys.
   * Set the Description field to GplusOpenMediaServerBackup.

10. Save the new record.

11. For the backup Named Subsystem record that you just created, enter the following parameters in the Profile Parameter applet:
    * Set the HTTP Request Method parameter to POST.
    * Set the HTTP Request URL Template to http://<host>:<port>/MEDIA_REQUEST#RouteRequest,

    where <host> and <port> are the host and the port of the G*plus* Open Media Server being used.

Note: After creating or updating a connection subsystem, you might have to restart your Siebel server to make the changes effective. Refer to your Siebel server documentation.

End of procedure

Next Steps

* Configure the Multimedia Component agents and communications. See the section, "Configuring the Multimedia Component Agents and Communications".

Configuring the Multimedia Component Agents and Communications

This section describes how to configure the Multimedia Component agents and communications.

Procedure:
Media Routing Component: Setting a value for the MediaRoutingDefaultQueue configuration parameter

Purpose: To set a value for the MediaRoutingDefaultQueue configuration parameter. This parameter defines the name of the queue where the Siebel eMail interactions that are marked as done are placed. No other parameter configuration is required, except what is used for the Multimedia component.

Start of procedure

1. In the Genesys environment, create SiebelEmail and BackgroundEmail media types. If you use the Media Routing Component for routing Siebel work items, create a custom media type for each type of work item.

   Note: Genesys recommends using different media types for different Siebel work item types.

2. Add the newly-created media types to the agent capacity rules on the Advanced tab of the SiebelEmail Properties window (see Figure 47 on page 492).

3. Configure the Genesys routing strategy for routing the newly-created media type interactions—for example, see the <Siebel Version>\GplusMediaRoutingSample.zcf file in the installation directory as a sample strategy.

   Note: No additional agent configuration is required, except for the configuration that was already done for the Multimedia Component configuration. Make sure that the agent has the following capacity rules enabled:
   - If an agent is working with Siebel eMail then the SiebelEmail media type is enabled.
   - If an agent is working with background Siebel eMail, the BackgroundEmail media type is enabled.

   Any custom media types are enabled if the Media Routing Component is used for routing Siebel work items.
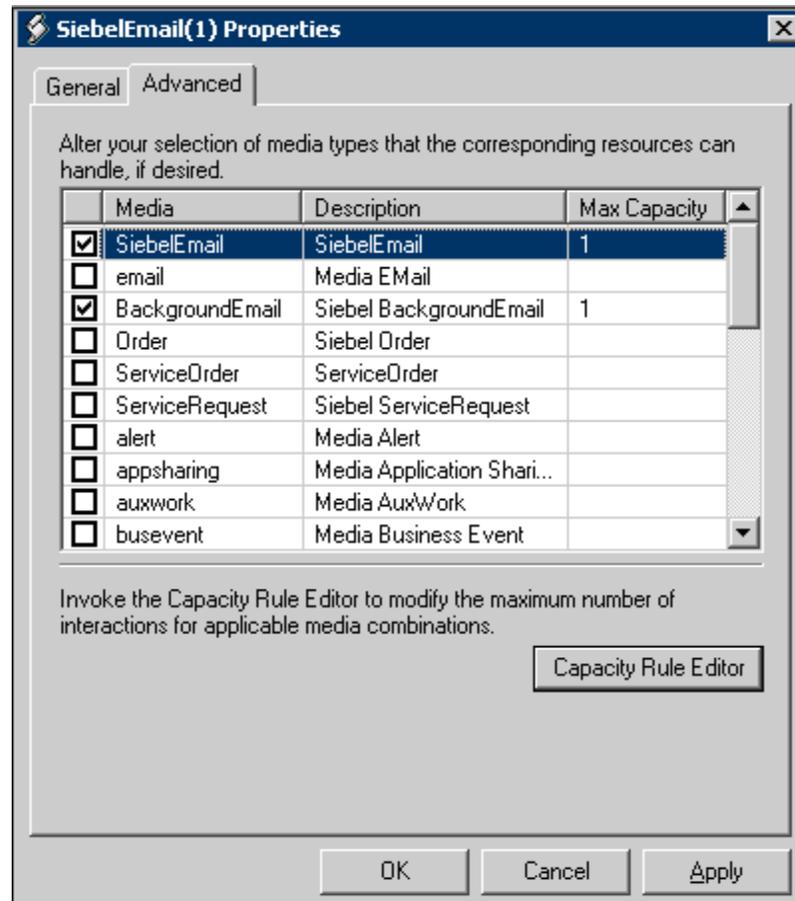
Figure 47: Capacity Rule Wizard

End of procedure

Next Steps

• Configure the Siebel eMail response. See the section, "Configuring the Siebel eMail Response".

Configuring the Siebel eMail Response

This section describes how to configure the Siebel eMail Response. A Siebel eMail Response is a Siebel application that communicates with the Siebel database in acknowledgement of a received email.

Configuring the Siebel eMail Response consists of the following procedures:

Note: This section is required only if you are using the Media Routing Component for routing Siebel eMail.

Media Routing Component: Activating the Siebel eMail Response Workflow Process

Purpose: To activate the Siebel eMail Response workflow process.

Start of procedure

1. Log into the Siebel Client as a Siebel Administrator.

2. Select Site Map > Administration > Business Process, Workflow Deployment.

3. In the Active Workflow Processes applet, choose the Import Processes menu option.

   The Workflow Process Import pop-up window displays.

4. In the Workflow Process Import window, select the following workflow processes:
   • eMail Response-Analyze Message
   • eMail Response-Append Thread Id
   • eMail Response-Client Send E-mail
   • eMail Response-Create Activity
   • eMail Response-Get Entitlement Id
   • eMail Response-Identify Language
   • eMail Response-Parse Junk E-mail
   • eMail Response-Response Workflow
   • eMail Response-Search Spec
   • eMail Response-Send Acknowledgement
   • eMail Response-Send Auto Response
   • eMail Response-SR Help
   • eMail Response-Update Activity Status

5. Make sure that the workflow is active. If not, select the workflow process you want to activate and click Activate.

   For more detailed information, see the *Siebel eMail Response Administration Guide*.

End of procedure

Next Steps

- Configure the Siebel Internet SMTP/POP3 Server driver profile. See, Procedure: Media Routing Component: Configuring the Siebel Internet SMTP/POP 3 Server driver profile.

Procedure:
Media Routing Component: Configuring the Siebel Internet SMTP/POP 3 Server driver profile

Purpose: To configure the Siebel Internet SMTP/POP3 Server driver profile.

Start of procedure

1. In the Siebel Site Map, select Site Map > Administration - Communications > Communications Driver and Profiles**.**

2. Select the record for the Internet SMTP/POP3 Server communication driver.

3. Select the Profile tab.

4. Click New on the Profiles form and set the name of the profile to the value of the e-mail address.

5. Click New to create the following necessary parameter value overrides:
   - From Address: the value must be the same as the e-mail address that the profile works with, or the profile name, which is the same.
   - Incoming E-mail Directory: the directory where the incoming e-mails are saved.
   - POP3 Account Name: the name of the POP account.
   - POP3 Account Password: the POP3 account password string.
   - POP3 Server: the name of the POP Server.
   - POP3 Server Port: the POP3 Server Port value (this value is usually 110).
   - Siebel Server: the name of the Siebel Server on which the eMail Response works.
   - SMTP Server: the name of the SMTP Server.
   - SMTP Server Port: the SMTP Server Port value (this value is usually 25).
   - SMTP Account Name: the name of the SMTP account.
   - SMTP Account Password: the SMTP account password string.

6. Set the responsibilities of the created Server Driver Profile to Universal Agent.

End of procedure

Next Steps

- Create a response group. See, Procedure: Media Routing Component: Creating a response group.

Procedure:
Media Routing Component: Creating a response group

Purpose: To create a response group that determines the behavior for inbound email.

Start of procedure

1. In the Siebel Site Map, select Site Map > Administration - Communications > All Response Groups.

   The Response Groups list appears.

2. Click New on the Response Groups form.

3. Set the name of the response group to the exact value of the Internet SMTP/POP3 driver profile name, with which the response group works.

4. Set the administrator e-mail address to an e-mail address in which you want to direct undeliverable e-mail that is sent to profiles in each response group.

5. Set the Server option to the value of the Siebel Server that is used for the eMail Response processing.

6. Set the value of the Startup option to Active.

7. Select the Profiles tab.

8. Click New and select the previously created profile for the Internet SMTP/POP3 driver from the list of profiles.

9. Select the Input Arguments tab and click Generate Defaults.

10. Replace the ProcessName argument value with the actual Siebel email processing workflow name.

    For Siebel Servers prior to 8.1.1.11/8.2.2.4, it must be GplusMediaRouting-ProcessMessage.

    For Siebel Servers 8.1.1.11/8.2.2.4 or later, use the name of the workflow referenced as <Gplus Email Process Message>. (See Figure 48 on page 496.)
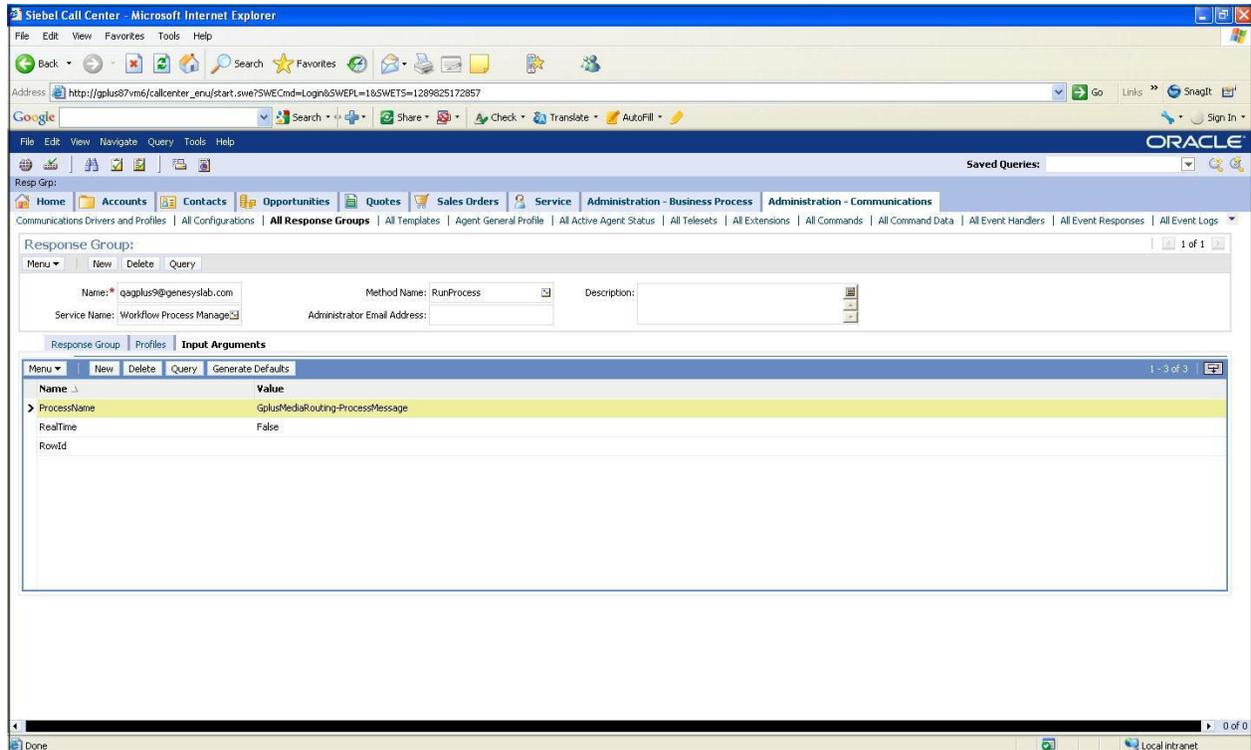
Figure 48: Input Arguments Editing

> **End of procedure**
>
> **Next Steps**
>
> • There are no further steps.

Nonreal-Time (Background) E-Mail Routing

Nonreal-time or background e-mail processing allows you to apply more server processing power to the e-mail routing, which is especially important when your site receives high volumes of e-mail. The nonreal-time (background) mode of processing e-mail allows the servers to process a high volume of inbound e-mails in the following way:

• Inbound mail messages are retrieved by the Siebel eMail handler.

• The Internet SMTP/POP3 driver converts the inbound e-mail into multi-purpose internet mail extension (MIME) format files.

• The Communications Inbound Receiver (CIR) transfers these files into the Siebel File System. The Siebel File System is a shared file system directory.

- The Communications Inbound Processor (CIP) then processes these files. Since the CIP uses the workflow to process the e-mail, you can set up multiple CIPs to process the volume of the e-mail flow.

The native Siebel eMail Response application uses the Siebel Universal Queue (UQ) to route the e-mails and to generate a @HandleNonRealtimeWorkItem Siebel special event. In the Genesys-adapted Siebel eMail Response application, the Genesys Universal Router (UR) functionality is used for e-mail routing, therefore the @HandleNonRealtimeWorkItem Siebel event is not generated.

The Media Routing Component uses workbins for background e-mail interactions. Genesys recommends using the same agent workbin for the Siebel background e-mail as for the Multimedia Component inbound e-mail. The mode of interaction processing is defined by a Genesys routing strategy—for example, the background interaction should be placed into the agent workbin, but the real-time interactions should be placed into the inbound queue. By default, the Media Routing Component uses different media types to instruct the Genesys Universal Router on how to process the e-mail call on the agent side—for example, refer to the provided strategy sample in the installation directory <Siebel Version>\GplusMediaRoutingSample.zcf file.

The SiebelEmail media type is used for the real-time e-mail processing mode, and the BackgroundEmail media type is used for nonreal-time e-mail processing mode. The media type is set as an input argument of the GplusMediaRoute business service in the email processing business process. You may add an attribute's value in an interaction's attached data as an indicator of the processing mode to be applied to an interaction, and update a strategy to check this attribute's value.

The nonreal-time (background) mode may be used for routing any Siebel work items. The mode of processing is defined by the strategy used, which is the same as for Siebel eMail.

Pulling/Stopping Siebel eMail Interactions

This section describes the pull/stop email functionality where agent requests for the data are responded to by the server. Customization is required to use the pull/stop email functionality provided in the Media Routing Component as the component does not provide a user interface for this functionality.

Enabling the Pull Functionality
The pull functionality is implemented by the following communication commands: PullbyInteractionIdMR and PullbyThirdPartyIdMR. To enable the pull functionality for an agent, you must allow an agent to invoke the PullbyInteractionIdMR or PullbyThirdPartyIdMR communication commands.

The command PullbyInteractionIdMR uses the InteractionId data parameter, which is the field name to store the Genesys interaction ID, and invokes a OpenMediaPullInteractionById multimedia driver device command.

The command PullbyThirdPartyIdMR uses the ThirdPartyId data parameter, which is the field name to store the Siebel record ID, and invokes the GplusMediaRoute.PullInteraction business service method.

Enabling the Stop Functionality

The stop functionality is implemented by the following communication commands: StopbyInteractionIdMR or StopbyThirdPartyIdMR.To enable the stop functionality for an agent, you must allow an agent to invoke the StopbyInteractionIdMR or StopbyThirdPartyIdMR communication commands.

The command StopbyInteractionIdMR uses the InteractionId data parameter, which is the field name to store the Genesys interaction ID, and invokes the GplusMediaRouteIXN.StopWorkItem business service method.

The command StopbyThirdPartyIdMR uses the ThirdPartyId data parameter, which is the field name to store the Siebel record ID, and invokes the GplusMediaRouteIXN.StopWorkItem business service method.

To invoke the commands, you may do use one of the following methods:

- Add a button into one of the activity applets to invoke the command.
- Add a toolbar button.
- Set the Hidden parameter of the command to True. The agent should locate the appropriate Siebel record and invoke a command.

For additional information, refer to the "Media Routing Component Customization" section of the G*plus Adapter 8.0 for Siebel CRM Developer's Guide*.

Configuring the Media Routing Component for Routing Siebel Work Items

This section describes how to configure the Media Routing Component for routing Siebel work items in both Genesys and Siebel environments. The Media Routing Component for Siebel can be used for routing any type of Siebel work item by using Genesys Multimedia, both in real-time and in nonreal-time (background) mode. The Media Routing Component provides the basic functionality for work item routing, but does not provide an out-of-the-box graphical user interface (GUI), so customization is required to create this functionality.

Procedure:
Media Routing Component: Configuring the Media Routing Component in a Genesys environment

Purpose: To configure the Media Routing Component in a Genesys environment.

Start of procedure

1. Create a new custom media type, making sure to use different media types for different Siebel work item types—for example, create the ServiceRequest and ServiceOrder media types to use the provided samples.

2. Add the newly-created media types to the agent capacity rules.

3. Configure the Genesys routing strategy for routing the newly-created media type interactions— for example, see the <Siebel Version>\GplusMediaRoutingSample.zcf file in the installation directory as a sample strategy.

End of procedure

Next Steps

- Configure the Media Routing Component in a Siebel environment. See, .

Procedure:
Media Routing Component: Configuring the Media Routing Component in a Siebel environment

Purpose: To configure the Media Routing Component in a Siebel environment.

Start of procedure

1. Add the newly-created media types into the Driver.Channel string.

2. Create Ready and NotReady commands for the newly-created media types. A sample of this command is provided in the <Siebel Version>/GenComm_universal.def file located in the installation directory. The ReadyForSiebelSRGroup and NotReadyForSiebelSR commands are located at the end of the file.
    - Add the new commands into the proper group commands. To use the provided samples, add the ReadyForSiebelSRGroup command into the ReadyGroup command group, and the NotReadyForSiebelSR command into the NotReadyGroup command group.

3. Create an event handler for the OpenMediaAccepted event for each of the newly-created media types—for example, refer to the sample event handler, OpenMediaAcceptedSR.
    - Set the proper Filter and Event Response parameters for the event handler.

4. Create an event handler for the OpenMediaSelected event for each of the custom media types —for example, refer to the sample event handlers, OpenMediaSelectedSR and OpenMediaSelectedOrder.

    a. Set the proper FilterSpec and Event Response parameters for the event handler.

    b. Set a Siebel view that is used for editing a routed work item as a SingleView parameter in the event response.

    c. Set the QuerySpec parameter to the FieldName='{ThirdPartyId}' value, where FieldName is the field ID for a Siebel work item—for example, refer to the sample OpenMediaSelectedSR and OpenMediaSelectedSO event responses located at the end of the <Siebel Version>/GenComm_universal.def file.

5. Create event handlers for the OnOpenMediaPulledSR and OpenMediaAccepted events for each of the custom media types.

    •   Set the proper Filter, MediaType, and Event Response parameters for the event handler—for example, refer to the sample OnOpenMediaPulledSR and OpenMediaAcceptedSR event handlers.

6. Create the commands to send a routing request.

    •   Set the ServiceParam.ThirdPartyId command parameter to the ID field for a Siebel work item—the ID field should be the same field as in a proper event handler. You may also send a routing request from a workflow by calling the routing method of a GplusMediaRouteIXN business service—for example, see the SendRouteSR sample command to route Siebel service requests in real-time mode, and see the sample SendRouteSO command to route service orders in nonreal-time (background) mode. Both of these sample commands are located at the end of the <Siebel Version>/GenComm_universal.def file.

    For more information, refer to the "Media Routing Component Customization" section of the *Gplus Adapter 8.0 for Siebel CRM Developer's Guide*.

7. Configure the MarkWorkItemDone command to support your media types—for example, adding the newly-created media types into the FilterMediaType parameter. This parameter contains a list of media types separated by commas—for example,

    •   For a ServiceOrder media type, you may use a MarkDoneSO command.

    •   For a ServiceRequest media type, you may use a MarkDoneSR command, or you may create your own MarkDone command.

    •   As an alternative method, you may customize a button on a work item view, or add a customized button on the communication toolbar to invoke a MarkDone command.

    For more information, refer to the *Gplus Adapter 8.0 for Siebel CRM Developer's Guide*.

8. Assign the correct responsibilities to the agents, to make sure that they have enough responsibilities to access to the appropriate views. For the provided samples, the agent should have access to the following views:

- Service Request Detail Personal Service Request List
- Service Order-Browse Catalog
- Order Entry - Line Items
- Order Entry - My Orders.

Notes: The provided samples show how to route Siebel service requests in real-time mode and how to route service orders in nonreal-time (background) mode. The samples use the ServiceRequest media type for Siebel service requests and the ServiceOrder media type for service orders.

- To use the samples, remove the comment marks in the <Siebel Version>/GenComm_universal.def file before importing a configuration, and follow the steps in Procedure: Media Routing Component: Configuring the Media Routing Component in a Genesys environment, on page 498.
- To use the service requests commands, open the All Service Request List view and then invoke a command from this view.
- To use the service orders commands, open the Order Entry - All Orders view and then invoke a command from this view.

9. If you are going to use the same pull/stop functionality for Siebel work items as for the Siebel eMail routing, you should provide a record field to store the Genesys interaction ID, as it is required to support a relation between the Siebel work item records and the Genesys interactions.

If a work item is represented in the Siebel database as an activity record, you may use a Call Id field, as done in Siebel eMail. For more information, refer to the "Media Routing Component Customization" section of the *G*plus *Adapter 8.0 for Siebel CRM Developer's Guide*.

End of procedure

Next Steps

- There are no further steps.

Driver and Configuration Parameters

For a complete list of the driver parameters that the Media Routing Component uses, refer to the driver parameters described in Chapter 8, "Deploying the Multimedia Component".

Table 71 below contains the parameters *specific* to the Media Routing Component.

Table 71: Media Routing Driver Parameters

Parameter Name	Default Value	Required	Comment
MediaRoutingDoneQueue	__STOP__	Yes	The multi-channel and routing (MCR) queue where the third-party interactions (for example, Siebel eMails) are placed after completion (the Send and Cancel buttons for Siebel eMail). __STOP__ means that the processing of the interaction is completely stopped. If post-processing for the completed third-party items is needed, this value should be changed to the post-processing queue where the post-processing strategy is running.
MediaRoutingDefaultQueue	(E-mails for agent processing)	Yes	The multi-channel and routing (MCR) queue where the third-party interactions (for example, Siebel eMails) are placed for processing

Device Commands and Events

The Media Routing Component uses the same driver as the Multimedia Component, so the device commands and events are the same as for the Multimedia Component. For a list of the device commands and events, refer to Chapter 8, "Deploying the Multimedia Component," on page 391.

![Genesys logo]

10 Deploying the iWD Routing Component

This chapter describes how to configure and install the G*plus* iWD Routing for Siebel CRM Component.

This chapter includes the following sections:

Overview

The process of configuring and installing the G*plus* iWD Routing for Siebel CRM Component includes the following general procedures:

- Configuring Genesys
- Installing the iWD Routing for Siebel CRM Component
- Configuring Siebel
- Configuring the Multimedia Component Agents
- Configuring the Siebel eMail Response

New in This Release

This section provides information about new features or functionality in the G*plus* iWD Routing for Siebel CRM Component.

Release 8.0.210   No new features were introduced in this release.

Release 8.0.2   The following new features or functionality are included in this release:

- Support for Siebel versions 8.1.1.14/8.2.2.14 (IP2014) for both HI and Open UI modes.
- Support for Red Hat Enterprise Linux 6.
- Support for Red Hat Enterprise Linux 7.
- Support for AIX 7.1.
- Support for Solaris 11.

Release 8.0.110   The following new features or functionality are included in this release:

- Support for Siebel versions 8.1.1.11/8.2.2.4 (IP2013) for both HI and Open UI modes.

Release 8.0.1   No new features were introduced in this release.

Release 8.0.0   This is the first release of this component.

Configuring Genesys

This section describes how to configure the Genesys section of the G*plus* iWD Routing for Siebel CRM Component.

It is assumed that the iWD solution is properly configured to accept interactions from the Siebel side. If required, the iWD Departments and Processes have to be configured to apply the special rules—for example, different media types. See the *intelligent Workload Distribution 8.0. Deployment Guide* for details.

Procedure:
iWD Routing: Configuring the intelligent Workload Distribution solution

Purpose: To configure the intelligent Workload Distribution (iWD) solution.

Start of procedure

1. For each media type that you need to route through iWD, create a Webservice Capture Point by doing the following:

   a. Run the iWD Manager.

   b. Log in and select an appropriate Tenant.

   c. Select the Services view.

   d. Select an appropriate iWD Solution and open the Services section (see Figure 49 on).
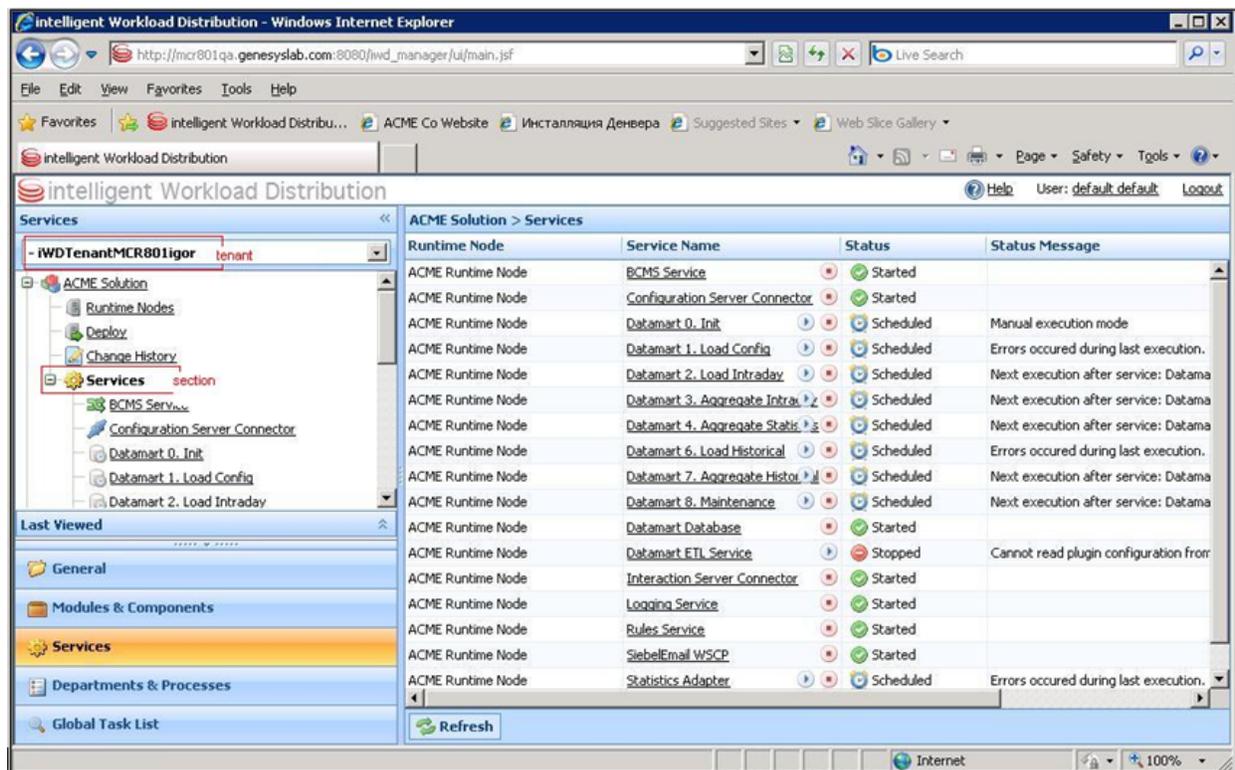
Figure 49: iWD Solution Services View

   e. Select the New Service.

   f. In the Service Creation form:

   i. Fill the Service Name field with a unique name.

   ii. Select Webservice Capture Point as the service template.

   iii. Select Runtime Node from the pull-down list.

   iv. Deselect the Default check-box of the mediaType property and select an appropriate media-type value from the pull-down list.

   v. Deselect the Default check box of the webserviceURLMapping property and enter a unique value—for example, refer to this value as <mediaURLMapping>.

   vi. Update the other properties, if required.

vii. Click Save to save the newly created service (see Figure 50 on page 506).
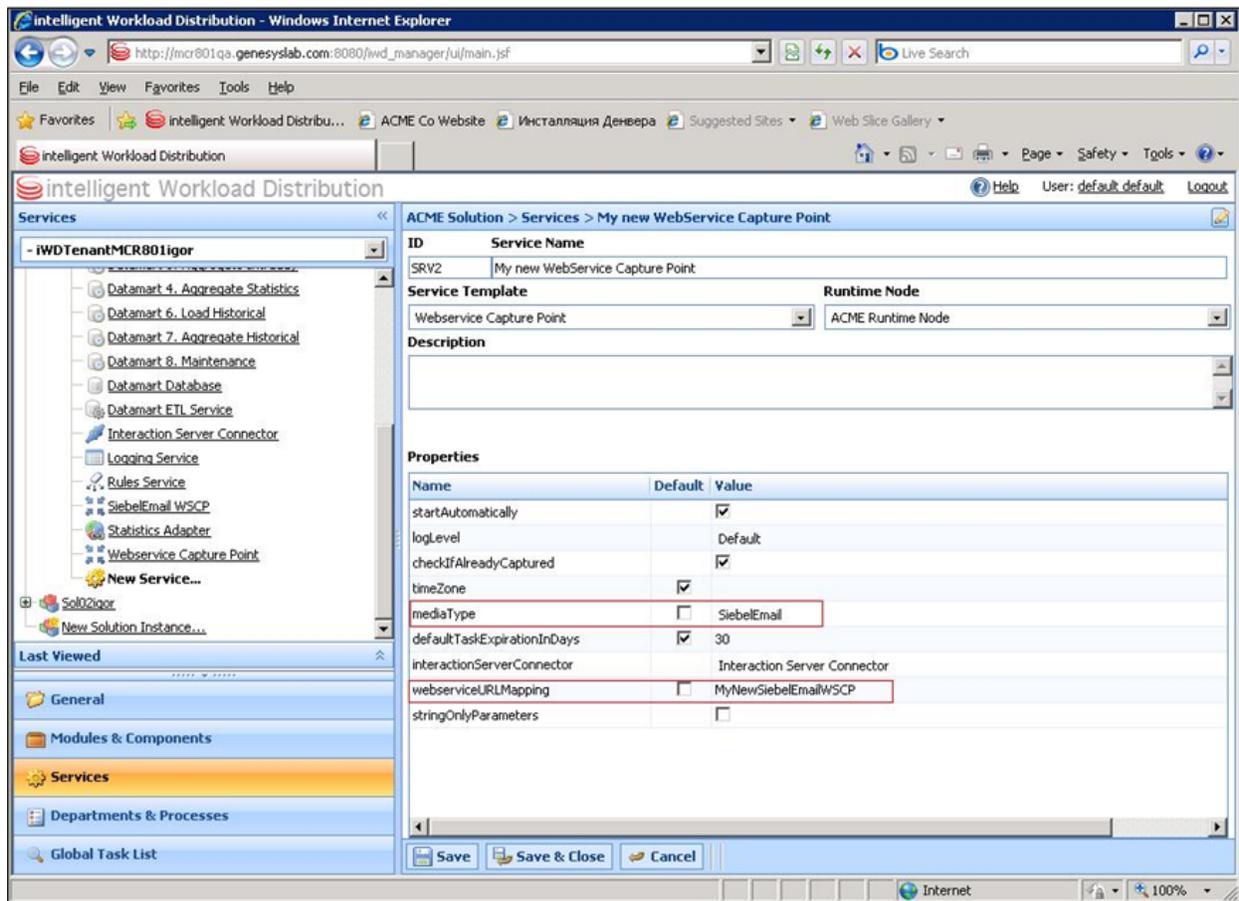
Figure 50: New WebService

g. Select the Deploy section.

h. Under the Deploy view, select the newly created services.

i. Click Deploy.

2. Store the Web Services Description Language (WDSL) representation of the service for each newly created Webservice Capture Point by doing the following:

a. Type the following URL in an internet browser:

<runtime node context URL>/services/<mediaURLMapping>?WSDL.

b. When the page has loaded, save the contents as an .XML file (menu File -> Save as)—for example, refer to this file as <mediaWSCP.WSDL>.

End of procedure

Next Steps

- Install the iWD Routing for Siebel CRM Component. See the section, "Installing the iWD Routing for Siebel CRM Component" on .

Installing the iWD Routing for Siebel CRM Component

This section describes the installation process for the G*plus* iWD Routing for Siebel CRM Component. The Component can be installed on Windows or UNIX platforms and is divided into the following sub-sections:

-
-
-

Notes:
- The G*plus* iWD Routing for Siebel CRM Component uses some of the same functionality as the G*plus* Adapter for Siebel CRM Multimedia Component, so the G*plus* iWD Routing for Siebel CRM Component should be installed after the Multimedia Component. The G*plus* iWD Routing for Siebel CRM Component will not function without the G*plus* Adapter for Siebel CRM Multimedia Component.
- The G*plus* iWD Routing for Siebel CRM Component is an alternative to the G*plus* Adapter for Siebel CRM Media Routing Component. It is not recommended to use both of these components simultaneously.

Installation Prerequisites

The following are prerequisites for installation of the G*plus* iWD Routing for Siebel CRM Component:

- Back up the Siebel database and the Siebel repository (*.srf) file.
- Install the G*plus* Adapter for Siebel CRM Multimedia Component (see Chapter 8, "Deploying the Multimedia Component," on page 391).

Installation

This section describes how to install the G*plus* iWD Routing for Siebel CRM Component on either a Windows or a Linux operating systems.

Note: Do not use any special symbols in the destination directory name when installing the G*plus* iWD Routing for Siebel CRM Component in a UNIX environment.

Procedure:
iWD Routing: installing the iWD Routing for Siebel CRM Component

Purpose: To install the G*plus* iWD Routing for Siebel CRM Component on either a Windows or a Linux operating system.

Start of procedure

1. To start the installation process, run the setup.exe file (for Windows) or run the install.sh file (for UNIX) from the G*plus* Adapter for Siebel CRM Multimedia Component installation package on the host where the G*plus* Communication Server is installed.

2. Enter the installation destination directory.
   The installation program will place the Siebel archive, scripts, and general production information in this folder.

End of procedure

Next Steps

- Configure Siebel. See the section, "Configuring Siebel" on .

Target Directory Structure and File Locations

The following generic directory and host names are used in the description of the directories structure:

- <Destination Directory> is the destination directory for installation used by the installation script to copy the Adapter for Siebel CRM iWD Routing files.

- Files in subdirectories <Destination Directory>/7.7 are related to Siebel versions 7.7 /7.8.

- Files in subdirectories <Destination Directory>/8.0 are related to Siebel version 8.0.

- Files in subdirectories <Destination Directory>/8.1 are related to Siebel version 8.1.

- Files in subdirectories <Destination Directory>/<Application Name>/8.1_8.2_OUI are related to Siebel version 8.1.1.11/8.2.2.4 (IP2013).

- Files in subdirectory <Destination Directory>/<Application Name>/IP2014 are related to Siebel 8.1.1.14/8.2.2.14 (IP2014).

Configuring Siebel

This section describes how to configure the Siebel part of the G*plus* iWD Routing for Siebel CRM and is divided into the following sub-sections:

Prestart Information

Before starting this part of the configuration process, you must make sure to do the following:

- You must have the Siebel Tools application running.

- You should be connected to the local copy of the Siebel Server database.

- You should have sufficient privileges to check-in and check-out projects from the Siebel Server repository.

- You should ensure that all required steps from "Patching and Configuring Siebel CRM" are performed.

Configuring Siebel Using the Siebel Tools

Use the Siebel Tools to compile an updated version of the Siebel repository file (SRF or *.srf file) for one, or more, of the Siebel applications that you use on your Siebel Server, which you will then deploy in the server. For more information about using Siebel Tools, see the Siebel documentation.

Configuring Siebel using the Siebel Tools consists of the following sub-sections:

Siebel Repository File Preparation

This section explains how to modify the Siebel repository (*.srf) file.

Procedure:
iWD Routing: Compiling and Deploying the Siebel Repository File

Purpose: To import the necessary Siebel repository (*.srf) files and prepare the Siebel repository file for compiling.

Start of procedure

1. Start Siebel Tools against the local database.

2. Lock the GplusMediaRouting project.

3. Import the GplusMediaRoutingiWD.sif file.

4. Create a new Web Service for each Webservice Capture Point using the WSDL files.

   a. Select menu File -> New object.

   b. Select the EAI tab and the Web Service form.

   c. Select GplusMediaRouting from the Select the Project to hold objects created from WSDL file drop-down list.

   d. Specify an appropriate WSDL file (<mediaWSCP.WSDL>) generated by the iWD Webservice Capture Point.

   e. Specify the file name for the run-time XML data (Siebel Web Service representation). The log name is automatically created. If necessary, you can change the file names. For example, refer to the run-time XML data file as <mediaWebService.XML>. See Figure 51 on .
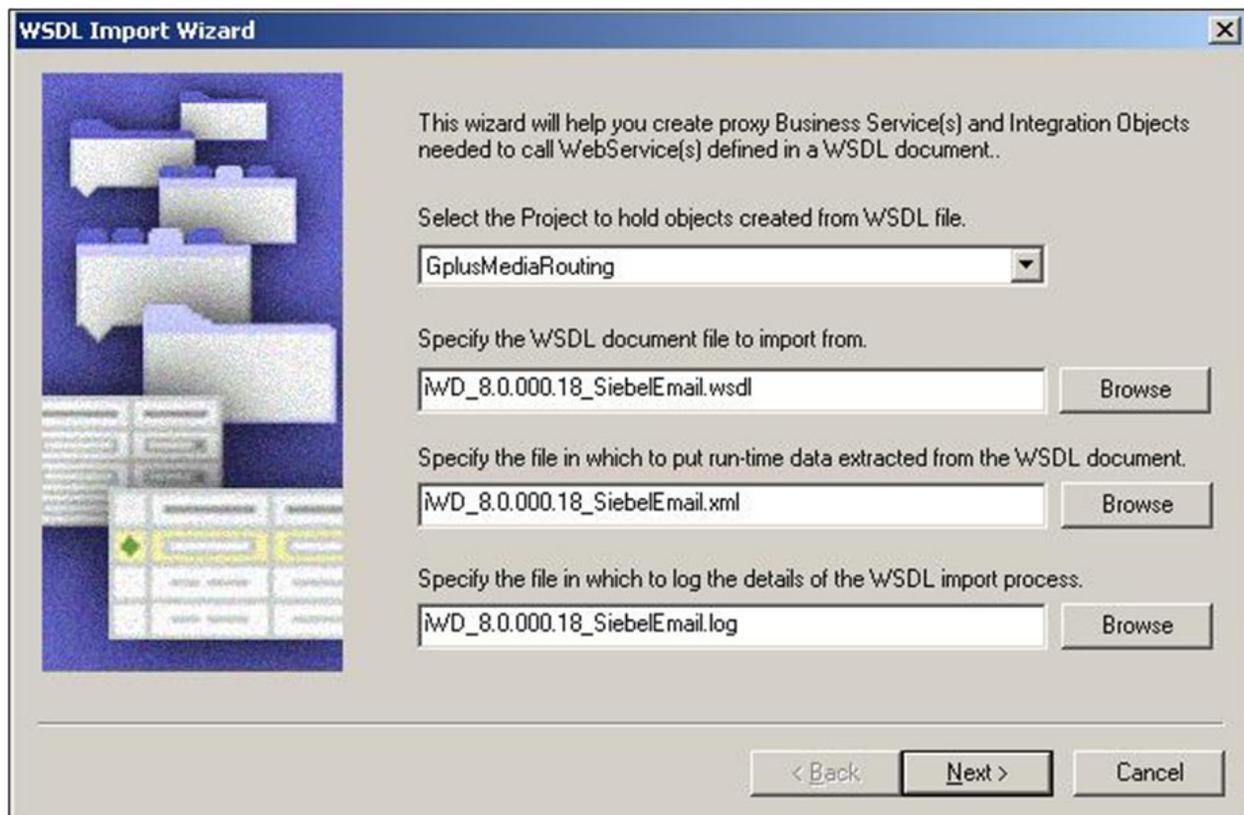
Figure 51: WSDL Import Wizard

5. Click Next.

6. Click Finish.

During Web Service creation, Siebel creates a number of Integration Objects and only one business service. By default, this business service is named <mediaURLMapping>PortType—for example, refer to this business service name as <mediaBusService>.

End of procedure

Next Steps

• Configure a Siebel workflow. See the section, "Configuring Siebel Using the Siebel Web Client" on page 511.

Configuring Siebel Using the Siebel Web Client

This section describes how to configure the Siebel application using the Siebel Web Client and consists of the following sub-section:

Siebel prior to 8.1.1.11/8.2.2.4: Configuring the Siebel Workflow Process

This section describes how to configure a Siebel workflow process. A Siebel workflow is a series of Siebel business services that are linked to complete a task.

Notes: • This procedure is required only if you are using the iWD Routing Component for the routing of Siebel eMail.

• This procedure only applies to Siebel Server versions prior to 8.1.1.11

Procedure:
iWD Routing: Configuring a Siebel workflow process

Purpose: To configure a Siebel workflow process for a series of Siebel business services.

Start of procedure

1. Lock the EMR Workflow (or the GplusMediaRouting) project.

2. Using the Object Explorer, choose the Workflow Process tab.

3. Import the GplusMediaRoutingiWD-ProcessMessage.xml file.

4. Define the string values for the following properties according to your environment (for mandatory properties values in the GplusMediaRoutingiWD-ProcessMessage.xml file, preset the values to the CHANGE_ME string):
    •   PrimaryOutboundDispatcherBS (mandatory)
    •   BackupOutboundDispatcherBS (optional)
    •   BusinessValue (mandatory)
    •   Priority (mandatory)

    Set the <mediaBusService> as the value for the PrimaryOutboundDispatcherBS property (and optionally for the BackupOutboundDispatcherBS property). This property defines the Business Service name to be used to access the Web Service to send the queries to iWD.

- The BusinessValue property defines an interaction (task) value in iWD terms. See the *intelligent Workload Distribution 8.0. Deployment Guide* for details.

- The Priority property defines the interaction (task) initial priority in iWD terms. See the *intelligent Workload Distribution 8.0. Deployment Guide* for details.

  If necessary, add and set any of the iWD createTask method parameters into the workflow process. See table Table 78 on page 523 for parameters description.

5. To add AttachedUserData values to a routing request, you must add custom input arguments to the GplusMediaRouting workflow step. To add these values, perform the following procedure:

   a. Select the GplusMediaRouting box in the Design view.

   b. Right-click on GplusMediaRouting.

   c. From the menu, select Show Input Arguments. (See Figure 49 on page 505).

   d. Add a new record in the Input Arguments view.

      All input arguments except the predefined arguments are attached to a routing request as AttachedUserData.

6. Deploy the workflow process.

7. Export the workflow process into a file. For example, you can refer to this process as <SEWorkflow.XML>.

8. Compile and apply the *.srf file.

End of procedure

Next Steps

- Configure the Siebel application. See, Procedure: Siebel prior to 8.1.1.11/8.2.2.4: iWD Routing: Activating the Siebel Workflow Process, on page 514.

Procedure:
Siebel prior to 8.1.1.11/8.2.2.4: iWD Routing: Activating the Siebel Workflow Process

Note: This procedure is required only if you are using the iWD Routing Component for the routing of Siebel eMail.

Note: This procedure only applies to Siebel Server versions prior to 8.1.1.11.

Purpose: To activate the Siebel workflow process.

Start of procedure

1. Log into the Siebel Client as a Siebel Administrator.

2. Import the <SEWorkflow.XML> file exported previously from the Siebel Tools into the Active Workflow Processes list/set.

3. Make sure that the workflow process is active. If not, select the workflow process you want to activate and click Activate.

End of procedure

Next Steps

• Configure the Siebel Web Services. See, Procedure: iWD Routing: Configuring the Siebel Web Services.

Siebel 8.1.1.11/8.2.2.4 or later: Configuring and Deploying Siebel Workflow Processes

This section describes how to configure and deploy two Siebel workflow processes required to apply G*plus* iWD Routing to Siebel email processing.

Siebel 8.1.1.11/8.2.2.4 or later: Configuring and Deploying Siebel Workflow Processes

This section describes how to configure and deploy two Siebel workflow processes required to apply G*plus* iWD Routing to Siebel email processing.

Note: This procedure is required only if you are using the G*plus* Adapter for Siebel CRM iWD Routing Component for the routing of Siebel eMail.

Note: This procedure only applies to Siebel Servers version 8.1.1.11/8.2.2.4
or later.

Procedure:
iWD Routing: Updating or cloning the Siebel email processing workflow process

Purpose: To configure and deploy Siebel workflow processes.

Start of procedure

1. Open the Siebel Tool against the Siebel Server DB and log in as a Siebel Administrator.

2. Lock the GplusMediaRouting project.

3. Using the Object Explorer, choose the Workflow Process tab.

4. In the Workflow Processes list, query for a process named eMail Response - Process Message.

5. You can modify the selected Siebel workflow process directly or create and modify a copy. For the purpose of this procedure, describe the modified workflow as <Gplus Email Process Message>.

6. If you want to modify the vanilla Siebel workflow process, select eMail Response - Process Message from the Workflow Processes list and click Revise.

7. In the Workflow Processes list, right-click <Gplus Email Process Message> and choose Edit Workflow Process. The Workflow Process Designer appears.

8. Replace the Route Message sub-process block with a new business service step block, with the following properties:

Table 72: Business Service Step Properties

Parameter	Value
Name	GplusMediaRoute
Business Service Name	GplusMediaRouteiWD
Business Service Method	createTask

Configure the block connections with the same settings as the previous block (see Figure 52).
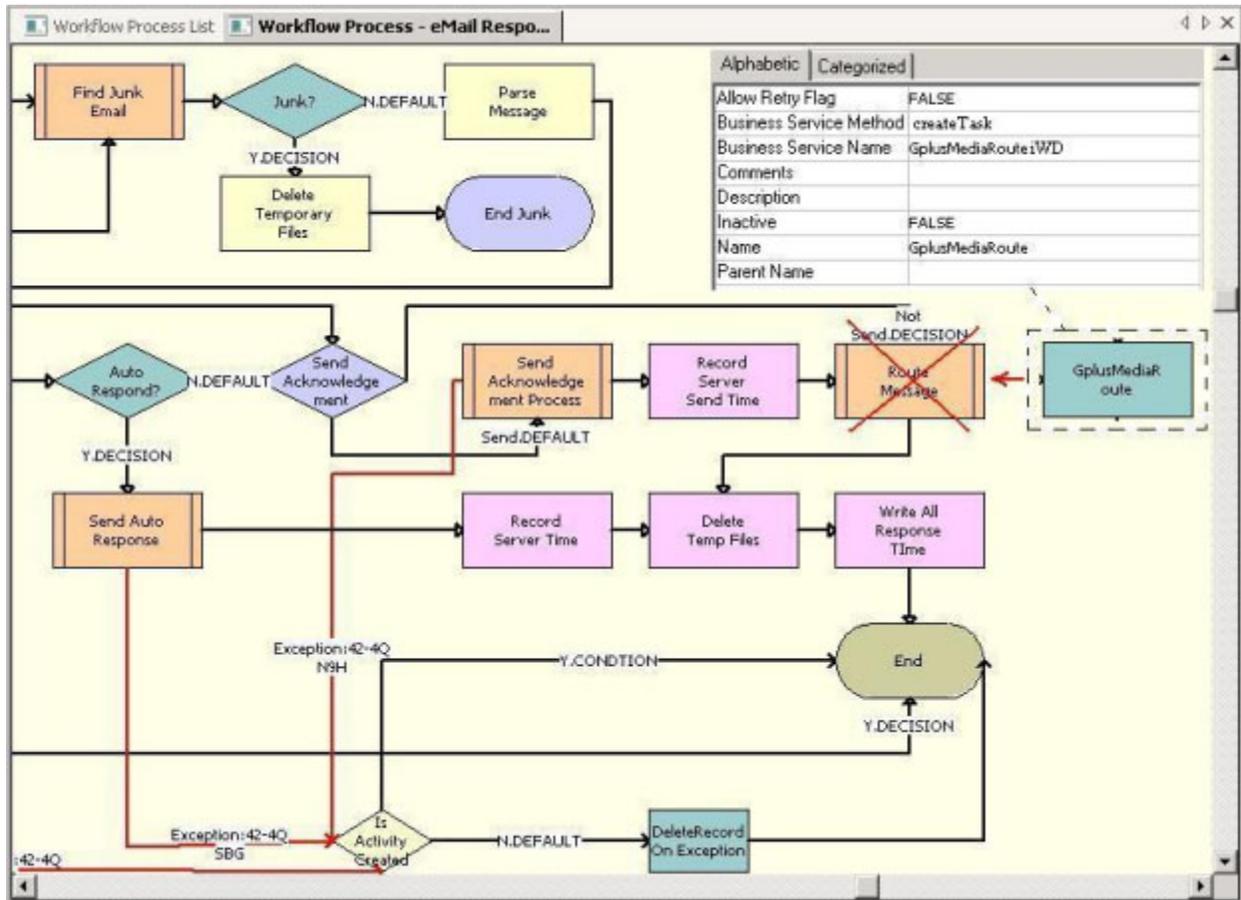
Figure 52: Workflow Process - eMail Response - Process Message

> **9.** Add the following workflow process properties:

Table 73: Workflow Process Properties

Name	Display Name	In/Out	Data Type
RouteMessage	RouteMessage	In/Out	string
RouteResult	RouteResult	In/Out	string

> **10.** Select the GplusMediaRoute business service step and from the Multi Value Property window, select the Input Arguments tab.

11. Add the following arguments:

Table 74: Input Arguments

Input Argument	Type	Value	Property Name
BackupOutboundDispatcherBS	Literal		
BusComp	Literal	Action	
BusObject	Literal	Action	
BusinessValue	Literal	<Business Value>	
DebugLogFile	Literal		
FailedStatus	Literal	NotQueued	
FromAddress	Process Property		MsgSenderAddress
FromPersonal	Process Property		MsgSenderName
InteractionField	Literal	Call Id	
PrimaryOutboundDispatcherBS	Literal	<Business Service Name>	
Priority		<priority value>	
StatusField	Literal	Status	
Subject	Process Property		MsgSubject
Success Status	Literal	Queued	
ThirdPartyId	Process Property		ActivityID

You can change the values, if required; however, you must provide the field values in the angled brackets "<>".

12. Select the GplusMediaRoute business service step and from the Multi Value Property window, select the Output Arguments tab.

13. Add the following output arguments:

Table 75: Output Arguments

Property Name	Type	Output Argument
RouteMessage	Literal	RouteMessage
RouteResult	Literal	RouteResult

14. Save the workflow process.

15. In the History Toolbar, click Back to return to Siebel Tools.

16. In the Workflow Processes list, select <Gplus Email Process Message> and choose Validate.

17. In the Workflow Processes list, select <Gplus Email Process Message> and click Publish/Activate.

End of procedure

Next Steps

- Configure the Siebel Web Services. See, "iWD Routing: Configuring the Siebel Web Services" on .

Procedure:
iWD Routing: Updating the Siebel workflow process for sending email

Purpose: To update the Siebel workflow process for sending emails.

Start of procedure

1. Using the Object Explorer, select the Workflow Process tab.

2. In the Workflow Processes list, query for a process named eMail Response - Client Send Email.

3. In the Workflow Processes list, select eMail Response - Client Send Email and click Revise.

4. In the Workflow Processes list, right-click eMail Response - Client Send Email, and choose Edit Workflow Process. The Workflow Process Designer appears.

5. Add a new business service step block between the Update Reply Activity and the Feedback Categories blocks (or any other location you prefer). Set up the following Business Service step properties:

Table 76: Business Service Step Properties

Parameter	Value
Name	GplusMarkWorkItemDone
Business Service Name	GplusMediaRoute
Business Service Method	MarkWorkItemDone

Update the corresponding connections (see Figure 53 on page 519).
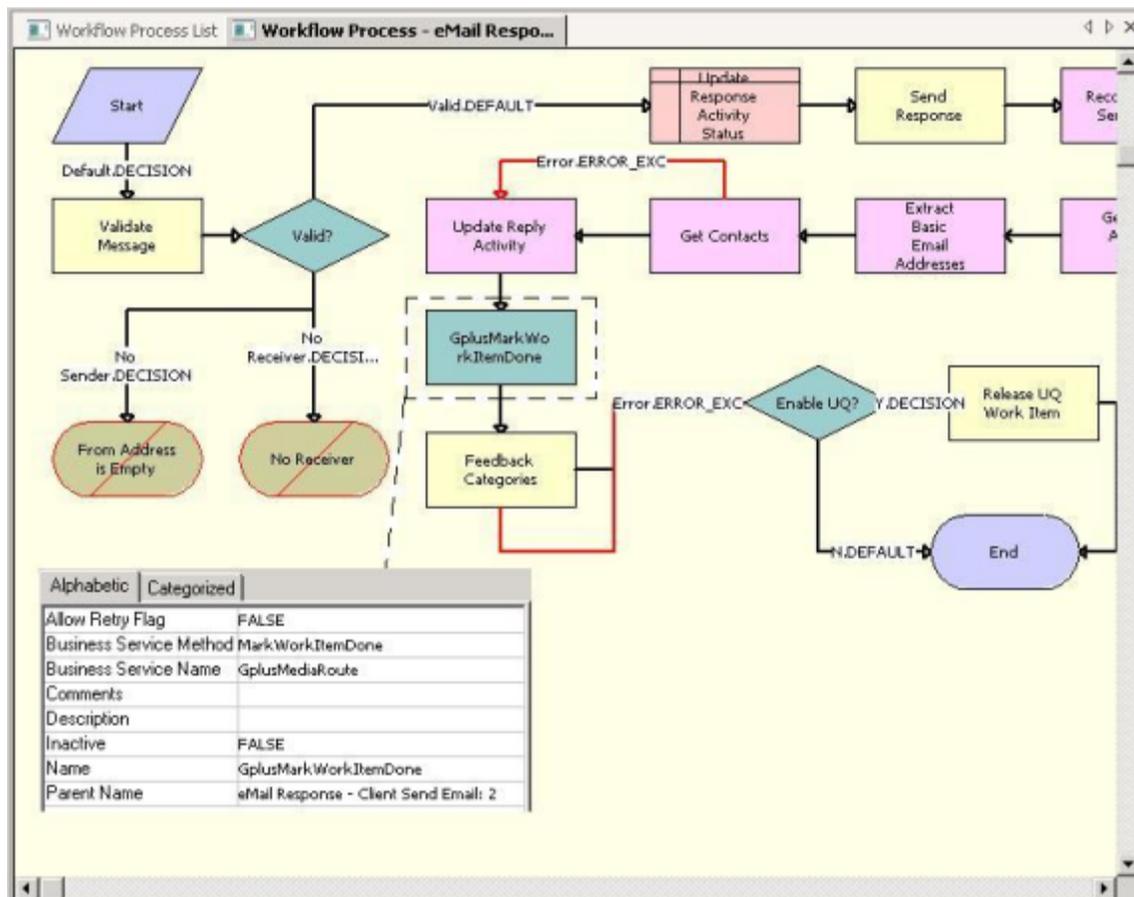
Figure 53: Workflow Process-eMail Response - Client Send Email

6. Select the GplusMarkWorkItemDone business service step. Select the Input Arguments tab in the Multi Value Property window.

7. Add the following input arguments:

Table 77: Input Arguments

Input Argument	Type	Value	Property Name
BusObject	Literal	Action	
BusComp	Literal	Action	
ChildActivityId	Process Property		Activity ID
DebugLogFile	Literal		
FailedStatus	Literal		
InteractionField	Literal	Call ID	
RecIdField	Literal	Activity ID	
ParentRecIdField	Literal	Parent Activity ID	
StatusField	Literal	Status	
SuccessStatus	Literal	Done	

8. Save the workflow process.

9. In the History Toolbar, click Back to return to Siebel Tools.

10. In the Workflow Processes list, select eMail Response - Client Send Email and choose Validate.

11. In the Workflow Processes list, select eMail Response - Client Send Email and click Publish/Activate.

End of procedure

Next Steps

- Configure the Siebel Web Services. See, "iWD Routing: Configuring the Siebel Web Services" on .

Procedure:
iWD Routing: Configuring the Siebel Web Services

Purpose: To configure the Siebel Web Services.

Start of procedure

1. Navigate through the Site Map to the Web Services configuration:

   Select Site Map -> Administration - Web Services -> Outbound Web Services.

2. For each WebService Capture Point (each generated <mediaWebService.XML> file), import the corresponding Outbound Web Service definition (<mediaWebService.XML>).

   a. In the Web Services applet, click Import.

   b. Select an appropriate <mediaWebService.XML> file.

   c. Click Import.

3. For each imported Web Service, ensure that the Address value of the record in the Service Ports applet (located below the Web Services applet) is correct—for example, the value contains the correct URL pointed to the corresponding Webservice Capture Point.

4. Restart the Siebel Server.

End of procedure

Next Steps

- There are no further steps.

Configuring the Multimedia Component Agents

See Chapter 8, "Multimedia Component Agent Administration," on page 416.

Configuring the Siebel eMail Response

See Chapter 9, "Configuring the Siebel eMail Response," on page 492.

The Siebel iWD API Representation

On the Siebel side, the iWD API (application programming interface) is implemented through the GplusMediaRoutingiWD business service. This business service has a set of user properties that represent the default values for some of the parameters. Set these parameter values, if necessary.

The GplusMediaRouteiWD business service provides server functionality and allows submitting requests that are supported by Genesys iWD for controlling the interaction routing process.

The GplusMediaRouteiWD business service contains the following methods:

- createTask Method
- updateTaskByCaptureId Method
- restartTaskByCaptureId Method
- cancelTaskByCaptureId Method
- completeTaskByCaptureId Method
- getTaskByCaptureId Method
- holdTaskByCaptureId Method
- ping Method
- resumeTaskByCaptureId Method

For a detailed description of these iWD methods, see the *intelligent Workload Distribution 8.0. Deployment Guide* for details.

The following tables describe the available methods and their arguments:

- Table 78, "createTask Input Parameters," on page 523
- Table 79, "updateTaskByCaptureId Input Parameters," on page 526
- Table 80, "restartTaskByCaptureId Input Parameters," on page 529
- Table 81, "cancelTaskByCaptureId Input Parameters," on page 531
- Table 82, "completeTaskByCaptureId Input Parameters," on page 534
- Table 83, "getTaskByCaptureId Input Parameters," on page 537
- Table 84, "holdTaskByCaptureId Input Parameters," on page 538
- Table 85, "ping Input Parameters," on page 540
- Table 86, "resumeTaskByCaptureId Input Parameters," on page 541

The following symbols, found in the tables listed above, represent the different methods and arguments:

M:	Represents mandatory arguments.
M*:	Represents parameters that must either be passed as an argument or have a default value set in the Business Service user properties.
M**:	Represents parameters that are required to update the Siebel record status.
O:	Represents the optional arguments.

For all arguments with descriptions that start with "iWD", refer to the iWD documentation for details, if necessary.

createTask Method

The createTask method is used to send a createTask request to the Genesys iWD. This method submits an interaction into a routing process.

For a list of input parameters and their descriptions, see Table 78.

Table 78: createTask Input Parameters

Argument Name	Symbol	Default User Property Name	Description
ThirdPartyId	M		The Siebel work item ID. Used as the CaptureId parameter value for the iWD.
PrimaryOutboundDispatcherBS	O	PrimaryOutbound DispatcherBS	The name of the business service based on CSSWSOutboundDispatcher class that is generated during the Web Service creation (<mediaBusService>) and intended to control the Outbound Web Service, which is used as a primary web service.
BackupOutboundDispatcherBS	O	BackupOutbound DispatcherBS	The name of the business service based on the CSSWSOutboundDispatcher class that is generated during the Web Service creation (<mediaBusService>) and is intended to control the Outbound Web Service, which is used as a backup.
ActivationDateTime	O		iWD: The date and time when the task becomes active. Before the task is activated, it remains queued and is not reprioritized and distributed. If this parameter is not provided, the task becomes active instantly. The format is YYYY-MM-DDThh:mm:ss.
Actor	O	defaultActor	iWD: The user or system that created the task. This argument is used only for auditing purposes and is set to SYSTEM if the value is not provided.
BusinessValue	M*	defaultBusiness Value	iWD: The business value of the task.

Table 78: createTask Input Parameters (Continued)

Argument Name	Symbol	Default User Property Name	Description
Category	O		iWD: The task's category—for example, Followup.
Channel	O	defaultChannel	iWD: The task's media channel—for example, Fax, Email or Webform.
DueDateTime	O		iWD: The date and time by which the task should be completed according to SLA. The format is YYYY-MM-DDThh:mm:ss.
ExpirationDateTime	O		iWD: The time when the task expires and is archived. Only the tasks that have been Canceled, Completed, or Rejected are archived. The format is YYYY-MM-DDThh:mm:ss.
Hold	M*	defaultHold	iWD: Decides whether to initially hold the task. If true, the task is created with the initial status set to NewHeld and is not processed any further until a subsequent call to resume the TaskByCaptureId.
Priority	M*	defaultPriority	iWD: The task priority, which is an integer number used to order tasks submitted to the distribution system.
ProcessId	O	defaultProcessId	iWD: The ID of the process to which the task should be assigned.
Reason	O	defaultReason	iWD
BusObject	M**		The Siebel business object name. * If provided, an attempt to update the record status (for example, activity) is made. There is no default value.

Table 78: createTask Input Parameters (Continued)

Argument Name	Symbol	Default User Property Name	Description
BusComp	M**		The Siebel business component name. * If provided, an attempt to update the record status (for example, activity) is made. There is no default value.
RecIdField	M**		The name of the business component field that contains the Siebel Record ID. The default value is Id.
InteractionField	M**		The name of the business component field that contains the Genesys Interaction Id. There is no default value.
StatusField	M**		The name of the business component field that contains the status information. There is no default value.
SubStatusField	O		The name of the business component field that contains the substatus information. There is no default value.
SuccessStatus	O		The value of the record's status in the case of a successful call to iWD. There is no default value.
SuccessSubStatus	O		The value of the record's substatus in the case of a successful call to iWD. There is no default value.
FailedStatus	O		The value of the record's status in the case of a failed call to iWD. There is no default value.
FailedSubStatus	O		The value of a record's substatus in the case of a failed call to iWD. There is no default value.

Table 78: createTask Input Parameters (Continued)

Argument Name	Symbol	Default User Property Name	Description
UserField	O		The name of a business component field that contains the user name. There is no default value.
UserName	O		The user name that is set for the Siebel record if the UserField argument is provided. There is no default value.

updateTaskByCaptureId Method

The updateTaskByCaptureId method is used to send an updateTaskByCaptureId request to the Genesys iWD. This method updates interaction information.

For a list of input parameters and their descriptions, see Table 79.

Table 79: updateTaskByCaptureId Input Parameters

Argument Name	Symbol	Default User Property Name	Description
ThirdPartyId	M		The Siebel work item ID. Used as the CaptureId parameter value for the iWD.
PrimaryOutboundDispatcherBS	O	PrimaryOutbound DispatcherBS	The name of the business service based on CSSWSOutboundDispatcher class that is generated during the Web Service creation (<mediaBusService>) and intended to control the Outbound Web Service, which is used as a primary web service.
BackupOutboundDispatcherBS	O	BackupOutbound DispatcherBS	The name of the business service based on the CSSWSOutboundDispatcher class that is generated during the Web Service creation (<mediaBusService>) and is intended to control the Outbound Web Service, which is used as a backup.

Table 79: updateTaskByCaptureId Input Parameters (Continued)

Argument Name	Symbol	Default User Property Name	Description
ActivationDateTime	O		iWD: The date and time when the task becomes active. Before the task is activated, it remains queued and is not reprioritized and distributed. If this parameter is not provided, the task becomes active instantly. The format is: YYYY-MM-DDThh:mm:ss.
Actor	O	defaultActor	iWD: The user or system that created the task. This argument is used only for auditing purposes and is set to SYSTEM if the value is not provided.
BusinessValue	M*	defaultBusinessValue	iWD: The business value of the task.
Category	O		iWD: The task's category—for example, Followup.
Channel	O	defaultChannel	iWD: The task's media channel—for example, Fax, Email, or Webform.
DueDateTime	O		iWD: The date and time by which the task should be completed according to SLA. The format is: YYYY-MM-DDThh:mm:ss.
ExpirationDateTime	O		iWD: The time when the task expires and is archived. Only the tasks that have been Canceled, Completed or Rejected are archived. The format is: YYYY-MM-DDThh:mm:ss.
Hold	M*	defaultHold	iWD: Decides whether to initially hold the task. If true, the task is created with the initial status set to NewHeld and is not processed any further until a subsequent call to resume the TaskByCaptureId.

Table 79: updateTaskByCaptureId Input Parameters (Continued)

Argument Name	Symbol	Default User Property Name	Description
Priority	M*	defaultPriority	iWD: The task priority, which is an integer number used to order tasks submitted to the distribution system.
ProcessId	O	defaultProcessId	iWD: The ID of the process to which the task should be assigned.
Reason	O	defaultReason	iWD
BusObject	M**		The Siebel business object name. * If provided, an attempt to update the record status (for example, activity) is made. There is no default value.
BusComp	M**		The Siebel business component name. * If provided, an attempt to update the record status (for example, activity) is made. There is no default value.
RecIdField	M**		The name of the business component field that contains the Siebel Record ID. The default value is Id.
InteractionField	M**		The name of the business component field that contains the Genesys Interaction Id. There is no default value.
StatusField	M**		The name of the business component field that contains the status information. There is no default value.
SubStatusField	O		The name of the business component field that contains the substatus information. There is no default value.

restartTaskByCaptureId Method

The restartTaskByCaptureId method is used to send a restartTaskByCaptureId request to the Genesys iWD. This method restarts the routing process for an interaction.

For a list of input parameters and their descriptions, see Table 80.

Table 80: restartTaskByCaptureId Input Parameters

Argument Name	Symbol	Default User Property Name	Description
ThirdPartyId	M		The Siebel work item ID. Used as the CaptureId parameter value for the iWD.
PrimaryOutboundDispatcherBS	O	PrimaryOutbound DispatcherBS	The name of the business service based on CSSWSOutboundDispatcher class that is generated during the Web Service creation (<mediaBusService>) and intended to control the Outbound Web Service, which is used as a primary web service.
BackupOutboundDispatcherBS	O	BackupOutbound DispatcherBS	The name of the business service based on the CSSWSOutboundDispatcher class that is generated during the Web Service creation (<mediaBusService>) and is intended to control the Outbound Web Service, which is used as a backup.
Actor	O	defaultActor	iWD: The user or system that created the task. This argument is used only for auditing purposes and is set to SYSTEM if the value is not provided.
Hold	M*	defaultHold	iWD: Decides whether to initially hold the task. If true, the task is created with the initial status set to NewHeld and is not processed any further until a subsequent call to resume the TaskByCaptureId.
Reason	O	defaultReason	iWD

Table 80: restartTaskByCaptureId Input Parameters (Continued)

Argument Name	Symbol	Default User Property Name	Description
BusObject	M**		The Siebel business object name. * If provided, an attempt to update the record status (for example, activity) is made. There is no default value.
BusComp	M**		The Siebel business component name. * If provided, an attempt to update the record status (for example, activity) is made. There is no default value.
RecIdField	M**		The name of the business component field that contains the Siebel Record ID. The default value is Id.
InteractionField	M**		The name of the business component field that contains the Genesys Interaction Id. There is no default value.
StatusField	M**		The name of the business component field that contains the status information. There is no default value.
SubStatusField	O		The name of the business component field that contains the substatus information. There is no default value.
SuccessStatus	O		The value of the record's status in the case of a successful call to iWD. There is no default value.
SuccessSubStatus	O		The value of the record's substatus in the case of a successful call to iWD. There is no default value.

Table 80: restartTaskByCaptureId Input Parameters (Continued)

Argument Name	Symbol	Default User Property Name	Description
FailedStatus	O		The value of the record's status in the case of a failed call to iWD. There is no default value.
FailedSubStatus	O		The value of a record's substatus in the case of a failed call to iWD. There is no default value.
UserField	O		The name of a business component field that contains the user name. There is no default value.
UserName	O		The user name that is set for the Siebel record if the UserField argument is provided. There is no default value.

cancelTaskByCaptureId Method

The cancelTaskByCaptureId method is used to send a cancelTaskByCaptureId request to the Genesys iWD. This method cancels an interaction from routing.

For a list of input parameters and their descriptions, see Table 81.

Table 81: cancelTaskByCaptureId Input Parameters

Argument Name	Symbol	Default User Property Name	Description
ThirdPartyId	M		The Siebel work item ID. Used as the CaptureId parameter value for the iWD.
PrimaryOutboundDispatcherBS	O	PrimaryOutbound DispatcherBS	The name of the business service based on CSSWSOutboundDispatcher class that is generated during the Web Service creation (<mediaBusService>) and intended to control the Outbound Web Service, which is used as a primary web service.

Table 81: cancelTaskByCaptureId Input Parameters (Continued)

Argument Name	Symbol	Default User Property Name	Description
BackupOutboundDispatcherBS	O	BackupOutbound DispatcherBS	The name of the business service based on the CSSWSOutboundDispatcher class that is generated during the Web Service creation (<mediaBusService>) and is intended to control the Outbound Web Service, which is used as a backup.
Actor	O	defaultActor	iWD: The user or system that created the task. This argument is used only for auditing purposes and is set to SYSTEM if the value is not provided.
Reason	O	defaultReason	iWD
BusObject	M**		The Siebel business object name. * If provided, an attempt to update the record status (for example, activity) is made. There is no default value.
BusComp	M**		The Siebel business component name. * If provided, an attempt to update the record status (for example, activity) is made. There is no default value.
RecIdField	M**		The name of the business component field that contains the Siebel Record ID. The default value is Id.
InteractionField	M**		The name of the business component field that contains the Genesys Interaction Id. There is no default value.
StatusField	M**		The name of the business component field that contains the status information. There is no default value.

Table 81: cancelTaskByCaptureId Input Parameters (Continued)

Argument Name	Symbol	Default User Property Name	Description
SubStatusField	O		The name of the business component field that contains the substatus information. There is no default value.
SuccessStatus	O		The value of the record's status in the case of a successful call to iWD. There is no default value.
SuccessSubStatus	O		The value of the record's substatus in the case of a successful call to iWD. There is no default value.
FailedStatus	O		The value of the record's status in the case of a failed call to iWD. There is no default value.
FailedSubStatus	O		The value of a record's substatus in the case of a failed call to iWD. There is no default value.
UserField	O		The name of a business component field that contains the user name. There is no default value.
UserName	O		The user name that is set for the Siebel record if the UserField argument is provided. There is no default value.

completeTaskByCaptureId Method

The completeTaskByCaptureId method is used to send a completeTaskByCaptureId request to the Genesys iWD. This method completes an interaction routing process.

For a list of input parameters and their descriptions, see Table 82 on page 534.

Table 82: completeTaskByCaptureId Input Parameters

Argument Name	Symbol	Default User Property Name	Description
ThirdPartyId	M		The Siebel work item ID. Used as the CaptureId parameter value for the iWD.
PrimaryOutboundDispatcherBS	O	PrimaryOutbound DispatcherBS	The name of the business service based on CSSWSOutboundDispatcher class that is generated during the Web Service creation (<mediaBusService>) and intended to control the Outbound Web Service, which is used as a primary web service.
BackupOutboundDispatcherBS	O	BackupOutbound DispatcherBS	The name of the business service based on the CSSWSOutboundDispatcher class that is generated during the Web Service creation (<mediaBusService>) and is intended to control the Outbound Web Service, which is used as a backup.
Actor	O	defaultActor	iWD: The user or system that created the task. This argument is used only for auditing purposes and is set to SYSTEM if the value is not provided.
completedDateTime	O		iWD: The date and time when the task was completed . The format is: YYYY-MM-DDThh:mm:ss.
Reason	O	defaultReason	iWD

Table 82: completeTaskByCaptureId Input Parameters (Continued)

Argument Name	Symbol	Default User Property Name	Description
BusObject	M**		The Siebel business object name. * If provided, an attempt to update the record status (for example, activity) is made. There is no default value.
BusComp	M**		The Siebel business component name. * If provided, an attempt to update the record status (for example, activity) is made. There is no default value.
RecIdField	M**		The name of the business component field that contains the Siebel Record ID. The default value is Id.
InteractionField	M**		The name of the business component field that contains the Genesys Interaction Id. There is no default value.
StatusField	M**		The name of the business component field that contains the status information. There is no default value.
SubStatusField	O		The name of the business component field that contains the substatus information. There is no default value.
SuccessStatus	O		The value of the record's status in the case of a successful call to iWD. There is no default value.
SuccessSubStatus	O		The value of the record's substatus in the case of a successful call to iWD. There is no default value.

Table 82: completeTaskByCaptureId Input Parameters (Continued)

Argument Name	Symbol	Default User Property Name	Description
FailedStatus	O		The value of the record's status in the case of a failed call to iWD. There is no default value.
FailedSubStatus	O		The value of a record's substatus in the case of a failed call to iWD. There is no default value.
UserField	O		The name of a business component field that contains the user name. There is no default value.
UserName	O		The user name that is set for the Siebel record if the UserField argument is provided. There is no default value.

getTaskByCaptureId Method

The getTaskByCaptureId method is used to send a getTaskByCaptureId request to the Genesys iWD. This method is used to request current information about the interaction.

For a list of input parameters and their descriptions, see Table 83 on .

Table 83: getTaskByCaptureId Input Parameters

Argument Name	Symbol	Default User Property Name	Description
ThirdPartyId	M		The Siebel work item ID. Used as the CaptureId parameter value for the iWD.
PrimaryOutboundDispatcherBS	O	PrimaryOutbound DispatcherBS	The name of the business service based on CSSWSOutboundDispatcher class that is generated during the Web Service creation (<mediaBusService>) and intended to control the Outbound Web Service, which is used as a primary web service.
BackupOutboundDispatcherBS	O	BackupOutbound DispatcherBS	The name of the business service based on the CSSWSOutboundDispatcher class that is generated during the Web Service creation (<mediaBusService>) and is intended to control the Outbound Web Service, which is used as a backup.

holdTaskByCaptureId Method

The holdTaskByCaptureId method is used to send a holdTaskByCaptureId request to the Genesys iWD. This method puts an interaction on hold.

For a list of input parameters and their descriptions, see Table 84 on .

Table 84: holdTaskByCaptureId Input Parameters

Argument Name	Symbol	Default User Property Name	Description
ThirdPartyId	M		The Siebel work item ID. Used as the CaptureId parameter value for the iWD.
PrimaryOutboundDispatcherBS	O	PrimaryOutbound DispatcherBS	The name of the business service based on CSSWSOutboundDispatcher class that is generated during the Web Service creation (<mediaBusService>) and intended to control the Outbound Web Service, which is used as a primary web service.
BackupOutboundDispatcherBS	O	BackupOutbound DispatcherBS	The name of the business service based on the CSSWSOutboundDispatcher class that is generated during the Web Service creation (<mediaBusService>) and is intended to control the Outbound Web Service, which is used as a backup.
Actor	O	defaultActor	iWD: The user or system that created the task. This argument is used only for auditing purposes and is set to SYSTEM if the value is not provided.
Reason	O	defaultReason	iWD
BusObject	M**		The Siebel business object name. * If provided, an attempt to update the record status (for example, activity) is made. There is no default value.

Table 84: holdTaskByCaptureId Input Parameters (Continued)

Argument Name	Symbol	Default User Property Name	Description
BusComp	M**		The Siebel business component name. * If provided, an attempt to update the record status (for example, activity) is made. There is no default value.
RecIdField	M**		The name of the business component field that contains the Siebel Record ID. The default value is Id.
InteractionField	M**		The name of the business component field that contains the Genesys Interaction Id. There is no default value.
StatusField	M**		The name of the business component field that contains the status information. There is no default value.
SubStatusField	O		The name of the business component field that contains the substatus information. There is no default value.
SuccessStatus	O		The value of the record's status in the case of a successful call to iWD. There is no default value.
SuccessSubStatus	O		The value of the record's substatus in the case of a successful call to iWD. There is no default value.
FailedStatus	O		The value of the record's status in the case of a failed call to iWD. There is no default value.
FailedSubStatus	O		The value of a record's substatus in the case of a failed call to iWD. There is no default value.

Table 84: holdTaskByCaptureId Input Parameters (Continued)

Argument Name	Symbol	Default User Property Name	Description
UserField	O		The name of a business component field that contains the user name. There is no default value.
UserName	O		The user name that is set for the Siebel record if the UserField argument is provided. There is no default value.

ping Method

The ping method is used to send a ping request to the Genesys iWD. This method checks if iWD Web Service Capture Point is alive or not.

For a list of input parameters and their descriptions, see Table 85 on .

Table 85: ping Input Parameters

Argument Name	Symbol	Default User Property Name	Description
PrimaryOutboundDispatcherBS	O	PrimaryOutbound DispatcherBS	The name of the business service based on CSSWSOutboundDispatcher class that is generated during the Web Service creation (<mediaBusService>) and intended to control the Outbound Web Service, which is used as a primary web service.
BackupOutboundDispatcherBS	O	BackupOutbound DispatcherBS	The name of the business service based on the CSSWSOutboundDispatcher class that is generated during the Web Service creation (<mediaBusService>) and is intended to control the Outbound Web Service, which is used as a backup.

resumeTaskByCaptureId Method

The resumeTaskByCaptureId method is used to send a resumeTaskByCaptureId request to Genesys iWD. This method resumes an interaction routing after it was held.

For a list of input parameters and their descriptions, see Table 86 on .

Table 86: resumeTaskByCaptureId Input Parameters

Argument Name	Symbol	Default User Property Name	Description
ThirdPartyId	M		The Siebel work item ID. Used as the CaptureId parameter value for the iWD.
PrimaryOutboundDispatcherBS	O	PrimaryOutbound DispatcherBS	The name of the business service based on CSSWSOutboundDispatcher class that is generated during the Web Service creation (<mediaBusService>) and intended to control the Outbound Web Service, which is used as a primary web service.
BackupOutboundDispatcherBS	O	BackupOutbound DispatcherBS	The name of the business service based on the CSSWSOutboundDispatcher class that is generated during the Web Service creation (<mediaBusService>) and is intended to control the Outbound Web Service, which is used as a backup.
Actor	O	defaultActor	iWD: The user or system that created the task. This argument is used only for auditing purposes and is set to SYSTEM if the value is not provided.
Reason	O	defaultReason	iWD
BusObject	M**		The Siebel business object name. * If provided, an attempt to update the record status (for example, activity) is made. There is no default value.

Table 86: resumeTaskByCaptureId Input Parameters (Continued)

Argument Name	Symbol	Default User Property Name	Description
BusComp	M**		The Siebel business component name. *If provided, an attempt to update the record status (for example, activity) is made. There is no default value.
RecIdField	M**		The name of the business component field that contains the Siebel Record ID. The default value is Id.
InteractionField	M**		The name of the business component field that contains the Genesys Interaction Id. There is no default value.
StatusField	M**		The name of the business component field that contains the status information. There is no default value.
SubStatusField	O		The name of the business component field that contains the substatus information. There is no default value.
SuccessStatus	O		The value of the record's status in the case of a successful call to iWD. There is no default value.
SuccessSubStatus	O		The value of the record's substatus in the case of a successful call to iWD. There is no default value.
FailedStatus	O		The value of the record's status in the case of a failed call to iWD. There is no default value.
FailedSubStatus	O		The value of a record's substatus in the case of a failed call to iWD. There is no default value.

Table 86: resumeTaskByCaptureId Input Parameters (Continued)

Argument Name	Symbol	Default User Property Name	Description
UserField	O		The name of a business component field that contains the user name. There is no default value.
UserName	O		The user name that is set for the Siebel record if the UserField argument is provided. There is no default value.

11

Deploying the Multimedia Session Failover Handling

This chapter contains the following sections:

Overview

The Multimedia Session Failover Handling feature has been introduced to fully recover from short-term system failovers (an order of magnitude of $x10$ seconds—for example, Interaction Server switchover/unexpected termination/restart and temporal network disconnect), while providing an alternative recovery from longer periods of service unavailability.

The process of configuring the Multimedia Session Failover Handling consists of the following items:

- A special AgentProcessing custom property must be added to the interactions. This value of this property can be set to yes, no, or pending.

- The used business processes that must be modified in order to postpone the further routing of interactions that are recovered after a failure to give a chance to the clients to pull back those interactions. Also, the information about where to pull the interaction from (<queue name>/<view name>) must be attached, as well as the interaction's User Data.

- The Siebel side of the *Gplus* Adapter for Siebel CRM Multimedia must store the agent's state (logged in/out, Ready/Not Ready) and interactions list as well as recover a session after a failure, if possible.

Solution Scope/Known Limitations

The following are the solution scopes and known limitations for the feature.

- The Siebel Server and browser failovers (browser termination or disconnect) are not covered by this solution.

- The Adapter only restores the agent state and pull interactions that were processed by the current agent session. If an agent session is closed, the Adapter does not restore the agent session state and does not pull interactions from the queue.

- The Adapter restores the agent state and the pull interactions upon receiving notification that the failed link has been restored.

- The Adapter restores the agent state and pull interactions, whether or not a timeout has expired. The pull interaction is executed when the connection is restored. To prevent this scenario from happening, the agent must close the browser and then reopen it.

- The Adapter restores the agent state and pull interaction only once after the connection is restored. Therefore, if the pull interaction fails for any reason, the session is not completely restored.

This feature is not available for Genesys chat.

Associated Genesys Components

The Multimedia Session Failover Handling solution consists of the following Genesys components and products:

- Multimedia: Interaction Server;
- Universal Routing Server;
- *Gplus* Communication Server for Siebel CRM;
- *Gplus* Adapter for Siebel CRM Multimedia.

Configuring the Multimedia Solution

This section describes the following procedures that are needed to configure the Multimedia solution:

*   Multimedia Session Failover Handling: Configuring the AgentProcessing interaction property, page 547
*   Multimedia Session Failover Handling: Configuring the Interaction Server application, page 548

Procedure:
Multimedia Session Failover Handling: Configuring the AgentProcessing interaction property

Purpose: To define the AgentProcessing interaction property by using Configuration Manager.

Start of procedure

1. In Configuration Manager, click the Business Attributes folder.

2. In the Business Attributes folder, create an attribute that is called InteractionCustomProperties (type="Custom").

3. Create an AgentProcessing attribute under the Attribute Values folder.

4. Create a translation section under the Annex tab and add the translate-to configuration option with either of the following values:
    *   CustomString<X>" (if you use Interaction Server version 7.6.0 or lower) where <X> corresponds to any unused custom field.
    *   agent_processing (if you use Interaction Server version than 7.6.0 or higher). You must create this field in the Interaction Server table, in interactions, as a varchar type.See Figure 54 on page 548.
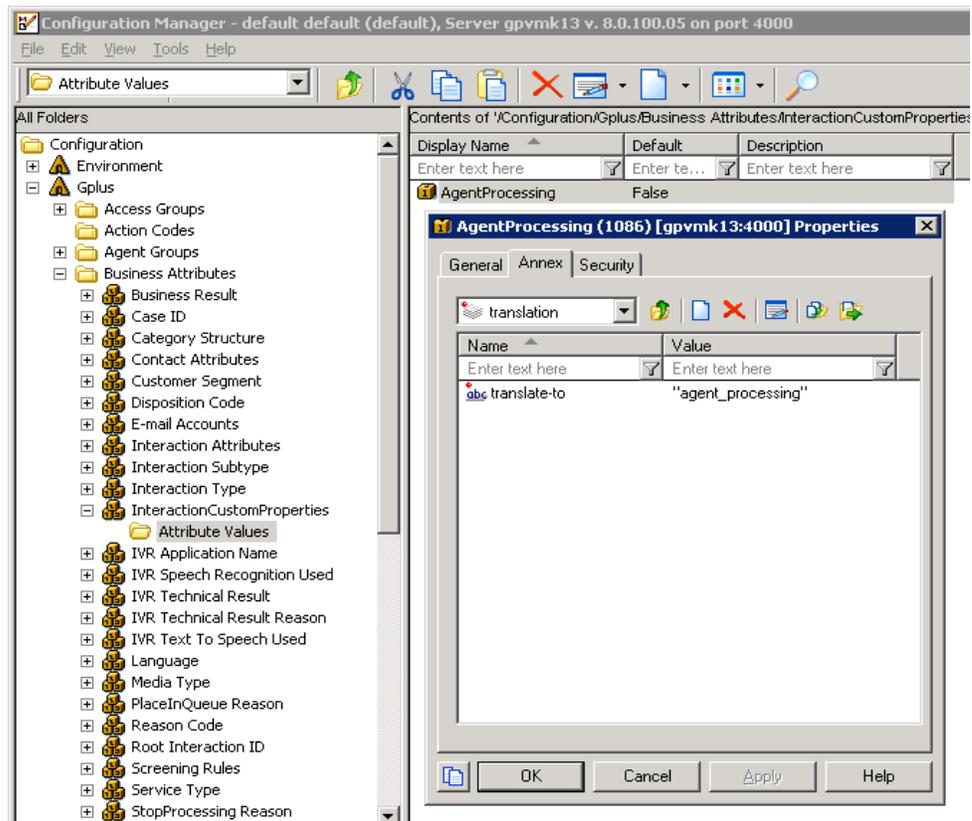
Figure 54: AgentProcessing Properties for Interaction Server 7.6.0 or higher

End of procedure

Next Steps

- Configure the Interaction Server application. See, .

Procedure:
Multimedia Session Failover Handling: Configuring the Interaction Server application

Purpose: To configure the Interaction Server application.

Start of procedure

1. In Configuration Manager, under the Options tab, set the value of the delay-updates configuration option to false in the settings section of the Interaction Server application. See Figure 55.
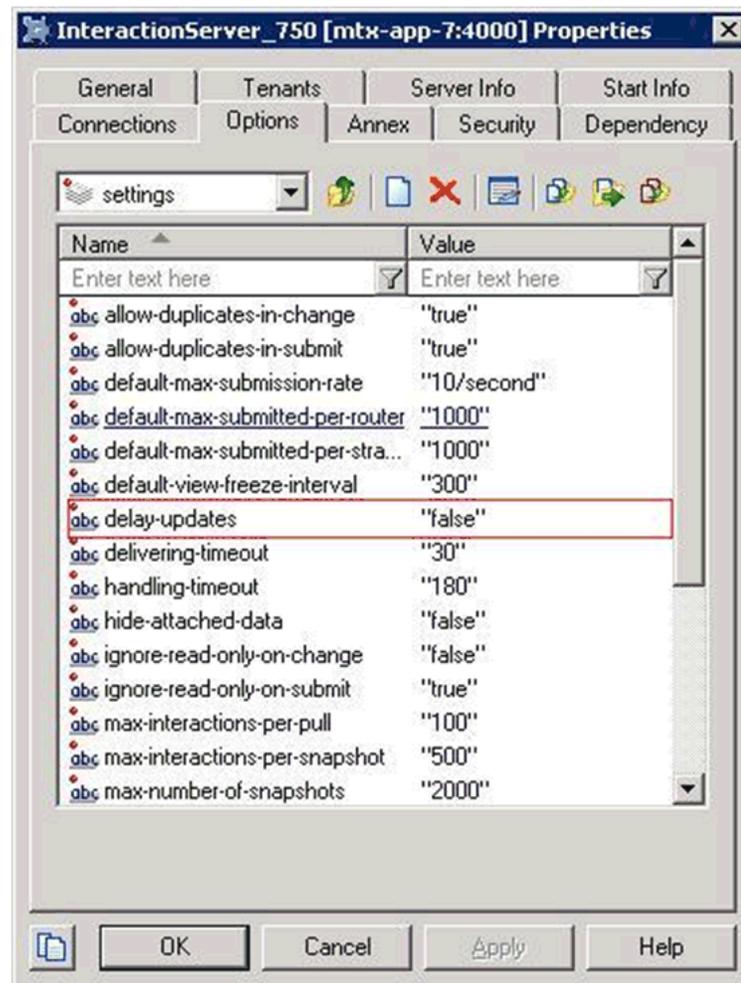
Figure 55: InteractionServer Properties

Note: Setting this configuration option value to false might hinder the performance of Interaction Server. The performance depends on the volume of the interactions that are being processed at the time.

End of procedure

Next Steps

*   Configure the Interaction Routing solution. See the section, "Configuring the Interaction Routing Solution" on page 550.

Configuring the Interaction Routing Solution

This section describes the following procedure that is needed to modify the business process by using the Genesys Interaction Routing Designer:

- Procedure: Multimedia Session Failover Handling: Modifying the existing business processes

Notes:
- This procedure *must* be performed for each queue from which the interactions are restored.
- The current views existing before starting this procedure are known as *pre-existing* views.

Procedure:
Multimedia Session Failover Handling: Modifying the existing business processes

Purpose: To modify the existing business processes.

Start of procedure

1. Start the Genesys Interaction Routing Designer application.
2. In the Login dialog box:
   a. Enter a user name.
   b. Enter a user password.
   c. Click either Details or More options to display additional input login fields.
   d. Enter the application name, which is the instance of the application to which you are logging in.
   e. Enter a host name, which is the name of the computer on which Configuration Server runs.
   f. Enter a port number, which is the number of the communication port that client applications use to connect to Configuration Server.
3. For the pre-existing views, add the following condition:

   AgentProcessing is NULL OR (AgentProcessing!='yes' AND
   AgentProcessing!='pending') OR
   (AgentProcessing='pending' AND (placed_in_queue_at <
   _timestampadd(_current_time(), - ###)))

   (See Figure 56 on .)
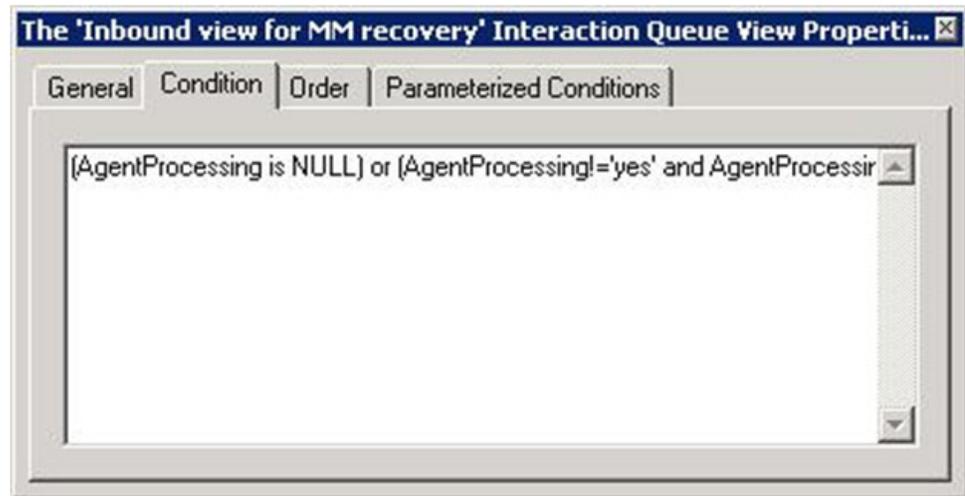
Figure 56: Adding a Condition to the Interaction Queue Workflow

4. Add a unique Prepare_for_Hold_for_MM_recovery view that has the following condition:

- AgentProcessing='yes' AND (placed_in_queue_at < _timestampadd(_current_time(), -###))

5. Add a unique strategy Prepare for Hold for MM recovery which sets the condition AgentProcessing='pending' and places the interaction into the original queue (see Figure 57).
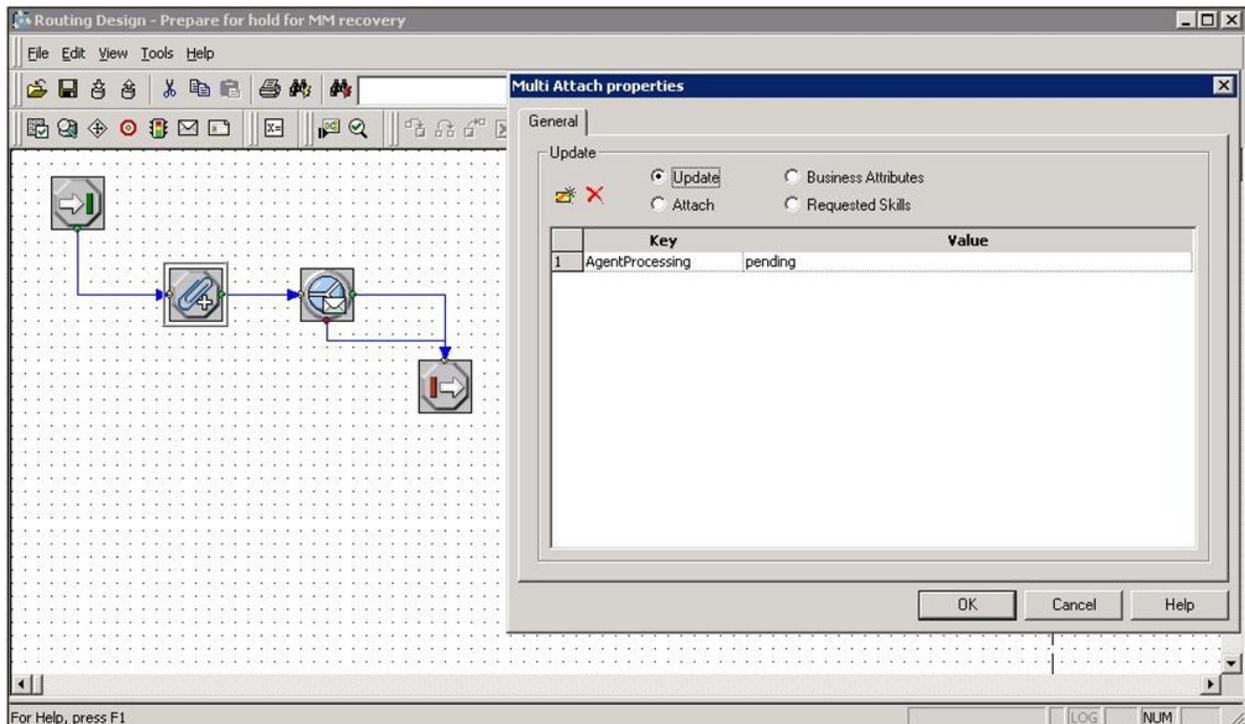
Figure 57: Routing Design—Prepare for Hold for MM Recovery View

6. Connect the Prepare_for_Hold_for_MM_recovery view with the Prepare for Hold for MM recovery strategy.

7. Add a unique Hold_for_Pullout_for_MM_recovery view that has the following condition:

- AgentProcessing='yes' OR (AgentProcessing='pending' AND (placed_in_queue_at > _timestampadd(_current_time(), -###)))

Note: Do not connect the Hold_for_Pullout_for_MM_recovery view to any strategy.

---

8. Set the freeze-interval value to 10 for all views.
The number value should be less than the number (###) of seconds in the condition. Setting the freeze-interval value can be done in the following two ways:

a. In Configuration Manager, by setting the Interaction Server configuration option, default-view-freeze-interval.

b. By setting the Interaction Server configuration option, default-view-freeze-interval, under the General tab of the View Properties window in the Check Interval (sec) field.
This method is applicable only for versions of Interaction Server that are earlier than 7.6.0.

9. Modify the strategies, which are connected to the pre-existing views. The very first step *must* be to alter the User Data as follows:

a. Set the value of the RecoveryPullViewSystem key to <queue name>/<view name>.

b. Set the value of the AgentProcessing key to no (see Figure 58 on).
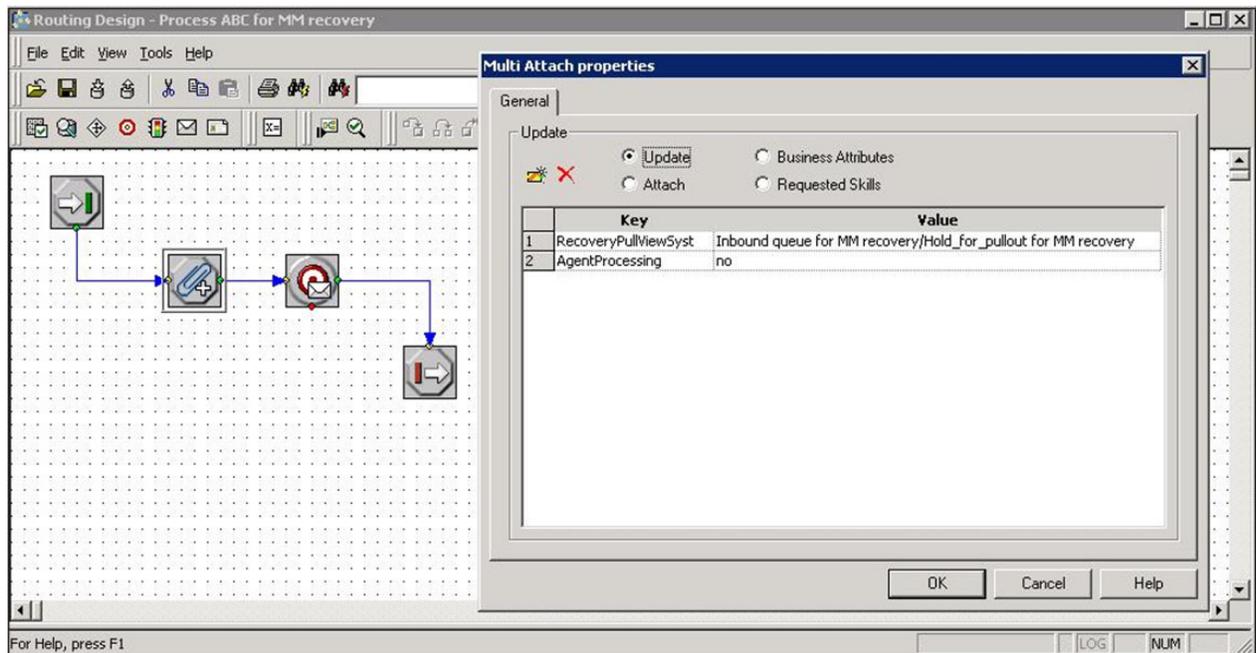
Figure 58: Routing Design—Modifying the Strategy

End of procedure

Next Steps

- Configure the Gplus Adapter for Siebel CRM Multimedia Component. See the section, "Configuring the Communication Server" on .

For all of the view conditions, the value ### should be the same and should correspond to a time interval that is measured in seconds (s) during which interactions can be pulled. The default value to is 300 (seconds or 5 minutes).

Note: The actual period of time during which interactions can be pulled is not very specific, and can be as much as the doubled value of the specified number of seconds.

Example of a Sample Business Process   The installation package for G*plus* Adapter for Siebel CRM Multimedia contains a sample of a business process (SampleBPforMMrecovery.wie). See Figure 59 on for an example of the sample business process.
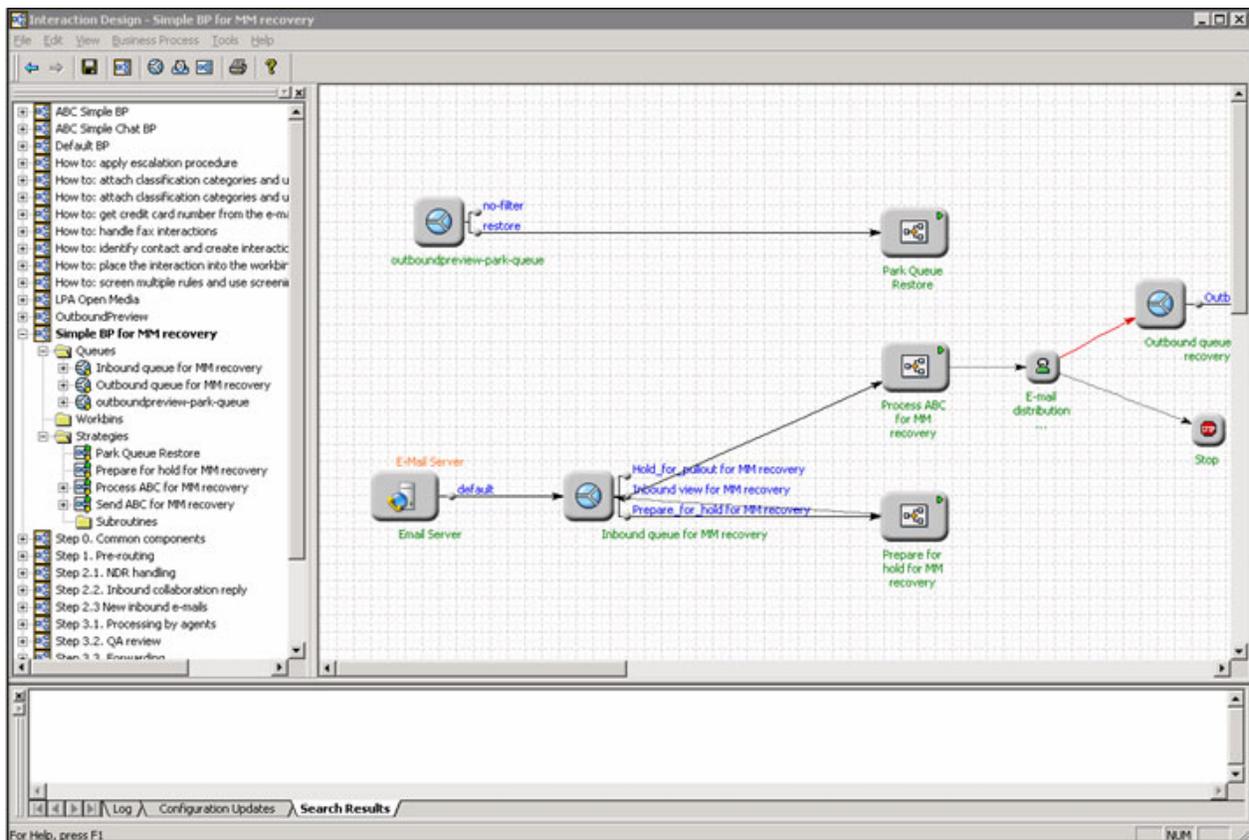
Figure 59: Sample Business Process

Configuring the Communication Server

An appropriate (non-zero) value for the G*plus* Communication Server on-connect-delay configuration option *must* be set. See "Chapter 3, "Deploying the Communication Server for Siebel CRM," on page 45" for details.

Configuring the Multimedia Component

This section describes how to configure the G*plus* Adapter for Siebel CRM Multimedia Component to use the Multimedia Session Failover Handling Feature.

Procedure:
Multimedia Session Failover Handling: Configuring the Multimedia Component

Purpose: To configure the G*plus* Adapter for Siebel CRM Multimedia. Component to use the Multimedia Session Failover Handling Feature.

Start of procedure

1. Using the Siebel Web Client, log in as a Siebel administrator.

2. Navigate to the Communication Administration, All Configurations view.

3. Find the target configuration.

4. Set the OpenMediaSessionRecovery configuration's parameter value to TRUE to turn on the failure-recovery mechanism.

5. Set an appropriate value for the RecoveryPullViewUserDataKey configuration's parameter (see Table 57, "The Multimedia Component Configuration Parameters," on for details).

End of procedure

Next Steps

• There are no further steps.

12 Deploying Basic HTTP Authentication

This chapter describes how to deploy Basic HTTP Authentication and consists of the following sections:

Overview

The Basic HTTP Authentication (RFC 2617) is supported on the following connections:

- From Siebel to the Open Media Server (Media Routing Component)
- From Siebel to the UCS Gateway (Multimedia Component)
- From G*plus* Communication Driver to G*plus* Communication Server (G*plus* Communication Server for the Siebel CRM Component)

Note: Because the Basic HTTP Authentication procedure includes unencrypted passwords into HTTP headers, Genesys does not recommend using this feature for non-secured communications.

The process of deploying Basic HTTP Authentication consists of:

- Configuring Genesys
- Configuring Siebel

Configuring Genesys

To enable HTTP Authentication, you need to complete the following procedures on the Genesys side:

- Configuring Persons Objects
- Configuring Gplus Adapter Applications

Configuring Persons Objects

You set up authentication data (the user name and password) by creating a non-agent Person object(s) that belongs to an existing tenant in the configuration environment. G*plus* Adapter applications will use those Person objects to authenticate client connections.

Procedure:
Configuring a Person object for client authentication

Purpose: To create a Person object for client authentication.

Start of procedure

1. In Configuration Manager, under a Tenant (where all Siebel-related objects are stored, such as Persons, Places, Campaigns, and so on), create a new Person with the following properties:
   - First = <user name>
   - Employee ID = <user name>
   - User Name = <user name>
   - Enter Password = <password>
   - Re-enter Password = <password>
2. Clear the Is Agent check box.

End of procedure

Next Steps

- "Configuring Gplus Adapter Applications" on page 559

Configuring G*plus* Adapter Applications

This section describes how to configure G*plus* Adapter applications that must perform HTTP Authentication for incoming client connections.

Procedure:
Enabling HTTP Authentication in G*plus* Adapter applications

Prerequisites

This procedure applies to the following applications:

*   G*plus* Communication Server
*   G*plus* Open Media Server
*   UCS Gateway

Make sure the corresponding components are installed.

Purpose: To enable HTTP Authentication

Start of procedure

1. In Configuration Manager, right-click the required application and select its Properties.
2. Navigate to the Options tab and click the HTTP section.
3. Set the Authentication option value to true.
4. Click OK to accept the changes.

End of procedure

Next Steps

*   Configure the Siebel side of the environment. See the section, "Configuring Siebel".

Configuring Siebel

To enable HTTP Authentication, complete the following procedures on the Siebel side:

*   Creating Authentication Data Storage
*   Configuring the Siebel Application

Creating Authentication Data Storage

Authentication data must be stored in a text file named GenCommDrv.pwd. This file must be placed in the same directory as the Genesys Communication Driver (GenCommDrv.dll or GenCommDrv.so).

Each line in the GenCommDrv.pwd must have the following format:

<destination host name>:<port>|<user name>|<password>

Where:

- <destination host name>—The host name of the destination server. This name must be lexically identical to the one mentioned in the target URL configured in corresponding places in Siebel.

- <port>—The listening port number of the destination server. This number must correspond to the number used in the target URL configured in corresponding places in Siebel.

- <user name>—The user name for authentication.

- <password>—The user's password.

You can have any number of lines in the file. For example, the GenCommDrv.pwd may looks like:

```
Gencommserverhost.mydoname.com:7060|Aladdin|open sesame 123#
Gencommserverhost:7060|Aladdin|open sesame 123#
UcsGatewayhost.mydoname.com:1900|Aladdin|open sesame 123#
```

You must configure authentication data for each host/port pair used for connections to G*plus* Adapter servers that require HTTP Authentication. It does not matter whether the same user credentials are used for all host/port pairs or different ones.

Notes: • Because the GenCommDrv.pwd contains unencrypted passwords, Genesys recommends making it readable to Siebel Server processes only.

• The GenCommDrv.pwd file name is fixed and case-sensitive for UNIX operating systems.

Configuring the Siebel Application

To enable sending HTTP Authentication data for a connection from G*plus* Communication Driver to G*plus* Communication Server, place the GenCommDrv.pwd into the same directory as the GenCommDrv.dll(so). No additional actions are required.

To enable sending HTTP Authentication data for a connection from Siebel servers to the Open Media server or UCS Gateway, set the AuthOn user property of the Genesys Auth Provider business service to true and re-compile .srf. The same GenCommDrv.pwd file is used as a source of authentication data.

Procedure:
Enabling HTTP Authentication for client connections from Siebel servers to G*plus* Adapter servers

Prerequisites

*   This procedure must be done after the Multimedia Component is installed or during its installation.

Purpose: To enable sending HTTP Authentication data for connections from Siebel servers.

Start of procedure

1. In Siebel Tools, use Object Explorer to navigate to the Business Service folder.
2. Select the Genesys Auth Provider business service.
3. On the object tree, select the Business Service User Prop subfolder of the Business Service folder.
4. On the right pane, select the AuthOn user property and set its value to true.
5. Re-compile .srf.

End of procedure

GENESYS

13 Application Monitoring

Application monitoring provides application-health data (metrics) to external monitoring systems.

Reporting is performed using SNMP traps that an application sends to the configured destination. One SNMP trap corresponds to one metric.

There are two types of metrics available: immediate and periodic. Immediate metrics are reported as they are appear in the application. Periodic metrics are reported with a predefined period of time.

Table 87: List of Metrics

Description	Name	Type	Data Type	Valid Values	Actual Data	Comments
Parse error during marshaling between GCD-GCS	NetParsingError	Immediate	Integer		Socket ID	Socket ID is auxiliary data
Availability of Multimedia component	MultimediaAvailability	Immediate	Boolean	0,1		0 - unavailable 1- available
Availability of Voice component	VoiceAvailability	Immediate	Boolean	0,1		0 - unavailable 1- available
PID of running process	ProcessID	Immediate	Integer			Reported only once at startup
Request from GCS to Siebel has expired	G2SRequestExpired	Immediate	String		Request description	Request descriptions auxiliary data

Table 87: List of Metrics (Continued)

Description	Name	Type	Data Type	Valid Values	Actual Data	Comments
Request from Siebel to GCS has expired	S2GRequestExpired	Immediate	String		Request description	Request descriptions auxiliary data
Average request (from Siebel to GCS) processing time for a period of 30 seconds	S2GAvgReqProcessingTime	Periodic	Integer			Time in seconds
Average request (from GCS to Siebel) processing time for a period of 30 seconds	G2SAvgReqProcessingTime	Periodic	Integer			Time in seconds

The reported metrics do not provide health status. Monitoring systems should check metrics for critical conditions to make decisions about the application health. For example, if the request execution time nears the configured request timeout, it may indicate problems, such as performance issues, misconfigurations, network issues, and so on.

The MIB for SNMP traps is distributed with the G*plus* Communication Server for the Siebel CRM component and is located at its installation folder in the GPLUS-MIB.txt file.

![Genesys logo]

14 Uninstallation Instructions

This chapter provides uninstallation instructions for the G*plus* Adapter 8.0 for Siebel CRM.

This chapter includes the following sections:

Overview

This chapter describes how to uninstall the G*plu*s Adapter 8.0 for Siebel CRM. The Adapter must be uninstalled one component at a time. To uninstall a component, refer to the corresponding section in this chapter.

Uninstallation is typically performed for the following reasons:

- You do not want to use one or more of the G*plus* Adapter's components.

- You want to upgrade to a newer (more current) release of the G*plus* Adapter, which may require uninstallation of the existing release.

Note: Some G*plus* Adapter 8.0 for Siebel CRM components share common files. Therefore, uninstalling one component may delete a file or the files necessary for the remaining component. This problem can be resolved in the following way: after you uninstall a component, you can reinstall the Adapter for the remaining component.

Uninstalling the Communication Server for Siebel CRM

This section describes how to uninstall the G*plus* Communication Server for Siebel CRM.

The process of uninstalling the G*plus* Communication Server for Siebel CRM includes two general procedures:

1. Delete the files installed by the G*plus* Communication Server for Siebel CRM in the installation directory.

2. Delete the files installed in the Siebel Server directory. This step is necessary when the Siebel Server runs on a UNIX platform, or when the Siebel Server and the G*plus* Communication Server for Siebel CRM are installed on different hosts.

Procedure:
Uninstallation: Deleting the files installed by the Communication Server for Siebel CRM

Purpose: To delete the files installed by the G*plus* Communication Server for Siebel CRM during deployment.

Start of procedure

For Windows Users
1. Complete the uninstallation program for the G*plus* Communication Server for Siebel CRM by using Add/Remove Programs in the Control Panel on the host on which you installed the component.

For UNIX Users
2. Delete the G*plus* Communication Server for Siebel CRM's files by removing the directory that was used as the destination directory during installation.

3. Delete the libGenCommDrv.so file from the Siebel Server directory.

End of procedure

Next Steps

• Delete the files installed by the G*plus* Communication Server for Siebel CRM from the Siebel Server directory. See, Procedure: Uninstallation: Deleting the files installed by the Communication Server for Siebel CRM from the Siebel Server directory.

**Procedure:
Uninstallation: Deleting the files installed by the
Communication Server for Siebel CRM from the Siebel
Server directory**

Purpose: To delete the files installed by the G*plus* Communication Server for
Siebel CRM from the Siebel Server directory.

Start of procedure

For Windows Users

1. Remove the GenCommDrv.ini and GenCommDrv.dll files from the Siebel
   Server's BIN directory.

For UNIX Users

2. Remove the GenCommDrv.ini and libGenCommDrv.so files from the Siebel
   Server's lib directory.

End of procedure

Next Steps

• There are no further steps.

Uninstalling the Configuration Synchronization Component

This section describes how to uninstall the G*plus* Adapter for Siebel CRM
Configuration Synchronization Component (Configuration Synchronization
Component).

The process of uninstalling includes the following three general procedures:

1. Changing the Genesys configuration.

2. Changing the Siebel configuration.

3. Uninstalling the Configuration Synchronization Component.

Note: The G*plus* Adapter for Siebel CRM Configuration Synchronization
Component was known as the Configuration Component prior to
release 7.0.

Changing the Genesys Configuration to Uninstall the Configuration Synchronization Component

This section describes how to change the Genesys configuration in order to
uninstall the Configuration Synchronization Component.

Prestart Information

Before changing the configuration process, you should have the following Genesys Framework applications running:

- Configuration Database
- Configuration Server
- Configuration Manager

As a Configuration Manager user, you should have sufficient privileges to make changes to the Configuration Layer objects.

Procedure:
Uninstallation: Deleting the objects and template in the Genesys configuration for the Configuration Synchronization Component

Purpose: To change the Genesys configuration by deleting the Application object, the Application template, and the Person object to uninstall the Configuration Synchronization Component.

Start of procedure

Deleting the Application Object

1. In Configuration Manager, right-click the Application object.
2. Select Delete.

   The Delete dialog box displays.
3. Click Yes to delete the Application object.

Deleting the Application Template

3. In Configuration Manager, right-click the Application template.

   The default name for the template is either
   Gplus_SiebelCRM_Config_Synch_800_for_CL_70.apd
   or
   Gplus_SiebelCRM_Config_Synch_800_for_CL_71_and_higher.apd
4. Select Delete.

   The Delete dialog box displays.
5. Click Yes to delete the Application template.

Deleting the Person Object

6. In Configuration Manager, right-click the Person object.

   The default user name for the object is siebel7gplus.
7. Select Delete.

   The Delete dialog box displays.

8. Click Yes to delete the Person object.

    This completes the changes you need to make to the Genesys configuration.

End of procedure

Next Steps

• Change the Siebel configuration to uninstall the COnfiguration Synchronization Component. See the section, "Changing the Siebel Configuration to Uninstall the Configuration Synchronization Component".

Changing the Siebel Configuration to Uninstall the Configuration Synchronization Component

This section describes how to change the Siebel configuration in order to uninstall the Configuration Synchronization Component.

Prestart Information

Before starting this part of the configuration process you must have the Siebel Tools application running.

You should be connected to the local copy of the Siebel Server database and have sufficient privileges to modify the Siebel Server repository.

Deleting the Runtime Events in the Siebel Configuration

To delete the runtime events that are required by the Configuration Synchronization Component, you must run the UninstallRunTimeEvents method of the Genesys Configuration Synchronization Siebel Business Service once using the Siebel Business Service Simulator applet.

Procedure:
Uninstallation: Deleting the Configuration Synchronization Component Runtime Events in the Siebel Configuration

Purpose: To delete the runtime events in the Siebel Configuration in order to uninstall the Campaign Synchronization Component.

Start of procedure

1. Navigate through the Site Map to Business Service Administration > Business Service Simulator.

2. In the Service Methods applet, create a new record.

3. Specify the following parameters for the new record:
    - Service Name: Genesys Config Synchronization
    - Method Name: UninstallRunTimeEvents

4. Click the Run button.

Note: After running this method, you should navigate in the Siebel client to Administration Runtime Events > Events, and select the system menu item Reload Runtime Events.

End of procedure

Next Steps

- Use the Siebel Tools to change the Siebel configuration. See the section, "Using Siebel Tools to Change the Siebel Configuration for the Configuration Synchronization Component".

Using Siebel Tools to Change the Siebel Configuration for the Configuration Synchronization Component

Use Siebel Tools to compile an updated release of the Siebel repository file for one or more of the Siebel applications you use on your Siebel Server, which you will then deploy on the server.

For information about using Siebel Tools, see the Siebel documentation.

Procedure:
Uninstallation: Checking out the existing Configuration Synchronization Component projects

Purpose: To check out the Genesys Configuration Synchronization and Personalization projects from the Siebel repository.

Start of procedure

1. In Siebel Tools, select Tools > Check Out...

    The Check Out dialog box displays.

2. In the Projects list, select the projects to be checked out.

3. Click Check Out.

Note: Make sure that the following projects are checked out from the Siebel repository:
- Genesys Configuration Synchronization
- Personalization

Next Steps

- Delete the objects from the Configuration Synchronization Component project. See, Procedure: Uninstallation: Deleting the objects from the Configuration Synchronization Component project in the Siebel repository.

Procedure:
Uninstallation: Deleting the objects from the Configuration Synchronization Component project in the Siebel repository

Purpose: To delete the objects from the projects in the Siebel repository.

Start of procedure

1. In Siebel Tools, in the Project field of the Object Explorer, select All Projects.
2. From the Object Explorer, select the Business Component folder.
    The Business Components window displays.
3. Select the Genesys CommSvr CM Agents / ACD Queues business component.
4. Right-click the selected record.
5. Select Delete Record from the drop-down menu.
6. When you see the prompt, ''Are you sure you want to delete the current record?'', click Yes to delete the record.

7. Repeat Steps 2 – 6 for all Siebel repository elements referenced in Table 88 on page 572.

Table 88: Objects in the Genesys Configuration Synchronization Project

Type	Name
Business Component	Genesys CommSrv CM Agents / ACD Queues
Business Component	Genesys CommSrv CM Agents /Adapter Profiles / Adapters
Business Component	Genesys CommSrv CM Agents / Telesets
Business Component	Genesys Profiles / Organizations
Business Component	Genesys Users / Organizations
Business Object	Genesys CommSrv CM Administration
Business Object	Genesys Users / Organizations
Business Service	Genesys Config Synchronization
Integration Object	Genesys Agent
Link	Genesys Users / Organizations/Employee Skill

End of procedure

Next Steps

- Delete the modification to the Personalization Action Set Business Component. See, Procedure: Uninstallation: Changing the Personalization Action Set business component.

Procedure: Uninstallation: Changing the Personalization Action Set business component

Purpose: To delete modifications that were made to the Personalization Action Set business component during deployment.

Start of procedure

1. From the Object Explorer, select the Business Component folder and navigate to the Personalization Action Set business component.

2. Using the Types tab of the Object Explorer, select the Field folder under the Business Component folder.

   The Fields window displays.

3. Select the Action Set ID field.

4. Right-click the selected record.

4. Select Delete Record from the drop-down menu.

5. When prompted, "Are you sure you want to delete the current record?", click Yes to delete the record.

6. Select the Actions field.

7. Right-click the selected record.

8. Select Delete Record from the drop-down menu.

9. When prompted, "Are you sure you want to delete the current record?", click Yes to delete the record.

10. Using the Types tab of the Object Explorer, select the Multi Value Field folder under the Business Component folder.

11. The Multi Value Fields window displays.

12. Make sure the Actions field is deleted.

End of procedure

Next Steps

- Compile the Siebel repository file. See, Procedure: Uninstallation: Compiling the Siebel repository file.

Procedure:
Uninstallation: Compiling the Siebel repository file

Purpose: To compile the Siebel repository file.

Start of procedure

1. In Siebel Tools, select Tools > Compile Projects...

2. Select Locked projects.

3. In the Siebel Repository File edit box, enter the name of the repository file.

4. Click Compile.

The status bar at the bottom of the Object Compiler window indicates when the compilation is finished.

5. When the compilation is finished, close Siebel Tools.

Note: It is possible for siebel_assert_xxx.txt file(s) to be generated by the Siebel environment after compiling the Siebel repository file.

End of procedure

Next Steps

- Delete the Configuration Synchronization Component from the Siebel Server. See, Procedure: Uninstallation: Deleting the Configuration Synchronization Component from the Siebel Server.

Procedure:
Uninstallation: Deleting the Configuration Synchronization Component from the Siebel Server

Purpose: To change the Siebel configuration by deleting the Configuration Synchronization Component implementation from the Siebel Server.

Start of procedure

Deploying the Repository File

1. Deploy the compiled Siebel repository file on your Siebel Server.

Notes: Also, you may need to generate and deploy browser scripts for the new repository file. For further information about how to deploy an updated repository file to the Siebel Server, refer to the Siebel documentation.

Deleting the Subsystems

2. Delete the following EAI Transport Data Handling and Connection Subsystems, which were created in your Siebel environment during deployment (listed here by their Alias):

- GplusConfSynchExportAllData
- GplusConfSyncConnectionPrimary
- GplusConfSyncConnectionBackup

3. Use the Delete named subsystem command in the Siebel Server Manager (srvrmgr) utility to delete these subsystems.

For information about how to use Siebel Server Manager, see the Siebel documentation.

Updating the Configuration Files

4. To update the eai.cfg file in the Siebel Server installation, open it and delete the following line from the [HTTP Services] section:

GplusConfSynchExportAllData = GplusConfSynchExportAllData.

Deleting the Genesys Tables

5. Delete the Genesys tables from the Siebel Server Database.

> **Note:** Deleting Genesys tables from the Siebel repository does not delete them from the Siebel database. You must delete them manually. The Genesys Campaign Synchronization project contains two tables: CX_GEN_QUEUE_ITM and CX_GEN_SYN_SUM. You can use any database tool (such as, Microsoft SQL Enterprise Manager or Oracle Enterprise Manager) to drop these tables.

End of procedure

Next Steps

- Uninstall the Configuration Synchronization Component on a Windows or a UNIX platform. See the section, "Uninstalling the Configuration Synchronization Component on a Windows or UNIX Platform".

Uninstalling the Configuration Synchronization Component on a Windows or UNIX Platform

Select one of the following topics depending on your environment.

For a Windows Environment

Complete the uninstallation program for the Configuration Synchronization Component by using the Add/Remove Programs in the Control Panel on the host where you installed the Configuration Synchronization Component.

For a UNIX Environment

Delete the Configuration Synchronization Component's files by removing the directory that was used as the destination directory during installation. Select one of the following procedures depending on your environment.

You have deleted the G*plus* Adapter for Siebel CRM Configuration Synchronization Component from your environment.

Uninstalling the Campaign Synchronization Component

The process of uninstalling the *Gplus* Adapter for Siebel CRM Campaign Synchronization Component (Campaign Synchronization Component) includes three general procedures:

1. Changing the Genesys configuration.

2. Changing the Siebel configuration.

3. Uninstalling the Campaign Synchronization Component.

Note: Stop all synchronization processes before uninstalling the Campaign Synchronization Component.

Changing the Genesys Configuration to Uninstall the Campaign Synchronization Component

This section describes how to change the Genesys configuration in order to uninstall the Campaign Synchronization Component.

Prestart Information

Before starting the configuration process you should have the following Genesys Framework applications running:

• Configuration Database

• Configuration Server

• Configuration Manager

As a Configuration Manager user, you should have sufficient privileges to make changes to the Configuration Layer objects.

Procedure:
Uninstallation: Deleting the Application object, Application template, and the Person object in the Genesys Configuration for the Campaign Synchronization Component

Purpose: To change the Genesys configuration by deleting the Application object, the Application template, and the Person object to uninstall the Campaign Synchronization Component.

Start of procedure

Deleting the Application Object

1. In Configuration Manager, right-click the Application object.

2. Select Delete.

   The Delete dialog box displays.

3. Click Yes to delete the Application object.

4. Repeat Steps 1- 3 for all Application objects related to the Campaign Synchronization Object that is being removed.

Deleting the Application Template

5. In Configuration Manager, right-click the Application template.

   The default name for the template is either Gplus_SiebelCRM_Camp_Synch_802_for_CL_70.apd or Gplus_SiebelCRM_Camp_Synch_802_for_CL_71_and_higher.apd

6. Select Delete.

   The Delete dialog box displays.

7. Click Yes to delete the Application template.

Deleting the Person Object

8. In Configuration Manager, right-click the Person object.

   The default user name for the object is siebel7gplus.

9. Select Delete.

   The Delete dialog box displays.

10. Click Yes to delete the Person object.

    This completes the changes you need to make to the Genesys configuration.

End of procedure

Next Steps

- Change the Siebel configuration. See the section, "Changing the Siebel Configuration to Uninstall the Campaign Synchronization Component".

Changing the Siebel Configuration to Uninstall the Campaign Synchronization Component

This section describes how to change the Siebel configuration in order to uninstall the Campaign Synchronization Component.

Prestart Information

Before starting this part of the configuration process, you must have the Siebel Tools application running.

You must be connected to the local copy of the Siebel Server database and have sufficient privileges to modify the Siebel Server repository.

Deleting the Runtime Events in the Siebel Configuration

To delete the runtime events that are required by the Campaign Synchronization Component, you must run the UninstallRunTimeEvents method of the Genesys CampSynch Tools Siebel Business Service once using the Siebel Business Service Simulator applet.

Procedure:
Uninstallation: Deleting the Campaign Synchronization Component runtime events in the Siebel configuration

Purpose: To delete the runtime events in the Siebel Configuration in order to uninstall the Campaign Synchronization Component.

Summary

In this procedure, you will run the UninstallRunTimeEvents method of the Genesys CampSynch Tools Siebel Business Service once by using the Siebel Business Service Simulator applet.

Start of procedure

1. Log into Siebel as an administrator.
2. Navigate through the Site Map to Business Service Administration > Business Service Simulator.
3. In the Service Methods applet, create a new record.
4. Specify the following parameters for the new record:
   - Service Name: Genesys CampSynch Tools
   - Method Name: UninstallRunTimeEvents
5. Click the Run button.

   After running the method, navigate in Siebel client to Administration > Runtime Events > Events, and select the Reload Runtime Events system menu item.

End of procedure

Next Steps

- Delete the workflow processes from the Siebel Server. See the section, "Deleting the Workflow Processes from the Siebel Server" on page 579.

Deleting the Workflow Processes from the Siebel Server

To delete the workflow processes that are required by the Campaign Synchronization Component, you must use the Siebel Web Client.

Procedure:
Uninstallation: Deleting the workflow processes from the Siebel Server

Purpose: To delete the workflow processes on the Siebel Server.

Start of procedure

1. Select Site Map > Administration > Business Process, Workflow Deployment.
2. In the Active Workflow Processes applet, select the Genesys CaS Campaign WF workflow process.
3. Choose the Delete Process menu option.

   The Genesys CaS Campaign WF workflow process is removed.
4. Repeat steps Steps 3 and 4 for the Genesys CaS CR WF workflow process.

   The Genesys CaS CR WF workflow process is removed.

End of procedure

Next Steps

- Change the Siebel configuration for the Campaign Synchronization Component by using Siebel Tools. See the section, "Using Siebel Tools to Change the Siebel Configuration for the Campaign Synchronization Component".

Using Siebel Tools to Change the Siebel Configuration for the Campaign Synchronization Component

You will use Siebel Tools to compile an updated release of the Siebel repository file for one or more of the Siebel applications you use on your Siebel Server, which you will then deploy on the server.

For information about using Siebel Tools, see the Siebel documentation.

Procedure:
Uninstallation: Checking out the existing Campaign Synchronization Component projects

Purpose: To check out the Genesys Campaign Synchronization and Personalization projects from the Siebel repository.

Start of procedure

1. In Siebel Tools, select Tools > Check Out...

    The Check Out dialog box displays.

2. In the Projects list, select the projects to be checked out. For example— Campaign.

3. Click Check Out.

End of procedure

Next Steps

*   Delete the objects from the projects. See, Procedure: Uninstallation: Deleting the Campaign Synchronization Component objects from the projects in the Siebel repository.

Procedure:
Uninstallation: Deleting the Campaign Synchronization Component objects from the projects in the Siebel repository

Purpose: To delete the Campaign Synchronization Component objects from the projects in the Siebel repository.

Start of procedure

1. In Siebel Tools, in the Project field of the Object Explorer, select the Campaign Synchronization2 project.

2. Delete all of the objects related to the selected project using the Object Explorer.

End of procedure

Next Steps

• Delete the modifications to the Campaign List Contact business component. See, .

Procedure: Uninstallation: Changing the Campaign List Contact Business Component

Purpose: To delete the modifications that were made to the Campaign List Contact business component during the deployment.

Start of procedure

1. In Siebel Tools, in the Project field of the Object Explorer, select Campaign.

2. From the Object Explorer, select the Business Component folder.

   The Business Components window displays.

3. Select the Campaign List Contact business component.

4. Using the Types tab of the Object Explorer, select the Field folder under the Business Component folder.

   The Fields window displays.

5. Select the Calculated Contact Id field.

6. Right-click the selected record.

7. Select Delete Record from the drop-down menu.

8. When you see the prompt, ''Are you sure you want to delete the current record?'', click Yes to delete the record.

9. Repeat Steps 4 – 8 for all elements of the Campaign List Contact business component referenced in Table 89 on page 581.

Table 89: Elements to Delete from the Campaign List Contact Business Component

Type	Name
Field	Calculated Time Zone Name
Field	Prospect Time Zone Id
Field	Prospect Time Zone Name
Field	Status LIC
Field	Time Zone Id

Table 89: Elements to Delete from the Campaign List Contact Business Component (Continued)

Type	Name
Field	Time Zone Name
Join	S_PRSP_TIMEZONE
Join	S_TIMEZONE

10. Select the Status field.

11. Change the value for the PickList property to PickList Campaign Status.

End of procedure

Next Steps

- Remove the link to the Administration - Campaign Synchronization screen. See, Procedure: Uninstallation: Removing the link to the Administration - Campaign Synchronization screen from the Site Map of the Application object.

Procedure:
Uninstallation: Removing the link to the Administration - Campaign Synchronization screen from the Site Map of the Application object

Purpose: To remove the link to the Administration - Campaign Synchronization screen from the Site Map of the Application object.

Start of procedure

1. Check-out or lock the project that corresponds to the Application object that you are using. See, Procedure: Campaign Synchronization Component: Checking out existing projects, on page 144.

2. In Siebel Tools, in the Project field of the Object Explorer, select the name of the project of the Siebel application that you are using—for example: Siebel Universal Agent.

3. Select the Application folder.

The Application window displays.

4. Select the Application object that you are using.

5. Using the Types tab of the Object Explorer, select the Screen Menu Item folder under the Application folder.

The Application Screen Menu Items window displays.

6. Remove the record related to the Administration - Campaign Synchronization screen from the Screen Menu Items table.

End of procedure

Next Steps

- Remove the Administration - Campaign Synchronization screen from the Page tab of the Application object. See, Procedure: Uninstallation: Removing the Administration - Campaign Synchronization screen from the Page tab of the Application Object.

Procedure:
Uninstallation: Removing the Administration - Campaign Synchronization screen from the Page tab of the Application Object

Purpose: To remove the Administration - Campaign Synchronization screen from the Page tab of the Application object.

Start of procedure

1. Check-out or lock the project that corresponds to the Application object that you are using. See, Procedure: Campaign Synchronization Component: Checking out existing projects, on page 144.

2. In Siebel Tools, in the Project field of the Object Explorer, select the name of the project of the Siebel application that you are using—for example: Siebel Universal Agent.

3. Select the Application folder.

The Application window displays.

4. Select the Application object that you are using.

5. Using the Types tab of the Object Explorer, select the Page Tab folder under the Application folder.

The Application Page Tabs window displays.

6. Remove the record related to the Administration - Campaign Synchronization screen from the Page Tabs table.

End of procedure

Next Steps

- Compile the Siebel repository file. See, Procedure: Uninstallation: Compiling the Siebel repository file for the Campaign Synchronization Component.

Procedure:
Uninstallation: Compiling the Siebel repository file for the Campaign Synchronization Component

Purpose: To compile the Siebel repository file.

Start of procedure

1. In Siebel Tools, select Tools > Compile Projects...

2. Select Locked projects.

3. In the Siebel Repository File edit box, enter the name of the repository file.

4. Click Compile.

   The status bar at the bottom of the Object Compiler window indicates when the compilation is finished.

5. When the compilation is finished, close Siebel Tools.

Note: It is possible for the siebel_assert_xxx.txt file(s) to be generated by the Siebel environment after compiling the Siebel repository file.

End of procedure

Next Steps

- Delete the Campaign Synchronization Component from the Siebel Server. See, Procedure: Uninstallation: Deleting the Campaign Synchronization Component implementations from the Siebel Server.

Procedure:
Uninstallation: Deleting the Campaign Synchronization Component implementations from the Siebel Server

Purpose: To change the Siebel configuration by deleting the Campaign Synchronization Component implementation from the Siebel Server.

Start of procedure

Deploying the   1. Deploy the compiled Siebel repository file on your Siebel Server.
Repository File

> **Note:**   Also, you may need to generate and deploy browser scripts for
> the new repository file. For further information about how to
> deploy an updated repository file to the Siebel Server, refer to the
> Siebel documentation.

Deleting the   2. Delete the connection subsystems, which were created in your Siebel
Subsystems   environment for use with the Campaign Synchronization Component.

Deleting the   3. Delete the Genesys tables from the Siebel Server database.
Genesys Tables

> **Note:**   Deleting Genesys tables from the Siebel repository does not delete
> them from the Siebel database. You must delete them manually.
> The Genesys Campaign Synchronization2 project contains the
> following tables: CX_GCAS_ASSIGN, CX_GCAS_CFG_OBJ,
> CX_GCAS_QUEUE_ITM, and CX_GCAS_SUMMARY. You can use
> any database tool (such as, Microsoft SQL Enterprise Manager or
> Oracle Enterprise Manager) to delete these tables.

End of procedure

Next Steps

*   Delete the Campaign Synchronization Component-specific List of Values
    from the Siebel Server. See, Procedure: Uninstallation: Deleting the
    Campaign Synchronization Component-specific values from the Siebel
    Server.

Procedure:
Uninstallation: Deleting the Campaign Synchronization Component-specific values from the Siebel Server

Purpose: To delete Campaign Synchronization Component-specific List of
Values from the Siebel Server.

Start of procedure

1. Delete the values of the following types from the List of Values:
   - GENESYS_SYNCH_ORDER
   - GENESYS_SYNCH_MODE
   - GENESYS_SBL_SYNCH_STATUS

End of procedure

Next Steps

- Remove the responsibilities and the view definitions for the Campaign Synchronization Component. See the section, Procedure: Removing the Responsibilities and the View Definitions for the Campaign Synchronization Component.

Removing the Responsibilities and the View Definitions for the Campaign Synchronization Component

This section describes how to remove the responsibilities and the view definitions for the Campaign Synchronization Component.

Procedure: Uninstallation: Removing the responsibilities and the view definitions for the Campaign Synchronization Component

Purpose: To remove the responsibilities and the view definitions f for the Campaign Synchronization Component.

Start of procedure

1. Log in as a Siebel administrator.

2. Select Site Map > Administration > Application, Responsibilities and delete the following record identified by their name:

   Genesys CampSynch Administrator

3. Select Site Map > Administration > Application, Views and delete the following records identified by their name:
   - Genesys CaS Config View
   - Genesys CaS Assign View
   - Genesys CaS Siebel Monitor View Status
   - Genesys CaS Siebel Monitor View Queue

- Genesys CaS Request Queue View
- Genesys CaS Siebel Monitor View Summary
- Genesys CaS Synch Summary View

End of procedure

Next Steps

- Uninstall the Campaign Synchronization Component on a Windows or a UNIX platform. See the section, "Uninstallation of the Campaign Synchronization Component on a Windows or a UNIX Platform".

Uninstallation of the Campaign Synchronization Component on a Windows or a UNIX Platform

Select one of the following topics depending on your environment.

For a Windows Environment

Complete the uninstallation program for the Campaign Synchronization Component by using the Add/Remove Programs in the Control Panel on the host where you installed the Campaign Synchronization Component.

For a UNIX Environment

Delete the Campaign Synchronization Component's files by removing the directory that was used as the destination directory during installation.

You have deleted the G*plus* Adapter for Siebel CRM Campaign Synchronization Component from your environment.

Uninstalling the Voice Component

This section describes how to uninstall the G*plus* Adapter for Siebel CRM Voice Component (Voice Component). Since there is a separate *.sif archive file for each Voice feature, the information about the individual features is inserted into the general discussion.

This section describes one uninstallation scenario: a full uninstallation of the Voice Component.

As in any installation or uninstallation procedure, back up all of the files and the configuration before starting.

The process of uninstalling the Voice Component involves four general procedures:

1. Deleting the Voice Component-specific values and settings from the Siebel application.

You use the Siebel Administration application to complete these related tasks.

2. Deleting the Voice Component drivers and the configurations from the Siebel application.

3. Deleting the Voice Component objects from the Siebel application using the Siebel Tools and compiling a new *.srf archive file.

Note: The Basic Voice Siebel objects from the GenComm.sif file may be shared among other Voice Components, so these objects should be removed only as the last step of a full uninstallation of the G*plus* Voice Adapter.

4. Deleting the files installed by the Voice Component installation.

Note: Before proceeding with Step 4 of the uninstallation of the Voice Component, stop the G*plus* Communication Server.

Step-by-step instructions for completing each process are provided below.

The Required Environment

The following environment is required in order to complete the uninstallation process:

• The Siebel CRM Server platform with the G*plus* Adapter 8.0 for Siebel CRM should be installed and deployed.

• You should have the Siebel Tools application running.

• You should have administrator rights for accessing the Siebel application.

• You should have sufficient administrator privileges when logging into the operating system.

Deleting the Voice Component-Specific Values and Settings from Siebel

Procedure:
Uninstallation: Deleting the Voice Component-specific values and settings from the Siebel application

Purpose: To delete the Voice Component-specific values and settings from the Siebel application.

Start of procedure

1. Before you delete these values, read "Configuring Siebel Using the Siebel Web Client" on page 212.

2. For the Voice Component features, delete the values of the following types from the List of Values (LOV):

    Basic Voice

    • AGENT_WORK_MODE

    • REASON_CODE

    Expert Contact

    • GENESYS_CALL_STATUS

    • GENESYS_LOCATION

    Outbound Campaign

    • GENESYS_CALL_RESULT

    • GENESYS_TREATMENT_TYPE

    • GENESYS_CALLBACK_TYPE

    Voice Callback

    • GENESYS_ROUTING_POINT

End of procedure

Next Steps

• Delete the Voice Component drivers and configurations from the Siebel application. See section "Deleting the Voice Component-Specific Values and Settings from Siebel".

Deleting the Voice Component Drivers and Configurations from Siebel

This section describes how to delete the drivers and configuration from the Siebel application.

Note: This procedure is optional.

Procedure:
Uninstallation: Deleting the Voice Component drivers and configurations from the Siebel application

Purpose: To delete the Voice Component drivers and configurations from the Siebel application.

Start of procedure

1. Log into Siebel as an administrator.

2. In the Siebel Site Map, navigate to Communications Drivers and Profiles.
    - Select Site Map > Administration - Communications> Communications Drivers and Profiles.

3. Navigate to the G*plus* driver record and delete it.

4. In the Siebel Site Map, navigate to All Configurations.

5. Select Site Map > Administration - Communications> All Configurations.

6. Navigate to any unused profiles associated with the G*plus* driver and delete them.

End of procedure

Next Steps

- Delete the Siebel components and objects from Siebel. See the section, "Deleting the Voice Components Objects from Siebel".

Deleting the Voice Components Objects from Siebel

The following elements of the configuration are affected during the uninstallation process:

Basic Voice   Refer also to the GenComm.sif archive file for details.

Bitmap Category HTML Command Icons

- Bitmap

Business Service Genesys Voice

- Business Service Server Scripts

Command

- "HTML Comm *"

Toolbar Communication

- Toolbar Items

Expert Contact   Refer also to the GenComm_ECS.sif file for details.

Command

- HTML Comm *KW*

Toolbar Communication

- Toolbar Items

Universal Callback   Refer also to the GenComm_VCB.sif file for details.

Contact Activity List Applet

- Server script
- Applet Web Template
- Control

Business Component Action

- BusComp Server Scripts
- Field

Business Component Contact

- BusComp Server Scripts
- Field

Pick List

- PickList Genesys Call Result
- PickList Genesys Rout Point

Command

- HTML Comm VoiceCallback*

Toolbar Communication

- Toolbar Items

Table S_EVT_ACT_X

- Fields

Outbound Campaign

Refer also to the GenComm_OCS.sif file for details.

Campaign Contact/Prospect Form Applet

- Server script
- Applet Web Template
- Control

Persistent Customer Dashboard Applet

- Control

Business Component Campaign List Contact

- BusComp Server Scripts
- Field
- Single Value Field

Business Service Persistent Customer Dashboard

- Business Service User Prop

Command

- "HTML Comm *OB*"

Pick List

- PickList Genesys Call Result

Toolbar Communication

- Toolbar Items

Toolbar Dashboard

Using Siebel Tools to Change the Siebel Configuration for the Voice Component

You will use Siebel Tools to compile an updated release of the Siebel repository file for one or more of the Siebel applications you use on your Siebel Server, which you will then deploy on the server.

For more information about using Siebel Tools, see the Siebel documentation.

If the components and objects above are already checked out and reside in the local database, you will need to lock each one. See the topic, "Configuring Siebel" on page 196 for instructions on how to check out and lock projects, if it is necessary.

Next Steps

- Check out and lock the existing Voice Component projects. See, Procedure: Uninstallation: Checking out and locking existing Voice Component projects.

Procedure: Uninstallation: Checking out and locking existing Voice Component projects

Purpose: To check out and lock the existing Voice Component projects from the Siebel repository.

Start of procedure

1. In Siebel Tools, select Tools > Check Out…

   The Check Out dialog box displays.

2. In the Projects list, select the projects to be locked as selected from Table 90 on page 593.

3. Click Check Out.

Table 90: Projects for All Supported Siebel Versions - Common for All Components and Feature-Specific Projects

Feature	Archive File	Common Projects (for all components)	Feature-Specific Projects
Basic Voice	GenComm.sif	• Command • Communication • Communication Administration • Persistent Customer Dashboard	None
Expert Contact	GenComm_ECS.sif		None
Outbound Campaign	GenComm_OCS.sif		• Campaign
Universal Callback	GenComm_VCB		• Activity • Contact • Contact (SSE) • Table Activity • Genesys Voice • \<Project related to an application on which the Adapter is deployed with the Universal Callback feature >*

* This project is locked when deploying the Universal Callback feature as described in "Overview of the Siebel Repository File Preparation" on page 197. This is the project related to an application on which the Adapter is deployed—for example, Siebel Financial Services.

The installation package for the G*plus* Adapter 7.5 for Siebel CRM Voice Component contains the toolbar buttons, the command descriptions, and the bitmap images for the Genesys Chat, Genesys E-mail, and Siebel eMail components. Only the elements related to uninstalling the components should be deleted. Delete all of the elements for a full uninstallation. See the instructions under "Uninstallation: Changing the Expert Contact Feature items" on page 599 and "Uninstallation: Changing the Bitmap Category HTML Command Icons records" on page 594 for the distribution of the elements between the components.

End of procedure

Next Steps

• Delete the G*plus* customizations from the Siebel repository. See the section, "Deleting the Gplus Customizations from the Siebel Repository" on page 594.

Deleting the G*plus* Customizations from the Siebel Repository

This section describes how to delete the modifications made to the following records and services located in the Siebel repository:

*   Bitmap Category HTML Command Icons records
*   Command object records
*   Genesys Voice Business Service

Procedure:
Uninstallation: Changing the Bitmap Category HTML Command Icons records

Purpose: To delete the modifications made during deployment to the Bitmap Category HTML Command Icons records.

Start of procedure

1. In Siebel Tools, from the Object Explorer, select the Bitmap Objects folder.

    The Bitmap Category window displays.

2. In the Bitmap Category window, select the Bitmap Category HTML Command Icons folder.

    The Bitmaps Objects window appears.

3. Delete the following records in the Bitmaps Objects window:

    a. Basic Voice Feature
    *   Comm Do Not Disturb
    *   Comm Do Not Disturb On
    *   Comm Ready
    *   Comm Ready Disabled

    b. Expert Contact Feature
    *   Comm Accept Callback Enabled
    *   Comm Accept Callback Disabled
    *   Comm Reject Callback Enabled
    *   Comm Reject Callback Disabled

    c. Outbound Campaign Feature
    *   Comm OB Add Disabled
    *   Comm OB Add Enabled
    *   Comm OB Cancel Disabled
    *   Comm OB Cancel Enabled
    *   Comm OB Chained Record Disable

- Comm OB Chained Record Enabled
- Comm OB Do Not Call
- Comm OB Do Not Call Disabled
- Comm OB Get Preview Record
- Comm OB Get Preview Record Dis
- Comm OB Record Disabled
- Comm OB Record Enabled
- Comm OB Record Processed Dis
- Comm OB Record Processed Ena
- Comm OB Record Reject Disabled
- Comm OB Record Reject Enabled
- Comm OB Schedule Disabled
- Comm OB Schedule Enabled
- Comm OB Start Preview Disabled
- Comm OB Start Preview Enabled
- Comm OB Stop Preview Disabled
- Comm OB Stop Preview Enabled
- Comm OB Update Disabled
- Comm OB Update Enabled

d. Voice Callback Feature
- Comm Callback Done Disabled*
- Comm OB Cancel Enabled*
- Comm OB Cancel Disabled*
- Comm OB Add Enabled*
- Comm OB Add Disabled*
- Comm OB Record Processed Ena*
- Comm OB Record Processed Dis*
- Comm OB Record Enabled*
- Comm OB Record Disabled*
- Comm OB Get Preview Record*
- Comm OB Get Preview Record Dis*
- Comm OB Record Reject Enabled*
- Comm OB Record Reject Disabled*
- Comm OB Schedule Enabled*
- Comm OB Schedule Disabled*
- Comm VCB Start Session Enabled
- Comm VCB Start Session Disable

* - Shared with other configurations.

e. Genesys Chat Feature
- Comm GChat Not Ready
- Comm GChat Ready

   f. Genesys E-mail Feature
   - Comm GEmail Not Ready
   - Comm GEmail Ready

End of procedure

Next Steps

- Delete the G*plus* customizations made to the Command object records in the Siebel repository. See, Procedure: Uninstallation: Changing the Command object records.

Procedure:
Uninstallation: Changing the Command object records

Purpose: To delete the modifications that were made to the Command object records located in the Siebel repository.

Start of procedure

1. In Siebel Tools, from the Object Explorer, select the Command Objects folder.

   The Commands window displays.

2. Delete the following records in the Commands window:
   a. Basic Voice Feature
   - HTML Comm AcceptGVoice
   - HTML Comm Alternate Call
   - HTML Comm Cancel Monitoring
   - HTML Comm Delete From Conf
   - HTML Comm Do Not Disturb
   - HTML Comm InitiateGVoice
   - HTML Comm LoginToGVoice
   - HTML Comm LogoutFromGVoice
   - HTML Comm Monitor Next Call
   - HTML Comm NotReadyGVoice
   - HTML Comm Query Address
   - HTML Comm ReadyGVoice
   - HTML Comm Send DTMF
   b. Outbound Campaign Feature
   - HTML Comm OB Add
   - HTML Comm OB Chain Request
   - HTML Comm OB Do Not Call

- HTML Comm OB Preview Over
- HTML Comm OB Preview Start
- HTML Comm OB Record Cancel
- HTML Comm OB Record Processed
- HTML Comm OB Record Reject
- HTML Comm OB Record Request
- HTML Comm OB Record Reschedule
- HTML Comm OB Scheduled Resched
- HTML Comm OB Update
- HTML Comm OB Request Logout

c. Voice Callback Feature
- HTML Comm VoiceCallBackCancel
- HTML Comm VoiceCallbackAdd
- HTML Comm VoiceCallbackDone
- HTML Comm VoiceCallbackQuery
- HTML Comm VoiceCallbackRecord
- HTML Comm VoiceCallbackReject
- HTML Comm VoiceCallbackResched
- HTML Comm VoiceCallbackStart

d. Genesys Chat Feature
- HTML Comm AcceptGChat
- HTML Comm LoginToGChat
- HTML Comm LogoutFromGChat
- HTML Comm NotReadyGChat
- HTML Comm ReadyGChat

e. Genesys E-mail Feature
- HTML Comm AcceptGEmail
- HTML Comm InitiateGEmail
- HTML Comm LoginToGEmail
- HTML Comm LogoutFromGEmail
- HTML Comm NotReadyGEmail
- HTML Comm ReadyGEmail

f. Siebel eMail Feature
- HTML Comm AcceptSEmail
- HTML Comm InitiateSEmail
- HTML Comm LoginToSEmail
- HTML Comm LogoutFromSEmail
- HTML Comm NotReadySEmail
- HTML Comm ReadySEmail

g. Universal Interaction Handling commands

- HTML Comm AddInteraction
- HTML Comm Cancel Interaction
- HTML Comm Interaction Done
- HTML Comm Reschedule Interact

End of procedure

Next Steps

- Delete the G*plus* customizations for the Genesys Voice Business Service. See, Procedure: Uninstallation: Changing the Genesys Voice Business Service.

Procedure: Uninstallation: Changing the Genesys Voice Business Service

Purpose: To delete modifications that were made to the Genesys Voice Business Service during deployment.

Start of procedure

1. In Siebel Tools, from the Object Explorer, select the Business Service Objects folder.

2. Select the Genesys Voice Business Service record from the list of Business Services.

3. Delete the record.

End of procedure

Next Steps

- Delete the Expert Contact Feature items. See the section, "Deleting the Gplus Adapter Customizations Made to the Expert Contact Feature Items".

Deleting the G*plus* Adapter Customizations Made to the Expert Contact Feature Items

This section describes how to remove the G*plus* Adapter customizations made to the following Expert Contact Feature items:

- Command object records
- Toolbar items

Procedure:
Uninstallation: Changing the Expert Contact Feature items

Purpose: To delete the modifications that were made to the Expert Contact Feature items during deployment.

Start of procedure

Command object records

1. In Siebel Tools, from the Object Explorer, select the Command objects folder.

    The Commands window displays.

2. Delete the following records in the Commands window:
    *   HTML Comm KwCallStatusResponse
    *   HTML Comm KwOnCall

Toolbar items

3. In Siebel Tools, from the Object Explorer, select the Toolbar objects folder.

    The Toolbars window opens.

4. Select Toolbar Communication in the Toolbars window.

5. From the Object Explorer, select the Toolbar Item folder.

6. Select the Kw On Call record.

7. Right-click the selected record and click delete.

End of procedure

Next Steps

*   Delete the changes made to the Outbound Campaign Feature items during deployment. See the section, "Deleting the Gplus Adapter Customizations Made to the Outbound Campaign Feature Items" on .

Deleting the G*plus* Adapter Customizations Made to the Outbound Campaign Feature Items

This section describes how to delete the modifications made to the following Outbound Campaign Feature items:

*   Campaign Contact/Prospect Form applet
*   Campaign Contact List business component
*   Universal Callback Financial Services Siebel application
*   Contact Activity List applet

Procedure:
Uninstallation: Changing the Campaign
Contact/Prospect Form applet

Purpose: To remove the modifications that were made to the Campaign Contact/Prospect Form applet during deployment.

Start of procedure

1. In Siebel Tools, from the Object Explorer, select the Applet Objects folder.

2. Select the Campaign Contact/Prospect Form Applet on the list of applets.

3. Expand the Applet Objects folder in Object Explorer.

4. Next, follow the steps below to process these categories:
   - Server Script
   - Applet Web Template
   - Control

Server Script
   a. On the List of Applets window, right-click on the record and select Edit Server Scripts from the drop down menu.

   b. Navigate to the WebApplet_PreCanInvokeMethod.

   c. If this method contains only G*plus*-related modifications, then replace the complete source code between the braces with the statement:

   return (ContinueOperations) ;

   If this method contains user customizations, remove only the G*plus*-related part of the code.

   For more details, refer to the procedure, "Uninstallation: Obtaining the server script source codes" on page 612.

Applet Web Template
   d. From the Object Explorer, go to the Applet Web Template category.

   e. From the Applet Web Templates window, select the record Edit.

Control
   f. From the Object Explorer, go to and expand the Control category.

   g. Navigate to the Control window.

   h. Delete the following record:

   RescheduleTime

End of procedure

Next Steps

- Change the Campaign List Contact business component. See, Procedure: Uninstallation: Changing the Campaign List Contact Business Component, on page 581.

Procedure:
Uninstallation: Changing the Campaign List Contact business component

Purpose: To delete the modifications that were made to the Campaign List Contact business component during deployment.

Start of procedure

1. In Siebel Tools, from the Object Explorer, select and expand the Business Component Objects folder.

    The Business Components window displays.

2. Select the Campaign List Contact business component in the Business Components window.

3. Next, follow the steps below to process the following categories:
    *   BusComp Server Scripts
    *   Field
    *   Pick List
    *   PickList Genesys Call Result

BusComp Server Scripts

a. Select the Campaign List Contact business component in the Business Components window.

b. Right-click on the record and select Edit Server Scripts from the drop down menu.

c. On the Script editor window explorer, expand the (general) category.

d. Select the (declarations) section.

e. Delete following statements:
    *   var cStatusValue = null;
    *   var bInitiatedByCommCommand = false;

f. Under the BusComp category navigate to the BusComp_PreInvokeMethod.

g. If this method contains only G*plus*-related modifications, then replace the complete source code between the braces with the statement:

    return (ContinueOperations) ;

    If this method contains user customizations, remove only the G*plus*-related part of the code.

    For more details, refer to the topic, "The Universal Callback server script." on page 612.

h. Repeat Steps f and g for the following event handlers:
    *   BusComp_PreSetFieldValue
    *   BusComp_SetFieldValue
    *   BusComp_WriteRecord

i.   Close BusComp[Campaign list Contact] - Script window using the Tools menu command Window > Close.

j.   From the Object Explorer, navigate to the BusComp Server Scripts category.

k.   From the Business Component Server Scripts window, delete the following records:

-   CallCommunicationsCommand
-   DisplayGenesysError
-   GetGenesysStatusCode
-   SetDoNotCallFlag
-   PostChangesToGenesys
-   WriteRecordNoGenesysUpdate

l.   Save files using the File > Save All menu command.

Field   m.   From the Object Explorer, navigate to and expand the Field category.

n.   Select the Status record.

o.   On the Properties dialog box, set the field's PickList value PickList Genesys Call Result to blank.

Pick List   p.   From the Object Explorer, go to the Pick List objects.

q.   Process the following actions for PickList Genesys Call Result.

**PickList Genesys   r.   Select the PickList Genesys Call Result on the PickList window.
Call Result**   s.   Delete the record.

End of procedure

Next Steps

-   Delete the modifications made to the Universal Callback items during deployment. See the section, "Deleting the Gplus Customizations Made to the Universal Callback Items" on .

Deleting the G*plus* Customizations Made to the Universal Callback Items

This section describes how to delete the modifications made to the following Universal Callback items during deployment:

-   Siebel Financial Services application
-   Contact Activity List applet

Procedure:
Uninstallation: Changing the Siebel Financial Services application

Purpose: To delete the modifications made to the Siebel Financial Services application during deployment.

Start of procedure

1. In Siebel Tools, from the Object Explorer, go to the Applications objects.

   The Applications list window displays.

2. On the Applications list window, select the application on which the G*plus* Adapter was deployed (for example—Siebel Financial Services), then right-click the record and select Edit Server Scripts from the drop- down menu.

3. Under the Application category, navigate to the Application_PreNavigate method.

4. If this method contains only G*plus*-related modifications, then replace the complete source code between the braces with the statement:

   return (ContinueOperations) ;

   If this method contains user customizations, remove only the G*plus*-related part of the code.

   For more details, refer to the topic, "Uninstalling the UCS Gateway Component" on page 613.

End of procedure

Next Steps

- To delete the modifications made to the Contact Activity List applet during deployment. See, Procedure: Uninstallation: Changing the Contact Activity List applet, on page 603.

Procedure:
Uninstallation: Changing the Contact Activity List applet

Purpose: To delete the modifications made to the Contact Activity List applet.

Start of procedure

1. In Siebel Tools, from the Object Explorer, select the Applet Object folder.

2. Select the Contact Activity List applet on the list of applets.

3. Expand the Applet Object folder in Object Explorer.

4. Next, follow the steps below to process the following categories:
   • Applet Web Template
   • List
   • Business Component Action
   • Pick List
   • PickList Genesys Routing Point
   • Command

Applet Web Template

a. From the Object Explorer, navigate to the Applet Web Template category.

b. From the Applet Web Templates window, select the Base record.

c. From the Object Explorer, expand the Applet Web Template Item.

d. From the Applet Web Template Items window, delete the following records:
   • Genesys RoutPoint

e. From the Applet Web Templates screen, select the Edit List record.

f. Repeat Steps c and d.

List

g. From the Object Explorer, navigate to and expand the List category.

h. Select then List Column item

i. Navigate to the List Columns window.

j. Delete the following record:
   • Genesys RoutPoint

Business Component Action

k. From the Object Explorer, navigate to and expand the Business Component objects.

l. Select the Business Component Action in the Business Components window.

m. Process the following actions for:
   • Field

n. From the Object Explorer, navigate to and expand the Field category. The Field window displays.

o. From the Field window delete the following records:
   • Genesys Call Id
   • Genesys Route Point

Pick List

p. From the Object Explorer, navigate to the Pick List Objects folder.

q. Process the following actions for the PickList Genesys Rout Point record.

r. Select the PickList Genesys Rout Point on the PickLists window.

s. Delete the record.

Command   **t.** From the Object Explorer, navigate to the Command Objects folder.

The Command window displays.

u. Delete the following records in the Command window:

- HTML Comm VoiceCallbackAdd
- HTML Comm VoiceCallbackDone
- HTML Comm VoiceCallbackQuery
- HTML Comm VoiceCallbackRecord
- HTML Comm VoiceCallbackReject
- HTML Comm VoiceCallbackResched
- HTML Comm VoiceCallbackStart

End of procedure

Next Steps

- To restore the S_EVT_ACT_X table structure in the local Siebel database. See the section, "Restoring the S_EVT_ACT_X Table Structure in the Local Siebel Database or in the Siebel Server Database" on .

Restoring the S_EVT_ACT_X Table Structure in the Local Siebel Database or in the Siebel Server Database

This section describes how to update and deploy the S_EVT_ACT_X table changes located in the local Siebel database.

Note: This operation is required only if new tables or new fields were created—for example, changes made to the GenComm_VCB.sif file.

Procedure: Uninstallation: Changing the S_EVT_ACT_X Table Structure

Purpose: To update the S_EVT_ACT_X table in the local Siebel database (or in the Siebel server database).

Start of procedure

1. Connect Siebel Tools to the server database.

2. In Siebel Tools, from the Object Explorer, navigate and expand the Table Object folder.

3. Navigate to the List of Tables window.

4. Locate the S_EVT_ACT_X table.

5. In the Object Explorer window, select the Columns category.

6. Locate and remove the following records:
  - X_ATTRIB_51
  - X_ATTRIB_52

7. Click Apply.

A pop-up warning window appears.

8. Click OK to accept the warning.

9. In the new Apply Schema window, from the Tables drop down list, select Current Row.

10. Enter the correct values for the Database user, the Database user password, and the ODBC data source.

You must enter the user name and password for a Database user who has an administrator's privileges in the Siebel environment.

See the Siebel documentation for more information about creating custom tables.

11. Click Apply.

The message "Changes successfully applied" appears, indicating that the tables were successfully created.

12. Click the Activate button to propagate the database changes and make them available to all users.

Note: It may be necessary to create new fields in the local database. Refer to the Siebel Tools documentation for more information.

End of procedure

Next Steps

- Update the S_EVT_ACT_X table changes in the Siebel server repository. See, Procedure: Uninstallation: Updating the S_EVT_ACT_X table changes in the Siebel Server repository, on page 606.

Procedure:
Uninstallation: Updating the S_EVT_ACT_X table changes in the Siebel Server repository

Purpose: To update the S_EVT_ACT_X table changes in the Siebel Server repository.

Start of procedure

1. In Siebel Tools, select Tools > Check In....

   The Check In dialog box displays.

2. In the Projects list, select the following projects:
   - Table Activity.

3. Click Check In.

   To apply changes to Siebel databases, follow the steps described in the section "Updating the S_EVT_ACT_X Table in the Genesys Project Server Database" on page 209.

End of procedure

Next Steps

- Delete the modifications that were made to the Communication Toolbar during deployment. See, Procedure: Uninstallation: Changing the Communication Toolbar items.

Deleting the G*plus* Customizations Made to the Communications Toolbar

This section describes how to delete the modifications made to the Communication Toolbar during deployment.

Note: Some of the Communication Toolbar buttons are deleted during the component uninstallation procedures. The scenario described below is provided to present an overview of the Communication Toolbar uninstallation procedures and to remove the Universal Interaction Control buttons, which are used with all of the Voice Components.

Procedure: Uninstallation: Changing the Communication Toolbar items

Purpose: To delete the modifications that were made to the Communication Toolbar items during deployment.

Start of procedure

1. In Siebel Tools, from the Object Explorer, navigate to and expand the Toolbar Object folder.

   The Toolbar window displays.

2. Select the Communication Toolbar folder in the Toolbar window.

3. Process the following actions for the Communication Toolbar items.

    a. From the Object Explorer, navigate to the Toolbar Item category.

    b. Delete the records as shown in Table 91 on .

Table 91: Toolbar Items

Feature	Records
Basic Voice Feature	• Start Voice Callback • Alternate Call • Cancel Monitoring • Delete From Conference • Do Not Disturb • LogIn GVoice • LogOut GVoice • Monitor Next Call • Query Address • Ready GVoice
Expert Contact Feature	• KW on Call
Outbound Campaign Feature	• Chained Record Request • Do Not Call • Preview Dialing Mode Over • Preview Dialing Mode Start • Preview Record Request
Genesys Chat Feature	• Accept GChat • LogIn GChat • LogOut GChat • Not Ready GChat • Ready GChat
Genesys Email Feature	• Accept GEmail • Initiate GEmail • LogIn GEmail • LogOut GEmail • Not Ready GEmail • Ready GEmail
Siebel Email Feature	• Accept Email • Initiate Email • LogIn SEmail • LogOut SEmail • Not Ready Email • Ready SEmail

Table 91: Toolbar Items (Continued)

Feature	Records
Universal Interaction Control buttons	• Add Interaction • Reject Interaction • Cancel Interaction • Reschedule Interaction • Interaction Done

End of procedure

Next Steps

- Compile the Siebel repository file. See, Procedure: Uninstallation: Compiling the Siebel repository file, on page 610.

Procedure:
Uninstallation: Compiling the Siebel repository file

Purpose: To compile the Siebel repository file.

Start of procedure

1. In Siebel Tools, select Tools > Compile Projects...
2. Select Locked projects.
3. In the Siebel Repository File edit box, enter the name of the repository file.
4. Click Compile.

    The status bar at the bottom of the Object Compiler window indicates when the compilation is finished.
5. When the compilation is finished, close Siebel Tools.

Note: It is possible that the siebel_assert_xxx.txt file(s) is generated by the Siebel environment after compiling the Siebel repository file.

End of procedure

Next Steps

- Delete the files installed by the Voice Component installation. See the section, "Deleting the Files Installed by the Voice Component Installation".

Deleting the Files Installed by the Voice Component Installation

This section describes how to delete the files that were installed during deployment.

Before you start uninstalling the files, read the section, "Installation" on . The information from this topic will help you successfully complete the uninstallation process.

Procedure:
Uninstallation: Uninstallation of the Voice Component on the current platform

Purpose: To uninstall the Voice Component on the current platform.

Start of procedure

For a Windows Environment
1. Log into the system as an Administrator.
2. Stop the G*plus* Communication Server.
3. Double-click setup.exe from the installation package (windows\setup.exe).
4. When prompted, "Genesys Installation Wizard lets you remove the current installation of the G*plus* Adapter for Siebel CRM Voice from your computer", choose Remove and click Next.
5. To complete uninstallation, remove the icons from the web server images directory.

For a UNIX Environment
6. Log into the system as an Administrator.
7. Stop the G*plus* Communication Server.
8. Go to the G*plus* Communication Server directory.
9. Manually remove the following G*plus* Adapter files:
   - <Gplus Communication Server directory>/libGenComm.so
   - <Gplus Communication Server directory>/libGenModel.so
10. Remove all of the files and subdirectories in the <Destination Directory> folder, which is the destination directory for installation that is used by the installation script to copy the G*plus* Adapter for Siebel CRM Voice files).

11. To complete uninstallation, remove the icons from the web server images directory.

Note: Make sure to restore the original Siebel icons for the Ready/Not Ready Communication toolbar buttons from the backup files. See the Installation section of Chapter 6, "Deploying the Voice Component," on page 175 for details.

End of procedure

Next Steps

*   Obtain the server script source codes. See the section, "Obtaining the Server Script Source Codes Included in the Gplus Adapter Installation Package".

Obtaining the Server Script Source Codes Included in the G*plus* Adapter Installation Package

You can obtain the source codes for the following server scripts:

*   The Campaign List Contact business component server script.
*   The Campaign Contact/Prospect Form applet server script.
*   The Universal Callback server script.

Procedure: Uninstallation: Obtaining the server script source codes

Purpose: To obtain the Campaign List Contact business component server script source code, the Campaign Contact/Prospect Form applet server script, and the Universal Callback server script.

Start of procedure

Campaign List Contact Business Component

1. Prepare a clean local Siebel database. Refer to the Siebel Tools documentation for detailed procedures.

2. Start the Siebel Tools application and set up to work with your local database.

3. Import the GenComm_OCS.sif archive file. For more information, see the topic, "Resolving Siebel Application Data Conflicts" on page 205.

4. From the Object Explorer, navigate to the Business Component objects.

5. In the Business component window, select the Campaign List Contact record.

6. Right-click the record and select Edit Server Script from the drop-down menu.

7. From the Siebel Tools main menu, select File > Export.

8. Save the source code of the script.

Campaign Contact/Prospect Form applet

9. From the Object Explorer, navigate to the Applet objects.

10. In the Applets window, select the Campaign Contact/Prospect Form Applet record.

11. Right-click the record and select Edit Server Script from the drop-down menu.

12. From the Siebel Tools main menu, select File > Export.

13. Save the script's source code.

Universal Callback

14. Navigate to the <InsDir>/Application_PreNavigate_VCB.es file.

    The source code for the Application_PreNavigate Event server script is located in this file.

Note: Refer to "Updating the Application's Application_PreNavigate Event Server Script" on in the Appendix.

End of procedure

Next Steps

*   There are no further steps.

Uninstalling the UCS Gateway Component

The section describes how to delete the files installed by the G*plus* Adapter for Siebel CRM UCS Gateway Component (UCS Gateway Component) in the installation directory during deployment.

Procedure:
Uninstallation: Deleting the files installed by the UCS Gateway Component

Purpose: To delete the files installed by the G*plus* Adapter for Siebel CRM UCS Gateway Component based on your platform.

Start of procedure

Windows 1. Complete the uninstallation program for the UCS Gateway Component by using Add/Remove Programs in the Control Panel on the host on which you installed the component.

UNIX 2. Delete the UCS Gateway Component's files by removing the directory that was used as the destination directory during installation.

End of procedure

Next Steps

- There are no further steps.

Uninstalling the Media Routing Component

This section describes how to uninstall the G*plus* Adapter for Siebel CRM Media Routing Component (Media Routing Component). The uninstallation process consists of three steps:

1. Backing up your files and configuration before starting the uninstallation process (recommended by Genesys).

2. Deleting the files installed in the installation directory.

3. Deleting the Siebel objects installed by the Media Routing Component during deployment.

The Required Environment

The following environment is required to complete the uninstallation process:

- The Siebel CRM Server platform with the G*plus* Adapter 8.0 for Siebel CRM should be installed and deployed.

- You should have the Siebel Tools application running.

- You should have administrator rights for accessing the Siebel application.

- You should have sufficient administrator privileges when logging into the operating system.

The Uninstallation Process

This section contains step-by-step instructions about how to uninstall the Media Routing Component from the Siebel Server and remove the unused components from the Siebel application.

Removing the Files Installed by the G*plus* Adapter for Siebel CRM Media Routing Component

Follow the instructions below based on your environment.

For Windows Customers

•   Complete the uninstallation program for the Media Routing Component by using Add/Remove Programs in the Control Panel on the host on which you installed the component.

For UNIX Customers

•   Delete the Media Routing Component's files by removing the directory that was used as the destination directory during installation.

Removing the Siebel Objects

Use the Siebel Tools application to remove the Siebel objects.

Procedure:
Uninstallation: Removing the Siebel objects (Media Routing Component)

Purpose: To remove the Siebel objects.

Start of procedure

1. Remove the GplusMediaRoute and GplusMediaRouteIXN Business Services.
    a. Lock the GplusMediaRoute and GplusMediaRouteIXN projects.
    b. From the Object Explorer, browse to the GplusMediaRoute and GplusMediaRouteIXN projects and select the Business Service tab.
    c. In the Business Services window, right-click the GplusMediaRoute and GplusMediaRouteIXN Business Services and select Delete from the drop-down menu.
2. Remove the MediaRouting Communication Detail View.
    a. From the Object Explorer, select the View tab.
    b. Click the New Query button on the toolbar.
    c. Set the Name to MediaRouting Communication Detail View, and click the Execute Query button.
    d. From the Views window, right-click the MediaRouting Communication Detail View record and select Delete from the drop-down menu.

3. Remove the MediaRouting Comm Inbound Item Form applet and MediaRouting Comm Outbound Item applet.

    a. From the Object Explorer, select the Applet tab.

    b. Click the New Query button on the toolbar.

    c. Set the Project to GplusMediaRouting, and click the Execute Query button.

    d. From the Applets window, right-click the MediaRouting Comm Inbound Item Form Applet record and select Delete from the drop-down menu.

    e. From the Applets window, right-click the MediaRouting Comm Outbound Item Applet record and select Delete from the drop-down menu.

4. Recompile the project.

Siebel Administration Application

5. Start the Siebel Server.

6. Deactivate and delete the GplusMediaRouting-ProcessMessage workflow process.

    a. Select Site Map > Business Process Administration > Workflow Processes.

    b. Make a query for the GplusMediaRouting-ProcessMessage workflow.

    c. Deactivate and delete all the GplusMediaRouting-ProcessMessage workflows.

    d. If you modified the vanilla Siebel workflow processes, undo those changes.

End of procedure

Next Steps

- There are no further steps.

Uninstalling the Multimedia Component

The process of uninstalling the G*plus* Adapter for Siebel CRM Multimedia Component (Multimedia Component) consists of the following steps:

- Remove the files installed in the installation directory.
- Remove the Siebel objects that were installed during the G*plus* Adapter for Siebel CRM Multimedia Component installation.
- Update the Siebel configuration.

The Uninstallation Process

This section contains step-by-step instructions about how to uninstall the G*plus* Adapter for Siebel CRM Multimedia Component from the Siebel Server and remove the unused components from the Siebel application.

Removing the Files Installed by the G*plus* Adapter for Siebel CRM Multimedia Component

Follow the instructions below based on your environment.

Windows
- Complete the uninstallation program for the Multimedia Component by using Add/Remove Programs in the Control Panel on the host on which you installed the component.

UNIX
- Delete the Multimedia Component's files by removing the directory that was used as the destination directory during installation.

Removing the Siebel Objects

This section describes how to remove the Siebel objects.

Procedure:
Uninstallation: Removing the Siebel objects (Multimedia Component)

Purpose: To remove the Siebel objects for the Multimedia Component.

Start of procedure

1. Start Siebel Tools.
2. Select the Genesys Multimedia project in the top drop-down combo box of the Object Explorer.

3. Go to the following folders and remove all of the objects belonging to the Genesys Multimedia project:
   - Applet
   - Business Component
   - Business Object
   - Business Service
   - Integration Object
   - Link
   - Screen
   - Symbolic Strings
   - View
   - Web Template

4. Remove the Page Tab record created during the deployment.

5. Compile a new *.srf archive file.

End of procedure

Next Steps

- Update the Siebel configuration. This procedure is optional. See, Procedure: Uninstallation: Updating the Siebel configuration for the Multimedia Component, on page 618.

Updating Siebel Configuration

This step is optional. This section describes how to remove the drivers and the configurations from the Siebel application.

Procedure:
Uninstallation: Updating the Siebel configuration for the Multimedia Component

Purpose: To remove the drivers and the configurations from the Siebel application.

Start of procedure

1. Log into Siebel as an administrator.

2. In the Siebel Site Map, navigate to Communications Drivers and Profiles:
   - Select Site Map > Administration -Communications > Communications Drivers and Profiles.

3. Navigate to the Gplus_OpenMedia record and delete it.

4. In the Siebel Site Map, navigate to All Configurations:
   - Select Site Map > Administration -Communications > All Configurations.

5. Navigate to any unused profiles associated with the Gplus_OpenMedia record and delete them.

End of procedure

Next Steps

- Remove the List of Values (LOV) associated with the Multimedia Component. See, Procedure: Uninstallation: Removing the List of Values associated with the Multimedia Component, on page 619.

Removing the List of Values (LOV) Associated with the Multimedia Component

This section describes how to remove the List of Values associated with the Multimedia Component.

Procedure:
Uninstallation: Removing the List of Values associated with the Multimedia Component

Purpose: To remove the List of Values associated with the Multimedia Component.

Start of procedure

1. Log into Siebel as an administrator.

2. In the Siebel Site Map, navigate to the List of Values administration applet and delete the following records identified by Type and Display value:
   - CHANNEL_TYPE, Gplus Universal
   - REPOSITORY_HTML_CTRL_TYPE, GChatControl
   - COMM_MEDIA_TYPE, Gplus Universal
   - TODO_TYPE, Email - Outbound (MCR)\
   - TODO_TYPE, Email - Inbound (MCR)
   - LOV_TYPE, MCR_WORKBIN_TYPE
   - MCR_WORKBIN_TYPE,*
   - EVENT_STATUS,Completed
   - EVENT_STATUS, NotQueued

End of procedure

Next Steps

Removing the Responsibilities and the View Definitions

This section describes how to remove the responsibilities and the view definitions for the Multimedia Component.

Procedure:
Uninstallation: Removing the responsibilities and the view definitions for the Multimedia Component

Purpose: To remove the responsibilities and the view definitions.

Start of procedure

1. Log into Siebel as an administrator.

2. In the Siebel Site Map, navigate to the Responsibilities administration view and delete the following record identified by Name:
    • Genesys Multimedia Agent

3. In the Siebel Site Map, navigate to the View administration view and delete the following records identified by Name:
    • MCR Chat plus Knowledge Manager
    • MCR Reply Email plus InAttachments
    • MCR Reply Email plus Knowledge Manager
    • MCR Reply Email plus OutAttachments
    • MCR Reply Email with Suggestion
    • MCR Reply Email plus InAttachments T
    • MCR Reply Email plus Knowledge Manager T
    • MCR Reply Email plus OutAttachments T
    • MCR Reply Email with Suggestion T
    • MCR Workbin View
    • MCR dummy 1
    • MCR dummy 2
    • MediaRouting Communication Detail View

End of procedure

Next Steps

- Remove the Outbound Web Service definition. See, on .

Removing the Outbound Web Service Definition

This section describes how to remove the Outbound Web Service definition.

Procedure: Uninstallation: Removing the Outbound web service definition

Purpose: To remove the Outbound Web Service definition.

Start of procedure

1. Log into Siebel as an administrator.
2. In the Siebel Site Map, navigate to the Outbound Web Service administration view and delete the following record identified by Name:
   - Genesys G*plus* WebService

End of procedure

Next Steps

- There are no further steps.

![Genesys logo]

Scripts

This appendix describes how to generate and deploy the browser and server scripts and includes the following sections:

Generating and Deploying Browser Scripts

You must generate and deploy the browser scripts each time a new Siebel repository (*.srf) file is deployed to the Mobile Web Client or to the Siebel Server.

These browser scripts are JavaScript files (*.js) that are automatically generated when you compile the objects to a repository file (this is for compiled projects only). You can specify where the files are to be compiled (see the option in Siebel Tools), but if you are deploying to a server environment, the files have to be manually migrated to a specific directory on the Web Server. This process has been replaced by a utility file that extracts the compiled browser script files from the repository file. This utility file is called genbscript.exe and is located in either the siebsrvr/bin or the client/bin.

The browser scripts on the Siebel Server machine are located in the following folder path: <SWEApp>/public/language_code.

Note: After you have created the browser scripts, you must re-start Siebel Server

For further information about browser script generation and deployment, see "Siebel Tools Reference" and "Alert 365: Browser Script Generation in Siebel v7" on the Siebel support website.

Updating the Application's Application_PreNavigate Event Server Script

During the G*plus* Adapter for Siebel CRM Voice Component or the G*plus* Adapter for Siebel CRM Multimedia Component configuration, you must add or modify, if it already exists, the server script handler for the Application_PreNavigate event for the Siebel application that is being used.

The content of the event handler for each component is provided in a special *.es file. The *.es for each component is listed as follows:

- For the Voice component, this file is:

    <InsDir>/Application_PreNavigate_VCB.es

- For the Multimedia component, this file is:

    <InsDir>/Application_PreNavigate.es

*.es File Structure Each of these files has the following structure:

```
function Application_PreNavigate (DestViewName, DestBusObjName)
{
    <function body>
    return (ContinueOperation);
}
```

where <function body> is the eScript code designed for the appropriate component.

If the original Application_PreNavigate event script consists only of a return(ContinueOperation) call, change the structure of the resulting eScript code as follows:

```
function Application_PreNavigate (DestViewName, DestBusObjName)
{
    <VOICE function body>
    <MULTIMEDIA function body>
    return (ContinueOperation);
}
```

Therefore, just copy the <function body> code from the appropriate files for each component into the Application_PreNavigate event script body and then finalize them by a return(ContinueOperation) call.

If the original Application_PreNavigate event script is more complex, make a *manual* merge.

![GENESYS]

Related Documentation Resources

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

G*plus* Adapter for Siebel CRM

- *Gplus* Adapter 8.0 for Siebel CRM *User's Guide*. Explains how to use the Adapter in your contact center environment. Introduces the G*plus* Adapter and highlights new features in the current release.

- *Gplus* Adapter 8.0 for Siebel CRM *Developer's Guide*. Describes the API (application programming interface) with which you can customize the export of campaigns, campaign contacts, and Do Not Call requests from Siebel to Genesys software.

- Release Notes and Product Advisories for this product, which are available on the Genesys Customer Care website at http://genesys.com/customer-care.

Genesys

- *Genesys Technical Publications Glossary,* available on the Genesys Documentation website, 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 Guide*

Consult the following additional resources as necessary:

- *Genesys Hardware Sizing Guide,* which provides information about Genesys hardware sizing guidelines for the Genesys 8.x releases.
- *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 Licensing Guide,* which introduces you to the concepts, terminology, and procedures that are relevant to the Genesys licensing system.
- *Genesys Database Sizing Estimator 8.x 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.genesyslab.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, 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:

80fr_ref_06-2008_v8.0.001.00

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

Table 92: 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 628).	Please consult the *Genesys Migration Guide* for more information. Do *not* use this value for this option. A *customary and usual* practice is one that is widely accepted and used within a particular industry or profession. The formula, $x + 1 = 7$ where x stands for . . .
Monospace font (Looks like teletype or typewriter text)	All programming identifiers and GUI elements. This convention includes: • The *names* of directories, files, folders, configuration objects, paths, scripts, dialog boxes, options, fields, text and list boxes, operational modes, all buttons (including radio buttons), check boxes, commands, tabs, CTI events, and error messages. • The values of options. • Logical arguments and command syntax. • Code samples. Also used for any text that users must manually enter during a configuration or installation procedure, or on a command line.	Select the Show variables on screen check box. In the Operand text box, enter your formula. Click OK to exit the Properties dialog box. T-Server distributes the error messages in EventError events. If you select true for the inbound-bsns-calls option, all established inbound calls on a local agent are considered business calls. Enter exit on the command line.
Square brackets ([ ])	A particular parameter or value that is optional within a logical argument, a command, or some programming syntax. That is, the presence of the parameter or value is not required to resolve the argument, command, or block of code. The user decides whether to include this optional information.	smcp_server -host [/flags]
Angle brackets (< >)	A placeholder for a value that the user must specify. This might be a DN or a port number specific to your enterprise. **Note:** In some cases, angle brackets are required characters in code syntax (for example, in XML schemas). In these cases, italic text is used for placeholder values.	smcp_server -host <confighost>

Index

Symbols

A

B

C