Genesys®

UC Connector 8.0

Deployment Guide

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.genesyslab.com` for more information.

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

Notice

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

Your Responsibility for Your System's Security

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

Trademarks

The Crystal monospace font is used by permission of Software Renovation Corporation, `www.SoftwareRenovation.com`.

Technical Support from VARs

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

Technical Support from Genesys

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

Ordering and Licensing Information

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

Released by

Genesys Telecommunications Laboratories, Inc. `www.genesyslab.com`

Document Version: 80ucc_dep_06-2013_v.8.0.301.00

Table of Contents

Table of Contents

UC Connector 8.0

List of Procedures

Genesys

Preface

Welcome to the *UC Connector 8.0 Deployment Guide.* This document provides an overview of the deployment of the UC Connector into the Genesys contact center, as well as the integration of the UC Connector deployment with the third-party UC platform in the Enterprise.

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

This preface 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 223.

Intended Audience

This guide is intended primarily for system engineers and other members of an implementation team who will complete the deployment and integration of the UC Connector into the Genesys environment and with the third-party UC solution. This guide assumes that you have a basic understanding of:

- Computer-telephony integration (CTI) concepts, processes, terminology, and applications.
- Unified Communications (UC) generally, as well as the specifics of the third-party UC platform deployed on the Enterprise side.
- The Genesys Management Framework architecture and functions that support T-Server, SIP Server, and Genesys routing.
- The Session Initiation Protocol (SIP).
- Network design and operation.

- Your own network configurations.

Making Comments on This Document

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

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

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

Contacting Genesys Technical Support

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

Before contacting technical support, please refer to the *Genesys Care Program Guide* 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 8.0.301.00

This section provides details about what is new or has changed significantly from version 8.0.201.00 of this document.

Table 1: Document Changes

Heading/Topic	Page	Details
Chapter 2, "How It Works," on page 31		
How It Works—Customized Knowledge Worker States	49	A new section describes how UC Connector allows you to customize presence states for Knowledge Workers.

Table 1: Document Changes (Continued)

Heading/Topic	Page	Details
How It Works—External Number Redirect	52	A new section describes the functionality of the external number redirect feature.
Chapter 5, "Deploying UC Connector," on page 75		
Customizing Knowledge Worker States	155	A new section describes how to configure customized presence states for Knowledge Workers.
Enabling a Redirect Number	158	A new section describes how to enable the redirect number feature.
Enabling After Call Work	158	A new section describes the configuration steps to enable After Call Work.
Appendix A, "Configuration Options," on page 189		
UC-Connector Section	189	New configuration options are added to the UC-Connector section: • gla-call-match-window • gla-kpl-time • gla-kpl-response-time • presence-location • redirect-setup-enabled
Microsoft-OCS Section	199	New configuration options are added to the Microsoft-OCS section: • presence-acw-note • presence-acw-status • presence-lg-note • presence-lg-status
Knowledge Worker Person Object Section	207	New configuration options are added to the UC-Connector section: • redirect-setup-enabled • redirect-enabled • redirect-number

New in Document Version 8.0.201.00

This section provides details about what is new or has changed significantly from version 8.0.101.00 of this document.

Table 2: Document Changes

Heading/Topic	Page	Details
Chapter 1, "About the UC Connector," on page 19		
Release 8.0.200	20	A new section identifies the features that were added in the 8.0.200.00 release of UC Connector.
Deployment Modes	30	A new section identifies the three deployment modes available in UC Connector.
Chapter 2, "How It Works," on page 31		
How It Works—Voice Scenarios	31	New graphics illustrate updated call flows.
Table 4: Status Change Scenarios	37	A new table row explains the Automatic changes in UC Connector client action.
The Overall Preview Interaction	40	You can now specify an audio file to be played if the audio-on-preview option is configured.
How It Works—Reporting Events	42	A new section describes how UC Connector produces and sends reporting-related events for the Interaction Preview mechanism to reporting platforms.
Chapter 4, "Deployment Prerequisites," on page 67		
Table 9: Configuring the Baseline Genesys Environment	68	A new step explains how to configure UC Connector in Genesys Administrator.
Chapter 5, "Deploying UC Connector," on page 75		
Task Summary: Deploying the UC Connector	76	New rows explain how to enable the following: • Automatic Login • Logout Menu • Audio on Preview or Ringing

Table 2: Document Changes (Continued)

Heading/Topic	Page	Details
Table 11: UC Connector—UC Connector Section	84	New rows identify new options in the UC-Connector section of the UC Connector Application object. • presence-gateway-mode • preview-shortkey-accept • preview-shortkey-reject • login-queue • enable-logout-menu • enable-preview-reporting • preview-state-name • presence-gateway-mode • popup-udata-key
Table 13: Interoperability Values for OCS Presence States	90	New values indicate OCS Presence State.
Task Summary: Configuring the Knowledge Worker	91	New procedures explain how to create the Knowledge Worker in Genesys Administrator: • Procedure: Creating the Knowledge Worker Place in Genesys Administrator. • Procedure: Creating the Knowledge Worker Person in Genesys Administrator
Task Summary: Enabling MTLS Communication	131	New steps and procedures explain how to enable MTLS Communication by generating client and server certificates.
Enabling Audio on Preview or Ringing	153	A new section explains how to enable an audio file to play when a Preview or Ringing pop-up window is displayed.
Configuring Hotkeys for Interaction Preview	154	A new section describes the steps to configure hotkeys to control accepting or rejecting a call when the Preview window is displayed.
Appendix A, "Configuration Options," on page 189		

Table 2: Document Changes (Continued)

Heading/Topic	Page	Details
UC-Connector Section	189	New configuration options are added to the UC-Connector section: • audio-on-preview • audio-on-ring • login-queue • enable-logout-menu • presence-gateway-mode • presence-gateway-mode • preview-state-name • preview-shortkey-accept • preview-shortkey-reject • enable-preview-reporting • popup-udata-key The default value of user-unregister-timeout is updated from 300000 to 60000.
Microsoft-OCS Section	199	The default value of agent-status-ready is updated from 3000-4499 to 0-4499. The default value of agent-status-logout is updated from 0-2888,18000- to 18000-.
Appendix C, "T-Server Compatibility with UC Connector," on page 213		
Table 30: UC Connector Compatibility with T-Server	213	A new table describes UC Connector's compatibility with each T-Server.

![Genesys]

1

UC Connector Overview

For an overview of the UC Connector and how it connects the contact center to the enterprise, as well as the kinds of deployments and call flow scenarios that the UC Connector supports, see the following chapters:

- Chapter 1, "About the UC Connector," on page 19
- Chapter 2, "How It Works," on page 31
- Chapter 3, "Supported Integrations," on page 57

1

About the UC Connector

The UC Connector 8.0 enables enterprise-wide customer service. It connects the Genesys Contact Center to the larger enterprise, allowing agents to consult the enterprise Knowledge Workers and take advantage of their special or expert skills. You can also use it independently from a Genesys Contact Center to enable knowledge workers to flexibly provide customer service in a more personal setting or when a contact center is not part of the company function.

This chapter includes the following sections:

What Is the UC Connector?

As a Genesys server application, you deploy UC Connector 8.0 in the Genesys environment. With UC Connector, Genesys routing capability and business rules that normally govern the flow of communication within the contact center are extended to control the flow of interactions into the back office or branch office, as well as to provide end-to-end reporting on these interactions. The UC Connector adds unique capabilities that make it easier to involve knowledge workers in customer interactions without impacting their normal workflow, providing enterprise-wide customer service. In addition, the UC Connector is used as an integral part of Genesys integration with Microsoft Lync Enterprise Voice, which is a contact center offering. In this setting, UC Connector acts as a presence gateway with Microsoft Lync server, propagating

presence information from Lync to Genesys and vice versa, and acting in concert with the Genesys SIP Server to provide call control services to users.

New In This Release

Release 8.0.300 The following new features were introduced in release 8.0.300.00 of UC Connector.

- **Customized Knowledge Worker states.** UC Connector now allows you to customize the states available to Knowledge Workers in the UC Connector web client drop-down menu. The Knowledge Worker states and the corresponding text displayed in the menu can be customized by editing application resources. See "How It Works—Customized Knowledge Worker States" on .

- **External number redirect.** A user or an Administrator can now enable an external redirect number. Enabling this feature allows agents to accept preview calls at the specified number. See "How It Works—External Number Redirect" on .

- **Propagation of Genesys After Call Work state to Lync, in integration with Microsoft Lync Enterprise Voice.** When an agent enters the After Call Work state, the agent's presence state is:
    - Preserved in Genesys until the agent uses the Lync client menu to change state, or the After Call Work timer expires.
    - Propagated to the Lync server so that the agent's unavailability is also reflected in the corresponding Lync presence, with a configurable presence status and note values.

  When the agent exits the After Call Work state (either automatically or manually), the agent's Lync presence state is set back to a value that is preserved from the Lync presence update. The agent's Genesys state is also updated with the corresponding value. See "Enabling After Call Work" on .

Release 8.0.200 The following new features were introduced in release 8.0.200.00 of UC Connector.

- **Interaction Preview-related reporting events.** UC Connector now creates reporting-related records in ICON for the user actions in the Preview window. Reporting tools can extract these records to create reports on the performance of Knowledge Workers while responding to previews. See "How It Works—Reporting Events" on .

- **Configurable hotkeys for interaction Preview.** You can now configure keyboard hotkeys to perform key actions when the Preview window is in focus. See "Configuring Hotkeys for Interaction Preview" on .

- **Play audio with interaction Preview.** You can now add a custom audio file that UC Connector plays when the Preview window is displayed. See "Enabling Audio on Preview or Ringing" on .

- **User logout ability.** Users can now explicitly logout of UC Connector by clicking a button in the user interface. See enable-logout-menu on page 191.
- **Default routing.** UC Connector now includes an application-wide option to allow default routing in case of URS failure, if this functionality is supported by the T-Server and other solution components. See login-queue on page 190.
- **Third-party call control window suppression.** UC Connector now suppresses web browser pop-ups for incoming calls based on User Data attached to the call. See popup-udata-key on page 198.
- **Username in browser window.** UC Connector now displays the username for the logged in user in the browser window. The displayed string shows "first name" + <space> + "last name" + "- Genesys", as specified in the Person object for the logged in user in Configuration Manager.
  - John Smith - Genesys

  If neither field is present, then the Person user ID will be used.
  - LyncKW9 - Genesys

Release 8.0.100   The following new features were introduced in release 8.0.100.00 of UC Connector.

- **Customized Help Button**. UC Connector now includes a Help button, which you can use to link to a customized help files located on your network or server. You can hide or show this help button as it appears in various parts of the user interface. See "How It Works—Customized Help" on page 46.
- **Customized Default Languages**. UC Connector now lets you select from a variety of supported languages to be used in the interface. See "How It Works—Customized Languages" on page 47.
- **Automated UC Connector Log-in**. UC Connector now supports automatic log-in for all users on startup. This feature must be enabled for integrations with Microsoft Lync Server 2010. See user-auto-registration on page 196.
- **Enhanced Support for Microsoft Lync Server 2010**. UC Connector now uses the Microsoft Lync extensibility window for integration with UC Connector. See "How It Works—Lync Integration" on page 44.
- **Countdown Timer**. The interaction Preview window disappears automatically after a timeout if the user does not accept or reject the interaction. This widget allows the user to see the time remaining to respond to the preview. See "The Overall Preview Interaction" on page 40.

Release 8.0.001   The following new features were introduced in release 8.0.001.00 of UC Connector.

- **Enhanced Instant Messaging Integration.** UC Connector now supports integrations with Genesys Instant Messaging (IM), independently of the third-party UC platform used. This allows for IM functionality in IBM

Sametime 8.x deployments. For Microsoft OCS, you can now choose instant messaging through Genesys IM or through the previous OCS-SIP Server integration.

- **Support for Microsoft Lync Server 2010**. UC Connector now supports integration with Microsoft Lync Server, for both voice and chat interactions.

Release 8.0.0   The following features were included in the initial release 8.0.0 of UC Connector:

- **Support for UC platforms.** UC Connector supports integration with Microsoft OCS 2007 R2 and IBM Sametime 8.5.

- **Presence Mapping.** UC Connector determines Knowledge Worker availability by subscribing to user presence (states/updates) provided by the UC platform.

- **Telephony Integration**. Knowledge Worker telephony integration is available through T-Server, for calls flowing from the contact center to the Enterprise. Voice call control is provided through the standard UC client.

- **Instant Messaging Integration.** When integrated with the Microsoft Office Communication Server, UC Connector supports interaction flows that use the IM integration through SIP Server. IM content and call control is provided through the Microsoft Office Communicator UC client.

- **Interaction Preview Notification.** Genesys Routing can send a preview to a targeted Knowledge Worker, letting the Knowledge Worker accept or decline the interaction before actually routing the interaction. This can be done for a selected Knowledge Worker, or round robin for a group of Knowledge Workers. Multiple preview notifications can also be sent simultaneously—broadcast—to a set of Knowledge Workers. In this case the first Knowledge Worker to respond receives the interaction.

- **Business Data Exchange.** The Knowledge Worker can access call context and attached data related to any interaction that is transferred to them. Genesys UserData is passed to the UC client, displayed on their Interaction window, the Preview window, or the custom UC Connector tab of their UC client, depending on the configuration.

- **Limited Enterprise Footprint.** All deployment related to the integration with the UC platform takes place on the Genesys side, with no need for any new applications to be running on the Knowledge Worker desktop.

- **Reporting.** Business and performance metrics about the Knowledge Worker voice activity is provided through the Genesys Reporting solution. Knowledge Workers are configured in the Genesys system as standard agents, and standard reporting products and templates can be used to generate reports on how the Knowledge Worker is used or how the call is handled.

Functional Overview

Figure 1 shows the main components and functionality involved in the UC Connector interaction between the contact center and the enterprise.
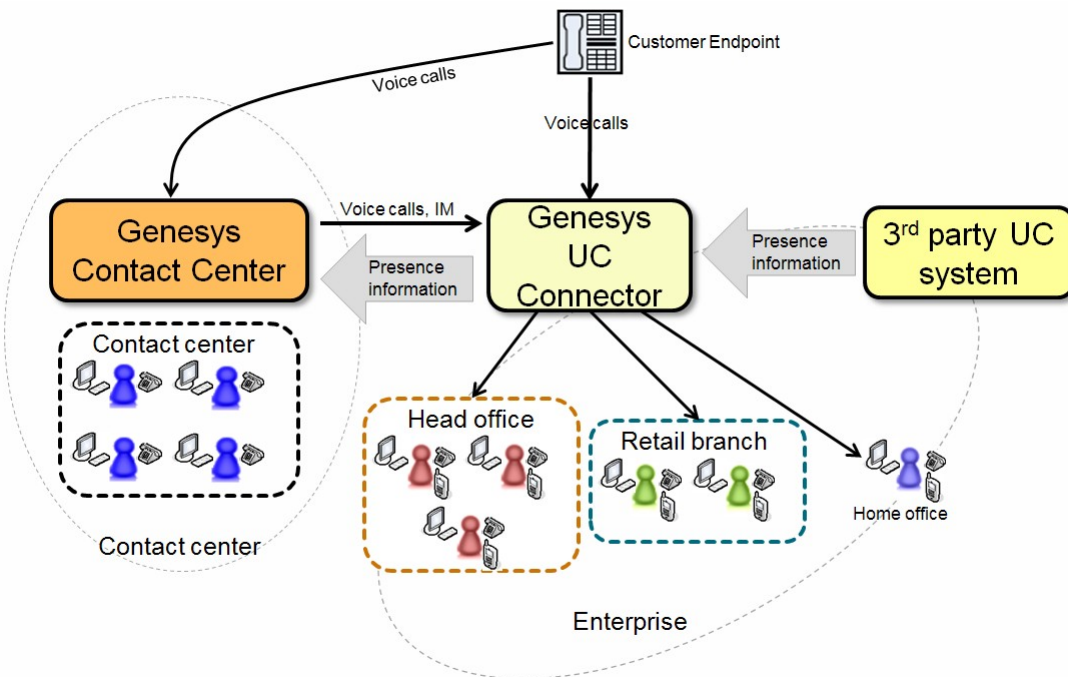
Figure 1:   Overview of the UC Connector Interactions

The Main Components

The main components involved in the UC Connector interaction between contact center and enterprise are as follows:

- **Customer**—Customers use voice channels to connect with customer service provided by the Genesys contact center or directly to the enterprise through Genesys UC Connector.

- **Contact Center Agent**—Contact center agents have CTI-enabled phones and an Agent Desktop client (Genesys Interaction Workspace, for example) that supports UserData. They can involve enterprise Knowledge Workers in the customer interactions through the UC Connector over voice or IM.

- **Enterprise Knowledge Worker**—Enterprise workers are not generally considered part of the contact center, but their expertise can benefit customer interactions. The UC Connector can involve them directly in customer service even if a contact center is not deployed.

  Knowledge Workers are represented inside Genesys as agents with access to all business rules, routing, and reporting available to regular agents.

- **Third-party UC solution**—The Unified Communications (UC) software system used by the enterprise. In 8.0, UC Connector supports integration with the following UC solutions:
  - ◆ Microsoft Lync 2010
  - ◆ Microsoft Office Communications Server 2007 R2
  - ◆ IBM Sametime 8.5
- **UC Connector**—The Genesys component used to integrate the Genesys environment with the third-party UC solution and enable enterprise-wide customer service.

The Main Functions Provided by UC Connector

The main functions provided by a UC Connector integration are:

- **Voice interaction**—The customer uses voice to connect with the contact center or directly with the enterprise customer service function. The agent can also use voice when contacting the Knowledge Worker to determine availability (for example, when initiating a voice transfer).
- **Instant Messaging (IM)**—The agent can contact a Knowledge Worker using IM, either to ask if the Knowledge Worker can accept a particular customer interaction, or to pass typed information for convenience during a phone conversation. Instant messaging is available through a platform-independent integration with Genesys Instant Messaging (IM), or through the SIP Server-OCS interaction for Microsoft OCS 2007 RN only.
- **Presence Monitoring**—By integrating with the third-party UC solution, the Genesys environment is able to determine the availability status of a given Knowledge Worker, so that it knows who might be available to handle the customer interaction.
- **Interaction Preview Notifications**—Knowledge Workers may not have job roles that allow for interruptions, even when their presence status shows they are available. When an agent sends a customer interaction to the Knowledge Worker, a screen pop or other kind of notification arrives at the Knowledge Worker's desktop, providing relevant information about the current customer interaction. This gives the Knowledge Worker the opportunity to accept or decline the interaction, based on their real availability.

Basic Architecture

Figure 2 shows the different server elements involved in a UC Connector deployment, as well as the kinds of interfaces used to communicate among the components.
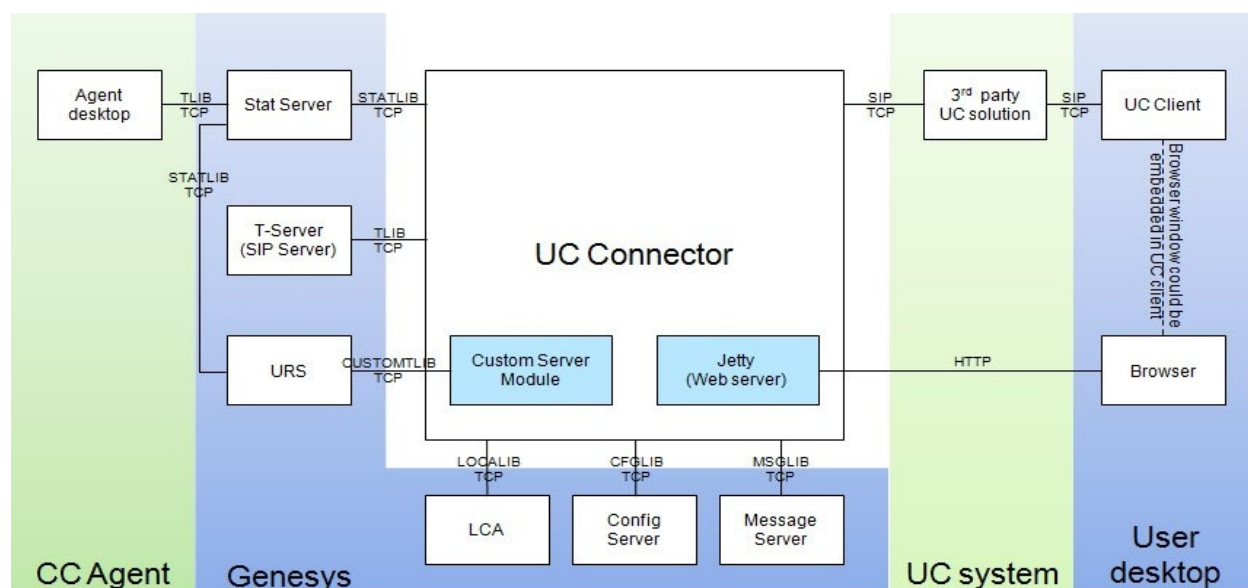
Figure 2:   Server-level Integration Architecture

Genesys Components in the Solution

The server components included in the basic architecture include the following:

- **SIP Server**—Provides both a SIP and T-Library interface. SIP Server can act as a T-Server and softswitch, for deployments with no third-party PBX. SIP Server is required for Instant Messaging.

- **T-Server**—Provides the T-Library interface between the UC Connector, the agent desktop, and the rest of the Genesys contact center components. T-Server monitors the DN status of the Knowledge Worker device (giving call control) as well as provides emulated agent functionality.

    Note: T-Server must support emulated agent functionality to integrate with UC Connector.

- **Universal Routing Server (URS)**—Provides Genesys routing for the customer interaction. URS processes the routing strategies, designed in Interaction Routing Designer (IRD) that govern how the interaction is processed. The UC Connector integrates with URS as a Custom Server, using a proprietary protocol to execute, in this case, the Preview Mechanism for the Knowledge Worker selected in the routing strategy.

- **Interaction Workspace**—The interface that appears on the desktop of Genesys contact center agents. Agents can use their workspace to send the customer interaction to the expert user or Knowledge Worker in the Enterprise, including any important notes or information that would be helpful to the expert.

  Note: Interaction Workspace 8.1 is not mandatory to the deployment. If using a different desktop client for your agents, some custom functionality might be required.

- **Stat Server**—Tracks information about the customer interaction. For UC Connector, Stat Server is used to monitor the DNs, agents, emulated agents, and other objects, making their states available to other Genesys components, in particular to URS for Genesys routing. UC Connector is also able to use statistics—for example, number of calls in a queue—that it can display in the Interaction window, so that Knowledge Workers can check availability of a Contact Point before transferring a call.

- **Local Control Agent (LCA)**—The LCA is deployed on each host computer running Genesys components, and is used to monitor the operating status of all locally running Genesys software.

- **Configuration Server**—Stores and manages the Configuration Database data that users can access through Configuration Manager or Genesys Administrator.

- **Message Server**—Receives error messages from all installed Genesys application and logs them into a common database.

Communication Between Server Components

Table 3 shows the protocols and libraries used for communication between the various server components.

Table 3: Server Interfaces

Function or Connection	Interface	Description
Call control	T-Library messaging	
Microsoft Lync and Microsoft OCS 2007 R2		
Presence	MS-PRES protocol (extended SIP from Microsoft)	MS-PRES extends SIP/SIMPLE in several ways for presence propagation

Table 3: Server Interfaces (Continued)

Function or Connection	Interface	Description
Instant Messaging	SIP messaging OR T-Library	SIP is used in direct Lync-SIP Server integration. T-Library messaging is used in Genesys chat integration.
Lync 2010 client	HTTP	A web-based "extensibility window" in Lync client uses HTTP between UC Connector and the Lync client.
Microsoft Office Communicator 2007 R2	HTTP	A web-based custom tab on MOC is maintained using HTTP between UC Connector and the MOC client.
IBM Sametime 8.5		
UC Connector to IBM Sametime	IBM Sametime SDK	The Genesys component UC Connector uses the IBM Sametime SDK to integrate with the IBM Sametime solution.
Sametime UC-client	HTTP	HTTP is used to communicate with the IBM Sametime UC client.
Instant Messaging	T-Library	Chat uses T-Library messaging in Genesys IM integrations.

High Availability

As a Genesys server component, UC Connector supports the *warm standby* redundancy method for a high availably (HA) deployment, meaning that a backup UC Connector server application remains initialized and ready to take over the operations of the primary server, when needed.

As a solution, different HA methods are used to provide greater reliability for the connections among the various components. HA is supported for the following connections:

• HA connections to Genesys servers established by UC Connector.

• HA connections to UC Connector established by Genesys servers.

• HA connections between UC Connector and the third-party UC platform.

• HA connections to UC Connector established by the UC client.

Figure 3 shows an overview of the HA connections to and from UC Connector with other parts of the integration.
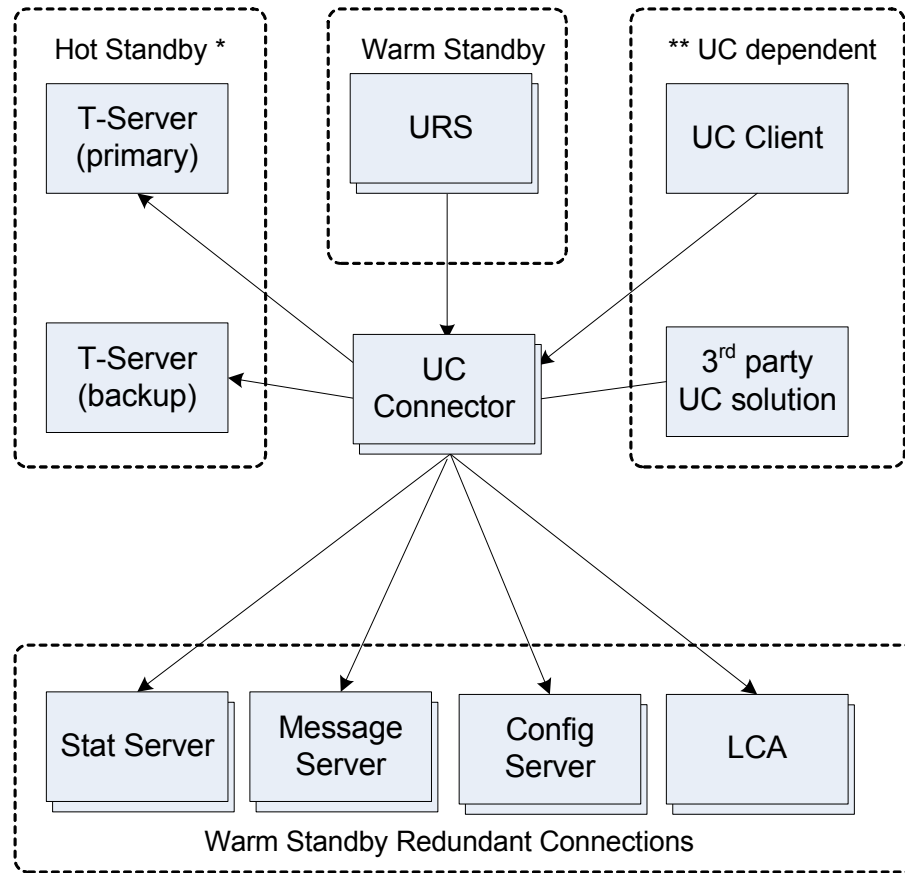
* Warm standby also supported.   ** See "Supported Integrations" for details.

Figure 3: HA Connections

High Availability with Genesys Servers

The warm standby method is used for connections with Genesys Stat Server, Configuration Server, and Message Server. In this case, if any of these connections fails, the UC Connector will try to re-connect with the backup server instance. After a successful reconnection, UC Connector continues normal operation.

High Availability with Genesys T-Servers

For connections with T-Server, the *hot standby* method for HA provides a more robust redundancy. In this case, the solution maintains a connection with both T-Server instances, actively passing call data from primary to backup T-Server. This lessens the interruption in service, and allows for better survival of calls after the switchover process.

Warm standby connections are also supported with T-Server.

For more information, consult the Deployment Guide for your T-Server.

High Availability with Universal Routing Server

For connections established from Universal Routing Server (URS) and the UC Connector, the warm standby method is used. If URS connection to UC Connector fails, it reconnects with the backup UC Connector instance, then resumes normal operation. Call processing during this process is dropped or delayed.

Connections to UC Connector initiated through the `Custom Server` are switched over as part of the Windows NLB configuration, together with the web interface. For configuration details, see Chapter 7 on .

High Availability with Third-party UC platform

For connections with the third-party UC platform, the HA method used depends on which UC platform the UC Connector is integrated with.

High Availability with Microsoft OCS

HA is only supported with Enterprise Edition of Microsoft OCS—Lync or OCS 2007 R2 (Standard Edition places all components on a single host).

In this case, multiple Front End Servers make up a highly available pool of resources. UC Connector initiates and manages the connection to the OCS pool, which is contacted through a single URL or virtual IP address. If the connection between UC Connector and this OCS contact point fails, UC Connector initiates the reconnection process.

In this case, the OCS platform is configured behind a third-party load balancer.

For a sample architecture diagram, see "Deployment with Microsoft OCS Enterprise Edition" on . For more information, consult the Microsoft documentation.

Load balancing through third-party hardware is also available in deployments with multiple instances UC Connector, so that the Microsoft Communicator client can use a single point of contact, regardless of which UC Connector instance is primary at any moment.

High Availability with IBM Sametime

IBM Sametime 8.5 provides Community Server clusters for load balancing and failover functionality in cases where HA is required in the deployment. Similar to integration with Microsoft OCS 2007 R2, UC Connector integrates with IBM Sametime through a single point of contact—either a URL or a virtual IP address. If the connection between UC Connector and this IBM Sametime contact point fails, UC Connector initiates the reconnection process, also using the warm standby mechanism. After a successful reconnection, UC Connector resumes normal operation.

Configuring High-Availability UC Connector Instances

For information about configuring UC Connector for a high-availability deployment, see Chapter 7, "UC Connector High Availability Deployment," on page 171.

Deployment Modes

You can deploy and configure UC Connector in two modes: normal and standalone.

Normal Mode

This is the typical UC Connector deployment, in which UC Connector subscribes to a Unified Communications system and interaction preview pop-ups are presented to users.

Standalone Mode

In standalone mode, UC Connector does not subscribe to a Unified Communications system to receive the user's presence. This may be because no compatible UC system is present in the Enterprise. In this case, UC Connector uses its own information to relay presence information to the Genesys environment. Users can control their presence state manually, by logging in and out of the UC Connector and acting on the control to put themselves in one of the presence states that are defined for them. See "Customizing Knowledge Worker States" on page 155 and the enable-logout-menu option.

Note: These options are only enabled in standalone mode.

Limitations

UC Connector 8.0 currently has the following limitation:

• UC Connector does not support After Call Work (ACW) time for Knowledge Worker users. Ensure that ACW in the Agent Login for the Knowledge Worker is disabled (set both `wrap-up-time` and `legal-guard-time` to `0`).

Note: ACW is supported for contact center agents who use Microsoft Lync Voice.

Chapter

2

How It Works

This chapter describes how the different functions enabled by the UC Connector work.

How It Works—Voice Scenarios

The UC Connector integration supports voice interactions between the Knowledge Worker, the contact center agent, and the customer. This can involve direct calls from the customer to the enterprise customer service function, or transfers and conference calls initiated by the agent. Knowledge Worker transfers to agents in the contact center are also available, as well as transfers and conferences between Knowledge Workers.

Contact Center to Knowledge Worker

In this scenario, either a customer calls directly into a number that maps to a group of Knowledge Workers engaged in customer support (bypassing the contact center) or a contact center agent engaged in a voice call with a customer decides that help from an expert outside of the contact center is needed, and so initiates a call transfer. In both cases the interaction can go

either to a particular Knowledge Worker, or more commonly to a group of Knowledge Workers. The call flow for this scenario is as follows:
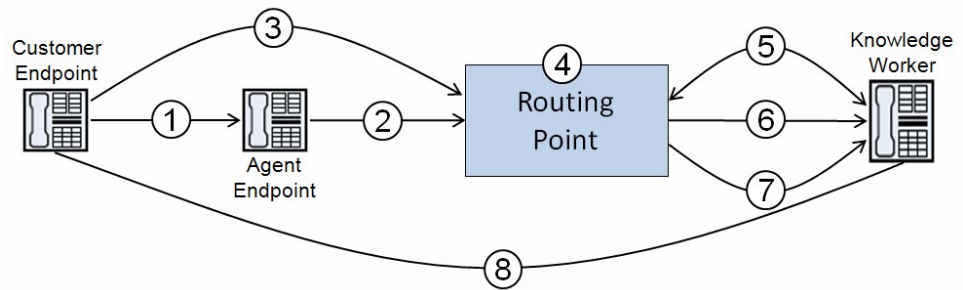
Figure 4:   Contact Center to Knowledge Worker Call Flow

1. The customer dials the contact center and a voice channel is established with a particular agent. Regular Genesys call distribution from a Routing Point or ACD Queue determines which agent will handle the customer interaction.

2. The agent decides that a Knowledge Worker is needed to satisfy the customer interaction. From their Interaction Workspace, the agent initiates a call transfer to a designated Knowledge Worker `Routing Point` DN.

3. Or, the customer contacts an Enterprise Customer Support number directly, which does not involve the contact center.

4. The routing strategy loaded onto this `Routing Point` DN determines how the particular Knowledge Worker is selected. You can design the strategy so that:

    •   The agent specifies the specific Knowledge Worker that they need.

    •   The routing strategy selects a particular Knowledge Worker from a group, using a round-robin approach based on worker availability. This distributes the workload, making sure the same expert is not overused (recommended approach).

    •   The routing strategy sends or "broadcasts" multiple Preview Notifications simultaneously to a group of Knowledge Workers (the interaction will be routed to the first Knowledge Worker who accepts).

5. Presence monitoring (through integration with Genesys Stat Server and Management Layer) determines which Knowledge Workers are currently available, and this information is made available to the routing strategy.

6. Using the presence information, the routing strategy selects an available Knowledge Worker.

7. Universal Routing Server (URS) sends a Preview Interaction to the selected Knowledge Worker—a screen pop arrives at the Knowledge Worker's desktop or device, asking if they will accept the interaction.

8. If the Preview is accepted, the customer call is then routed to the Knowledge Worker—a new voice channel between customer and enterprise expert is created.

   If the Preview is declined, URS can apply default routing, or select another available Knowledge Worker for Preview, depending on the strategy.

Knowledge Worker to Contact Center

In this case, the Knowledge Worker is already involved in a customer call—typically after a successful transfer from contact center to Knowledge Worker earlier in the customer interaction. For whatever reason—to collect more user information, for example, or process a new order—the Knowledge Worker decides it is necessary to transfer the customer back to the contact center. As part of the UC Connector integration, the Knowledge Worker is able to use customizations on their UC client to send the call to an agent in the contact center. Regular Genesys routing is used to select a particular agent—for example, the Knowledge Worker sends the call to a Genesys `Routing Point` DN, where the strategy selects a particular agent based on skill, group, and so on.

A common call flow for this scenario is as follows:
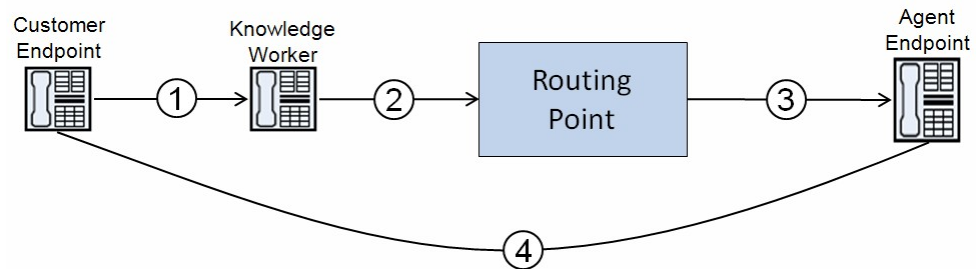
Figure 5:   Knowledge Worker-to-Contact Center Call Flow

1. A Knowledge Worker is engaged in a voice call with a customer, with no other party or contact center agent involved (in other words, this is not a conference or consultation call with the initiating agent still involved).

2. The Knowledge Worker decides that the call should be sent to the contact center, either back to the original agent who started the customer interaction, or for further processing by the contact center (for example, to an IVR for a customer satisfaction survey).

   To initiate the transfer, the Knowledge Worker uses the Interaction window to select a configured UC Connector contact point. This contact point is mapped to a `Routing Point` DN in the Genesys environment.

3. Once the call arrives on the `Routing Point` DN, the routing strategy loaded on the DN is responsible for selecting an available agent, configured for the business rules of the contact center.

4. Connection between customer and selected agent is established.

Knowledge Worker to Knowledge Worker

In this case, a Knowledge Worker already engaged on a customer voice call decides that the interaction should be handled by another Knowledge Worker in the enterprise. The call is routed through the Genesys environment, and the targeted Knowledge Worker is sent a Preview interaction asking if they can accept the customer call.

A common call flow for this scenario is as follows:
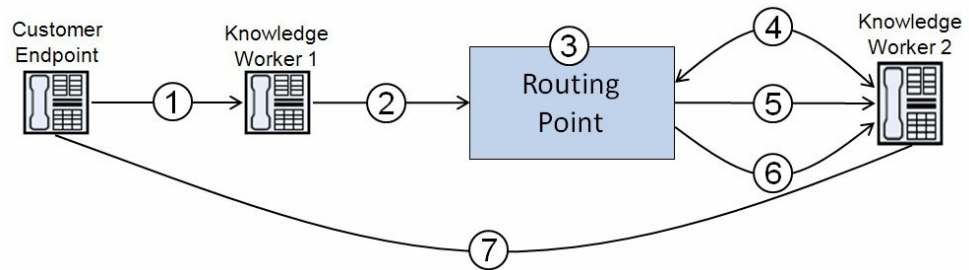
Figure 6:   Knowledge Worker-to-Knowledge Worker Call Flow

1. A Knowledge Worker is engaged in a voice call with a customer.

2. The Knowledge Worker initiates a transfer or conference to another Knowledge Worker in the enterprise by selecting a contact point in the `Interaction` window. This contact point is mapped to a `Routing Point` DN in the Genesys environment.

3. Once the call arrives on the `Routing Point` DN, the routing strategy loaded on the DN begins the process of selecting an available Knowledge Worker.

4. Presence monitoring determines which Knowledge Workers are currently available, and this information is made available to the routing strategy.

5. Using this presence information, the routing strategy selects an available Knowledge Worker.

6. Universal Routing Server (URS) sends a Preview Interaction to the selected Knowledge Worker—a screen pop arrives at the Knowledge Worker's desktop or device, asking if they will accept the interaction.

7. If the Preview is accepted, the customer call is then transferred to the new Knowledge Worker.

    If the Preview is declined, URS can apply default routing, or select another available Knowledge Worker for Preview, depending on the strategy.

How It Works—Instant Message Scenarios

Contact center agents can use Instant Messaging to communicate with a Knowledge Worker in the enterprise, either to approach a Knowledge Worker

about a particular customer interaction, or to convey information through an IM session (text) instead of by voice over an existing call

Agent Sends IM During a Customer Call

In this scenario, an agent is currently on a call with a customer when the agents starts an Instant Message with a Knowledge Worker. Once the IM with the Knowledge Worker is established, the agent can continue chatting with the Knowledge Worker—for example, to obtain some expert details to pass on to the customer on the call—or they can initiate a voice connection (escalate the customer interaction) to the Knowledge Worker by transfer or conference.

A common call flow for this scenario is as follows:
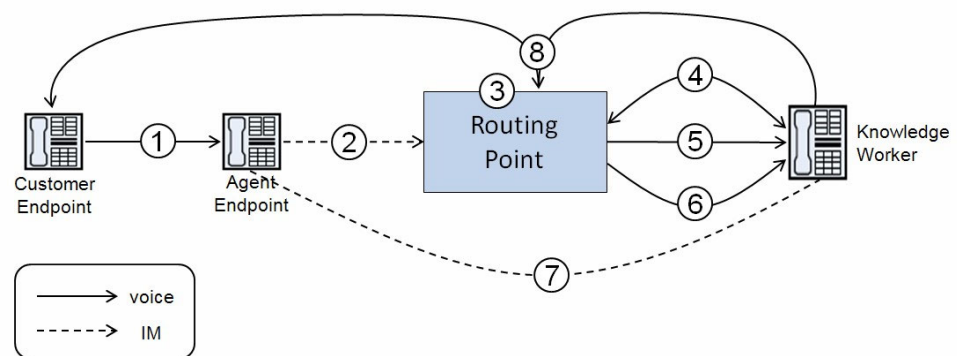
Figure 7:   Agent Sends Instant Message to Knowledge Worker

1. An agent is engaged in a voice call with a customer.

2. The agent decides that a Knowledge Worker is needed to satisfy the customer interaction. From their Interaction Workspace, the agent initiates an IM to a designated Knowledge Worker `Routing Point` DN.

3. The routing strategy loaded onto this `Routing Point` DN determines how the particular Knowledge Worker is selected.

4. Presence monitoring (enabled by UC Connector ability to map UC presence status to Genesys agent states) determines which Knowledge Workers are currently available, and this information is made available to the routing strategy.

5. Using the presence information, the routing strategy selects an available Knowledge Worker.

6. Universal Routing Server (URS) sends a Preview Interaction to the selected Knowledge Workers—a screen pop arrives at the Knowledge Worker desktop, asking if they will accept the interaction.

7. If the Preview is accepted, the IM is then routed to the Knowledge Worker, and the IM session between agent and Knowledge Worker begins.

If the Preview is declined, URS can apply default routing, or select another available Knowledge Worker for Preview, depending on the strategy.

8. The agent can then continue chatting, or the agent can transfer or conference (escalate) the customer call to the Knowledge Worker. In the case of transfers, this can be done through either of the following methods:

   ◆ The call is sent to a `Routing Point` DN with the relevant attached UserData to identify the Knowledge Worker currently handling the IM. Depending on the business rules, the strategy can then present a preview or just route the call directly.

   ◆ With Interaction Workspace, the voice DN of the Knowledge Worker who accepted the IM is presented as a transfer target to the initiating agent. The agent can then transfer the call directly to this Knowledge Worker, without going through a Routing Point.

   The IM and the voice call are handled separately.

How It Works—Presence Monitoring

To determine whether a particular Knowledge Worker is available for a customer interaction, the UC Connector monitors the presence status of the Knowledge Worker on the UC platform. Knowledge Workers are integrated into the Genesys environment as emulated agents, and their presence status in the UC client is mapped to a corresponding "agent status" in the Genesys contact center. Any change in the status in the UC client for a particular Knowledge Worker triggers a corresponding change in agent status, as monitored by Stat Server.

Changing Knowledge Worker Status in Genesys

When the UC Connector receives notification about a change in presence status for a particular Knowledge Worker, it sends a T-Library request to change the status of the corresponding Knowledge Worker "agent" (`RequestAgentLogin`, `RequestAgentReady`, and so on) to the T-Server that manages the Knowledge Worker's Genesys DN. The exact mapping of UC client presence status to Genesys agent status is configurable, but the default mapping is shown in Table 4.

Table 4: Mapping Presence to Agent Status

UC Presence Status	Genesys Agent Status
Online	Ready
Offline	LoggedOut
Any other status	NotReady + Reason code

T-Server Support for Emulated Agents

Because the Knowledge Worker must be configured as an emulated agent, the T-Server managing the Knowledge Worker DN must support the emulated agent feature.

Presence and Agent-Status Scenarios

Changes in the Knowledge Workers presence in Genesys can come from the UC Client side, or from the UC Connector (Genesys) side, depending on the scenario. UC Connector integrates both sources of presence for each user. Table 5 describes some of the scenarios that result in a change of presence for Knowledge Workers.

Table 5: Status Change Scenarios

Action	Resulting Change In Status
Manual change in UC-client.	The Knowledge Worker changes status manually in the UC-client. Whatever change is made is reflected in the Genesys environment, with the emulated agent DN that represents the Knowledge Worker being set to `Ready`, `NotReady`, or `LoggedOut`.
Knowledge Worker accepts a call.	On accepting a routed call from the contact center, the Knowledge Worker's presence status is changed in the UC-client to `Do Not Disturb`.
Knowledge Worker declines a preview.	If the Knowledge Worker declines the preview interaction, the Knowledge Worker's agent status is changed (by default) to the agent status of `NotReady`.
	After a configured `NotReady` timeout period (5 minutes by default), the Knowledge Worker's agent status is updated to whatever their current presence status is in the UC-client. For example, if the UC-client status is still `Online`, then the Knowledge Worker is put back into `Ready` agent status.

Table 5: Status Change Scenarios (Continued)

Action	Resulting Change In Status
Knowledge Worker is unavailable for contact-center work.	The Knowledge Worker changes status to `Do Not Disturb` in the UC Connector GUI, which is either included in the UC-client or shown in a browser window. This sets their Genesys agent status (by default) to `NotReady`, but does not otherwise affect their presence status for the enterprise. When this setting is enabled, regular UC-client status changes are not mapped to changes in agent status in Genesys. Only once the Knowledge Worker indicates that they are again `Available to the Contact Center` does regular presence-to-agent status mapping apply.
Automatic change in UC Connector client	Most UC Clients are integrated with the user's calendar and are aware if the Knowledge Worker enters a meeting or is otherwise unavailable. When this happens, the user's presence status is reflected in the corresponding agent status in Genesys. Similarly, UC Clients detect when the user has been idle for a while, and this is also captured by UC Connect.

Presence Integration with Microsoft OCS

UC Connector communicates with Microsoft Lync (and OCS 2007 R2) using MS-PRES (a Microsoft protocol that extends SIP for presence) over TCP-IP and TLS. UC Connector connects to Lync through the Front-End Server, directly or through a third-party load balancer. For more information about supported deployments, see "Microsoft Office Communication Server" on .

UC Connector registers with OCS using SIP `REGISTER` requests, and OCS responds with a `200 OK`. After registration is completed, UC Connector subscribes to the status of Knowledge Workers using SIP `SUBSCRIBE` requests, and OCS responds with a `200 OK`. This occurs only when the Knowledge Worker logs in to the `UC Connector` tab in Office Communicator.

Whenever a change is made to the Knowledge Worker's presence status in Microsoft Office Communicator, OCS sends a `BENOTIFY` (Best Effort `NOTIFY`, an MS-PRES message) to UC Connector with the updated status. UC Connector changes the agent status for the Knowledge Worker's corresponding Genesys DN based on mapping rules.

Presence Integration with IBM Sametime

UC Connector communicates with IBM Sametime using an IBM proprietary protocol implemented through Sametime SDK. This connection is established over TCP/IP.

UC Connector logs into Sametime as a server. When successfully logged in, UC Connector receives login notifications from the users (Knowledge Workers) in the Sametime community. If the Knowledge Worker is configured in Configuration Layer and logs into the custom panel—the Genesys Contact Center panel—in their Sametime Connect client, UC Connector opens a status notification subscription with Sametime for that Knowledge Worker.

Whenever a change is made to the Knowledge Worker's presence status in the IBM Sametime Client, UC Connector changes the agent status for the Knowledge Worker's corresponding Genesys DN, based on rules configured in the application or in the Knowledge Worker `Person` object.

Push On-Call Status from Genesys to the UC Platform

When integrated into a UC platform, Genesys can push the telephony status of a Knowledge Worker to the UC provider. If push-status functionality is enabled, when the Knowledge Worker receives a call, UC Connector updates the Knowledge Worker UC status to the default `Do Not Disturb`. This status setting is configurable (see `oncall-status` for OCS, and `oncall-status` for Sametime)

How It Works—The Preview Window

UC Connector uses the Custom Server module, a Universal Routing Server (URS) component built into the UC Connector itself, to handle the Preview interaction. By using the Custom Server module, UC Connector processes proprietary `CUSTOMLIB` protocol messages sent from the routing strategy, in order to initiate the Preview interaction with the third-party UC client.

HTTP is used as the transport method for both supported UC platforms.

Figure 8 shows how the embedded Custom Server handles the Preview Interaction between URS and the UC platform.
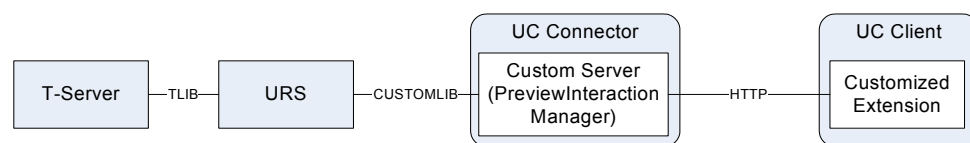
Figure 8:   Routing Integration Architecture

The Overall Preview Interaction

Depending on how the routing strategy is configured, the Preview interaction can be sent to a particular individual Knowledge Worker, or in a round-robin manner (consecutively) to a pool of available Knowledge Worker resources, continuing until one of them accepts the interaction. You can also design the routing strategy to broadcast notifications to several Knowledge Workers in a group, where the interaction is then sent to the first Knowledge Worker that accepts the preview.

The call flow for the overall interaction is as follows:

1. Customer interaction is initiated towards the Knowledge Worker,

2. Based on the Knowledge Worker presence status in the UC client (mapped to the corresponding agent status on the Genesys side), URS selects an available Knowledge Worker and initiates a preview interaction with that user—the Preview window appears on their device. If the `audio-on-preview` option (introduced in release 8.0.2) is configured, the specified audio will also play to alert Knowledge Workers who may not be at their desk that an interaction Preview has arrived.

3. A countdown timer appears in the preview window (the length of the timeout period is configurable). The Knowledge Worker must respond before this timer runs out, otherwise the call is returned to URS, where the strategy can select a new Knowledge Worker.

4. On accepting the Preview, an incoming call notification window appears on the Knowledge Worker device (typically, the device is also ringing). If accepted, the voice call between customer and Knowledge Worker is established.

The Preview Notification Window

Figure 9 shows a sample Preview window for an incoming voice call, as it appears on the desktop of a Knowledge Worker with Microsoft Lync. The interaction also appears in the UC Connector GUI in a browser window (not show).
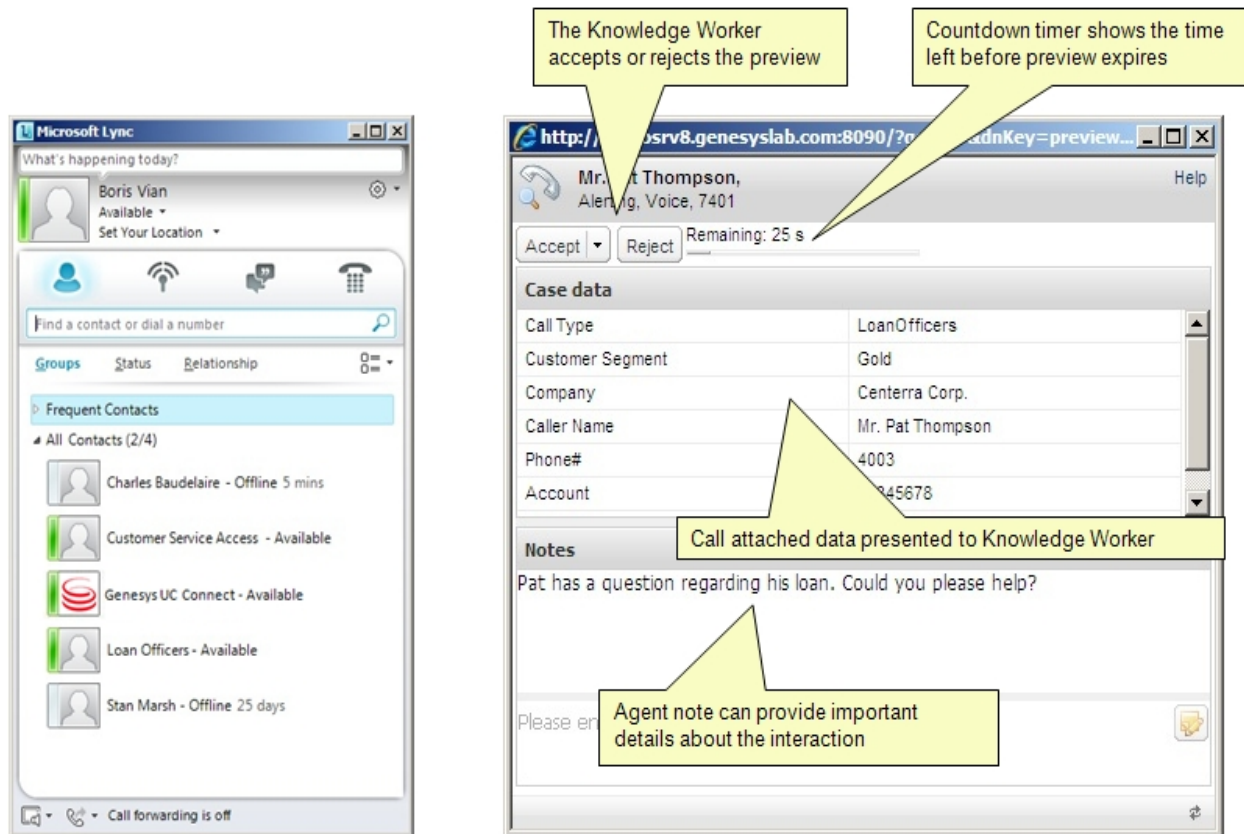
Figure 9: Sample Preview Window

The Preview window can display any user information available to the routing strategy—in other words, any customer information stored or collected earlier in the interaction, in order to give relevant details of the interaction to the Knowledge Worker. In this case, customer information such as phone number and service level are included.

Closing the Preview Window

This Preview window will close if the Knowledge Worker clicks the Accept or Reject buttons. It will also close automatically if the Knowledge Worker fails to respond before the timeout period, or if the interaction is canceled on the agent/customer end.

How It Works—The Interaction Window

When the interaction is accepted, the Interaction window or IM session interface will appear instead of the Preview window, depending on the kind of interaction requested.

Figure 10 shows the Interaction window, which allows the Knowledge Worker to see information about the interaction and perform any call control actions they need (transfer, hold, and so on). Because the window is browser based, it has the same appearance no matter which UC platform is used.
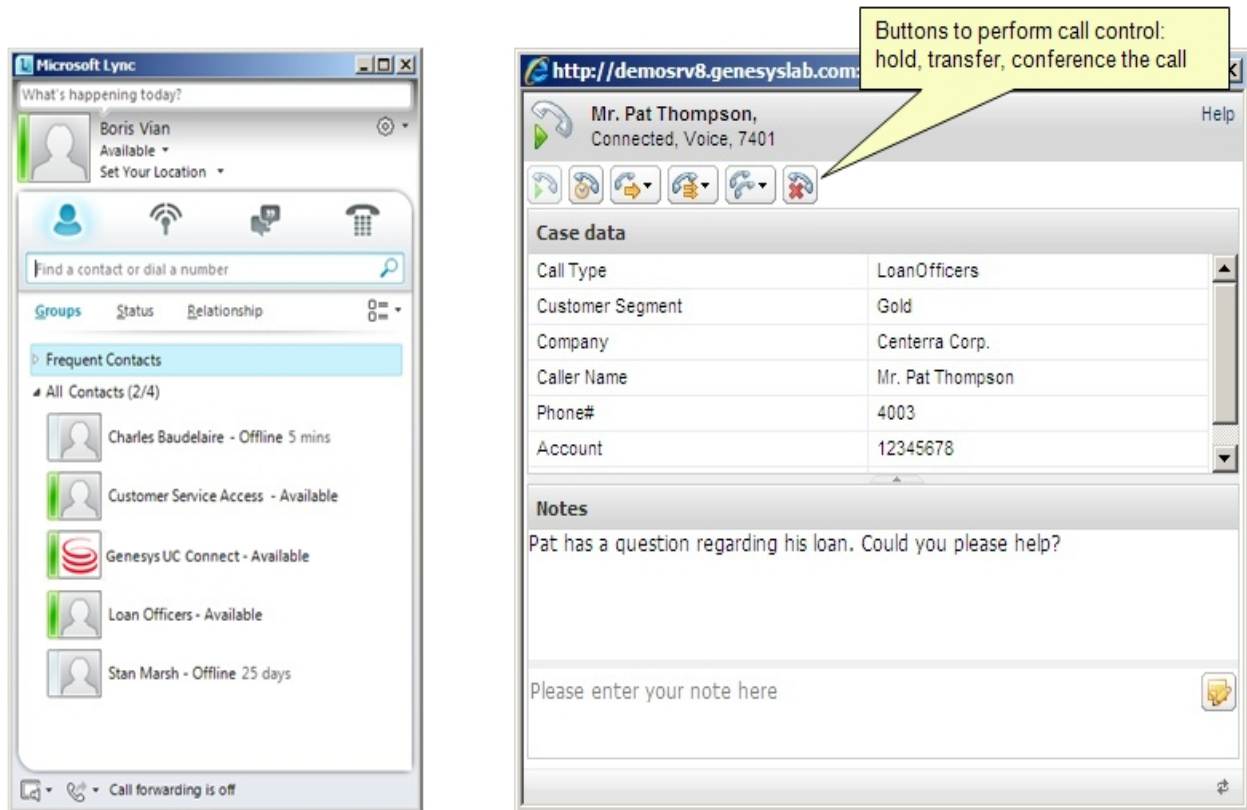
Figure 10: Sample Voice Interaction Window

How It Works—Reporting Events

UC Connector produces and sends reporting-related events for the Interaction Preview mechanism to reporting platforms through the T-Server/SIP Server. To enable reporting, set the `enable-preview-reporting` option to `true` in the `uc-connector` section of the UC Connector `Application`.

The feature is designed to be used with the ICON Custom Agent State feature, although other reporting platforms can be used. See the Using Custom States section of the *Interaction Concentrator 8.1 Deployment Guide* for form information on ICON.

The T-Library function `TDistributeUserEvent` is used to report Preview offers and completion results for each Knowledge Worker offered an interaction Preview. UC Connector sends a request to T-Server to distribute the corresponding event to all registered clients. A reporting application then makes relevant detailed records upon receiving these events.

When there are invalid configuration parameters, the default values will be used.

Message Content

All events sent by UC Connector to T-Server will contain the following:

- `ConID`—Connection ID of the current call.
- `ThisDN`—The main DN of the Knowledge Worker.
- `AgentID`—The Knowledge Worker's Agent ID.
- `UserData`—This is the custom state-related information to be sent to T-Server/SIP Server.

The Preview offer will start the custom state identified by the key `UCC_Preview` and the number `3271`, which is customizable by the configuration option `preview-state-name`. The Preview offer event will also record the associated ConnID using the key `UCC_ConnID`, and the Knowledge Worker's user ID using the key `UCC_UserId`, and the Knowledge Worker's login ID using the key `UCC_AgentId`.

The Preview termination event will record the reason for the termination using the key `UCC_Reason`. One of the following reasons will be recorded:

`Accepted`—The user accepts the interaction.

`Rejected`—The user explicitly rejects the interaction by pressing the appropriate button or key.

`Timeout`—The user does nothing while the interaction Preview is open.

`Taken`—Another user accepts the interaction. This can only occur in broadcast mode.

`Error`—An error occurred while showing the interaction Preview window.

`Cancel`—The interaction was abandoned by the original caller.

Note: The keys `UCC_ConnId`, `UCC_UserId`, `UCC_AgentId`, and `UCC_Reason` are configurable with the option `presence-gateway-mode`.

Preview offer event User Data

```
UCC_Preview = "+"
UCC_ConnId = "3271, <ConnID(hex number)>"
UCC_AgentId = "3271, <agent_id>"
UCC_UserId = "3271, <user_id>"
```

Preview termination event User Data

```
UCC_Reason = "3271, {Accepted|Rejected|...}"
UCC_Preview = "-"
```

How It Works—Lync Integration

In integration with Microsoft Lync, Knowledge Workers see the UC Connector user interface on a browser window, or possibly in a Lync Conversation Extensibility window. Both these windows are web-based and allow the Knowledge Worker to control their UC Connector presence status, to accept or reject interactions in the Preview window, and to control ongoing chat or voice calls in the Interaction window.

There are two ways for the Knowledge Worker to access the UC Connector from the Lync client:

• The Knowledge Worker manually opens the UC Connector web client from the Lync `Tools` menu.

• The Knowledge Worker accepts a Lync preview interaction from UC Connector, which opens the extensibility window.

Note: If the Knowledge Worker uses Safari as their default browser, they must keep a browser window open in order to receive previews for incoming interactions. If the user closes the browser, Lync-based alerts will not be received.

The Genesys UC Connector Web Client

In the Lync client, under `Tools`, select `Genesys UC Connector` from the drop-down menu to open the UC Connector web client.
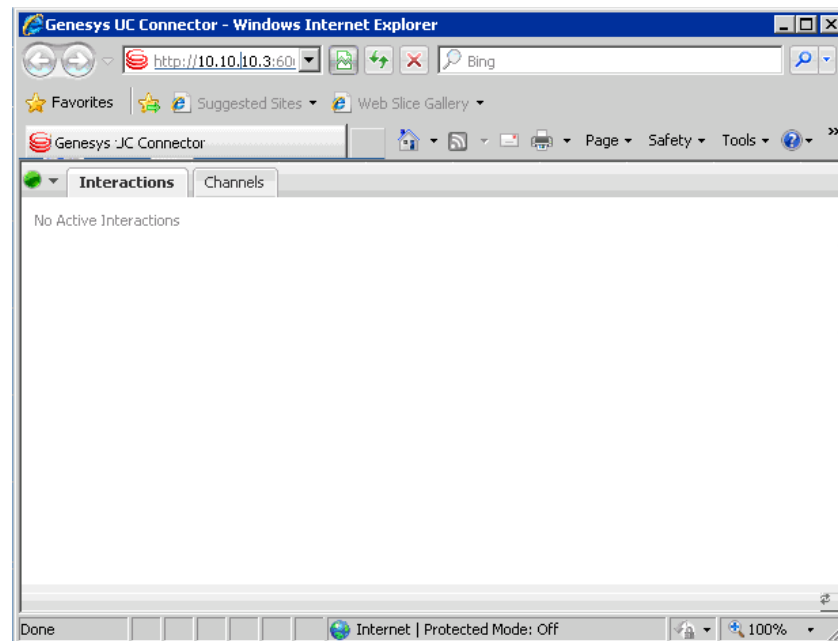
Figure 11: Genesys UC Connector Web Client

Using the drop-down menu, the Knowledge Worker can change their UC Connector presence status. This status does not change if the Knowledge Worker then closes the web client.

While this window is open, an incoming interaction arrives as a regular UC Connector-based Preview notification.

If this window is closed, an incoming interaction will arrive as a Lync Preview invitation (see Figure 12). If this invitation is accepted, the UC Connector extensibility window opens (see Figure 13), followed by the UC Connector Preview notification.

The Lync Preview Invitation

If the extensibility window or the UC Connector web client (see previous section) is not already open, a new interaction will arrive at the Knowledge Worker's desktop using the native Lync chat request.
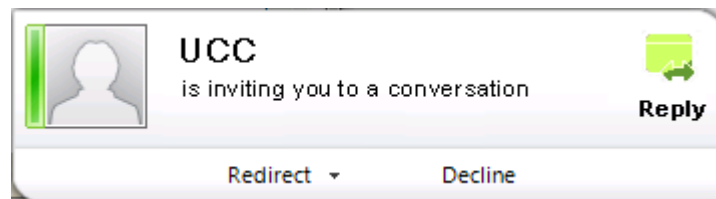
Figure 12: Lync Preview for a UCC Interaction

If the Knowledge Worker accepts this invitation, the extensibility window opens.

The Extensibility Window

This window shows both the UC Connector tab as well as a Lync chat window used by the system to provide a configurable welcome message (`invite-message`, page 202) to the Knowledge Worker. This window will only be enabled if the `invite-message` option is present and has a value configured.
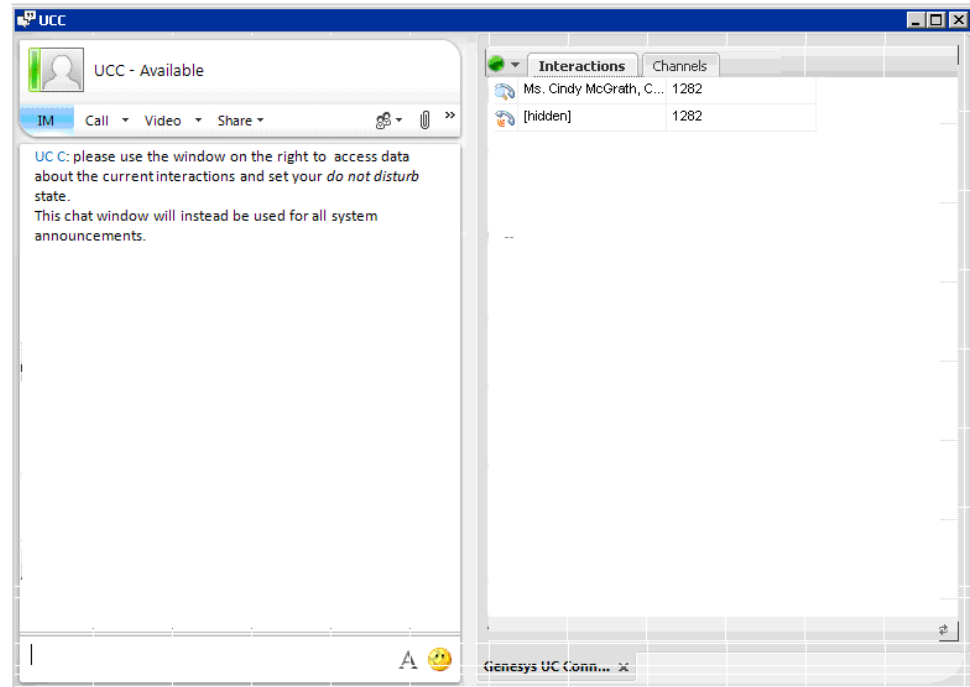
Figure 13: Extensibility Window On Accepting an Interaction

After the interaction is finished, the UC Connector tab remains open in the extensibility window. If the Knowledge Worker keeps this window open, any new interaction will arrive as a separate web-based Preview window. If the extensibility window is closed, new interactions will arrive once again as a native Lync chat request.

Preview and Interaction Windows

With Lync (as with other the other supported UC systems), the Preview window opens when a new interaction arrives at the Knowledge Worker's desktop, and the extensibility window has already been opened (the web connection to UC Connector is established). If the Knowledge Worker accepts the invitation in the Preview window, the Interaction window opens in its place.

How It Works—Customized Help

UC Connector lets you create customized Help files that Knowledge Workers can access from the web-based user interface. If enabled on a particular window, a clickable Help button appears, providing a link to a `.html` help file that opens in the Knowledge Worker's browser. A sample `.html` help file is included on the product CD, under `Documentation/Help`. You can use this file, or create one of your own, stored at a network-accessible location.

You can specify a different help file for each of the following windows in the user interface:

- Login screen (`help-login-url`)

- Interaction window (`help-interaction-url`)

- Preview and Interaction windows (`help-callcontrol-url`, the same option is used for both windows)

For detailed procedures, see Task Summary: Customizing the Help Button, on page 152.

The following figures show some sample screens where the Knowledge Worker can access an enabled `Help` button.
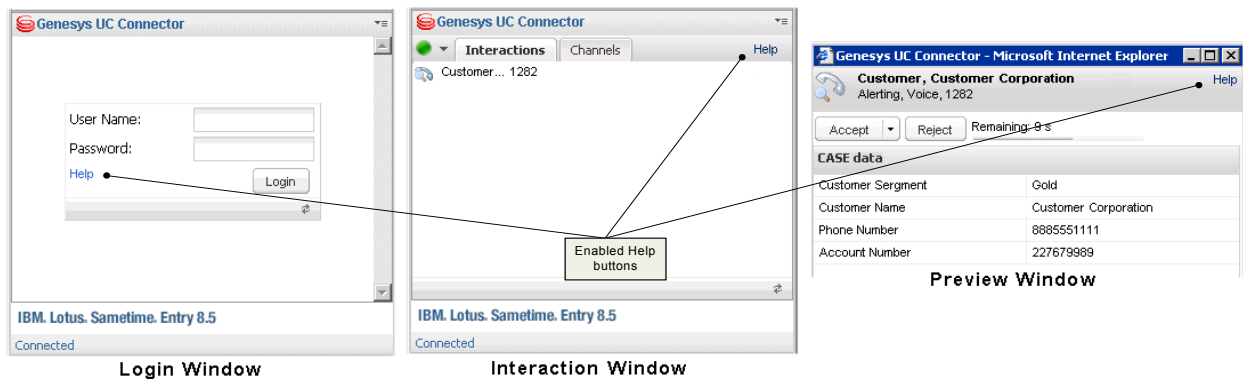
Figure 14: Customized Help Button

How It Works—Customized Languages

UC Connector allows you to change the language used in the client interface. By default, UC Connector uses English for the various tools and labels in the interface. Using the configuration option `locale`, you can change the default language to one of several supported languages.

For configuration details, see Task Summary: Changing the Default Language, on page 153.

The following figure shows a sample Interaction window localized for the French (fr) language.
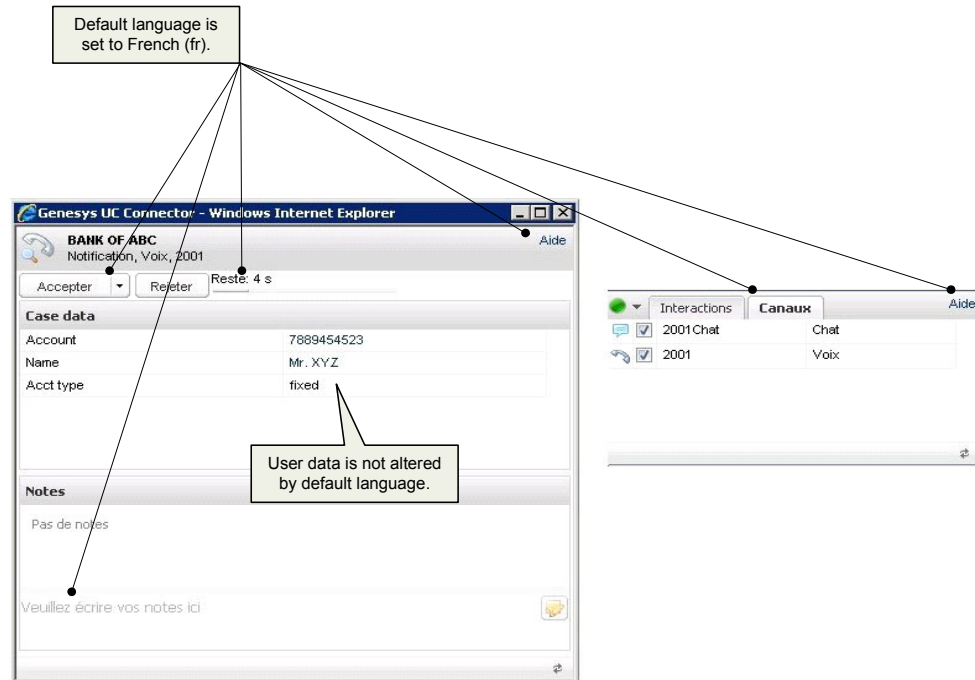
Figure 15: UC Connector with Default Language Set to French

List of Supported Languages

The following table lists the languages supported by UC Connector, as well as the two-character country codes used to specify the language in the configuration.

Table 6: Supported Languages

Supported Languages	Language Codes
English	en
French	fr
German	de
Italian	it
Russian	ru
Spanish (Latin America)	es

When Other Languages May Be Used

If `locale` option is set to `default`, then UC Connector uses the language of the local operating system—provided it is included in the list of supported

languages. If UC Connector integrates with a web browser that uses a preferred language setting, then the web browser's preferred language takes precedence over the internal UC Connector setting—again, only if the web browser language is included in the list of supported UC Connector localizations.

Note: With IBM Sametime and Microsoft OCS, the UC Connector tab is embedded in the UC client using Internet Explorer. If the Knowledge Worker machine uses another browser (not IE) as default, then you must set the language preference in both browsers (IE for embedded tab and non-IE for default browser) in order for the localized strings to appear correctly in both the UC Connector tab and the Interaction/Preview windows.

If the `locale` option is not configured at all, then English is used as the absolute default.

How It Works—Customized Knowledge Worker States

UC Connector allows you to customize the states available to knowledge workers in the UC Connector web client drop-down menu. The "agent states" and the corresponding text displayed in the menu can be customized by editing application resources. This custom state functionality is only supported in the standalone deployment mode. See "Standalone Mode" on . If UC Connector is connected to an external presences source, such as Microsoft Lync, the presence updates from Lync will override the state set in T-Server.

Each customizable state is characterized by a unique combination of the major "agent state" (Ready or Not Ready), the agent mode, and a reason code. The customized states can be made available for display in the UC Connector web client drop-down menu. The customized states can also be used to reflect the current user's state in the `Channels` tab.

Before release 8.0.300, UC Connector included two fixed states that mapped to the Genesys states Agent Not Ready and Ready (without an agent mode or reason code):

• Do Not Disturb On

- Do Not Disturb Off.
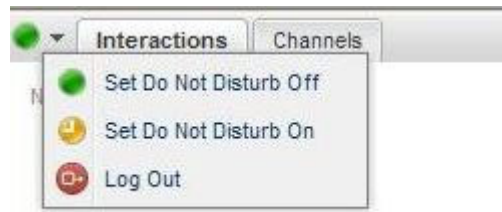
Figure 16: Default menu options.

In release 8.0.300, the UC Connector application is installed with these two states as an example in the XML resource file. The states are also built in the application to enable default handling when the resources are corrupted.

The editable XML resource file contains the state definitions and their corresponding visual representation in the UC Connector web client menu, such as text strings, icons, and translations. The provided example XML can be edited by the system administrator or integrator as required.

The XML file can be provided on the local file system or through HTTP(s) service. The HTTP(s) service can be convenient for managing multiple instances of UC Connector for redundancy or load sharing. The instance of Jetty web server packed with UC Connector can also be used to service this file and its related resources, such as localization files, icons, and graphics.

See "Customizing Knowledge Worker States" on for details on the setup procedures.

Defining Knowledge Worker States

Each Genesys agent state in the XML file is composed of a unique combination of the following:

- Main state—Ready or Not Ready. This is the 'agent state" that will show up in real-time or historical reporting.
- Agent mode—Manual, After Call Work, Legal Guard, Auxiliary Work, Walk Away, and Nodisconnect
- Reason code string—A free-form text string that UC Connector attaches as a Reason or Extension attribute with a user-defined key name to the corresponding agent state change request sent to T-Server. The key name `ReasonCode` is used by default, but the name can be customized.

The agent mode and the reason code are optional.

UC Connector uses the definitions of the states to display the current user state in the agent status indicator on the main panel and the corresponding channel state in the `Channels` tab of the UC Connector web client. The appearance of the user state is defined by the icon and the status string associated with the corresponding state definition.

To make custom state recognizable in the UC Connector web client, you must provide distinct relevant icons. The system integrator is responsible for providing the corresponding artwork.
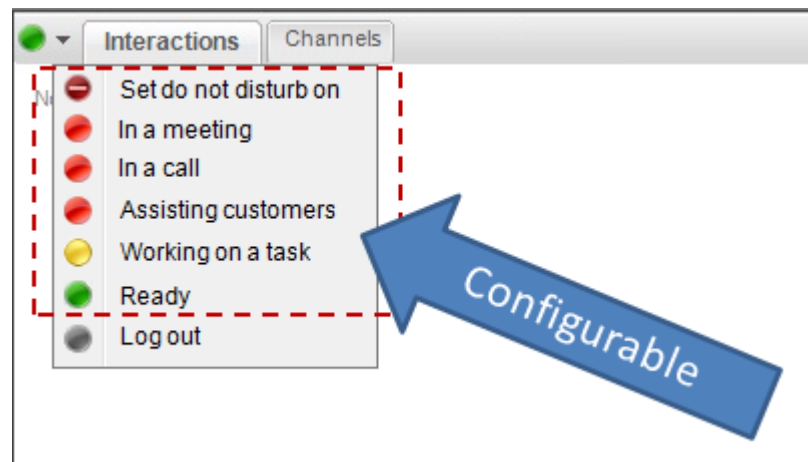
Figure 17: Customized menu options.

The text representation of the custom state can be defined in several languages simultaneously. UC Connector transmits localized resources to the user's browser to be selected based on local preferences.

When T-Server reports the agent state event, UC Connector matches the event against a set of defined agent states. When the reported state is matched, UC Connector updates the corresponding status indicators in the main panel and Channels tab of the UC Connector web client.

To display a non-matching agent state, UC Connector uses one of the predefined states that have no matching elements defined. For example, UC Connector receives an event with the following:

• Agent state: Not Ready

• Agent mode: After Call Work

• Reason code: 101

If UC Connector cannot match the reason code, it might use a definition for the Not Ready state with an After Call mode and no reason defined. If it cannot match the reason code and the state, it might use a definition for a plain Not Ready state without a mode.

UC Connector uses built-in plain Ready, Not Ready (previously called "DND on" and "DND off"), and DND state definitions for states that cannot be matched against the XML file.

The system integrator is responsible for defining all the states that must be represented in the UC Connector web client.

> **Note:** Some states, such as Not Ready with a work mode Legal Guard, cannot be requested because they are controlled exclusively by T-Server or the PBX. Other states, such as Not Ready with a work mode After Call Work, might not be compatible with a particular T-Server or PBX. UC Connector does not validate the state definition. If the state is defined as an available web client menu item, UC Connector attempts to use it as defined, but the operation may fail if the target state is not supported by the T-Server.
>
> The states that are controlled by T-Server and cannot be executed as an agent command can still be defined in the XML document so that the corresponding status can be displayed visually.

Multiple Channels

Users may have multiple channels provisioned in their place, but the UC Connector web client only has one combined agent state indicator.

When a state change command is issued from the menu, UC Connector attempts to apply the same agent state request to all channels. If all the resulting channel states map to the same defined state, UC Connector displays and icon corresponding to that state. If the resulting channel states map to different custom states, UC Connector updates the main agent status indicator in the web client with one of the following predefined states:

`partial-ready`—at least one of the channels is in the "ready" state

`not-ready`—at least one channel is in the "logged in" state

`logout`—all channels are logged out

> **Note:** The indicator for an individual channel reflects the state of the channel only if the icon for the badge is supplied.

How It Works—External Number Redirect

A Knowledge Worker or an Administrator can enable an external redirect number, which allows Knowledge Workers to accept preview calls at the specified number.

When this features is enabled, the preview window has a button to accept the incoming call on an external number.
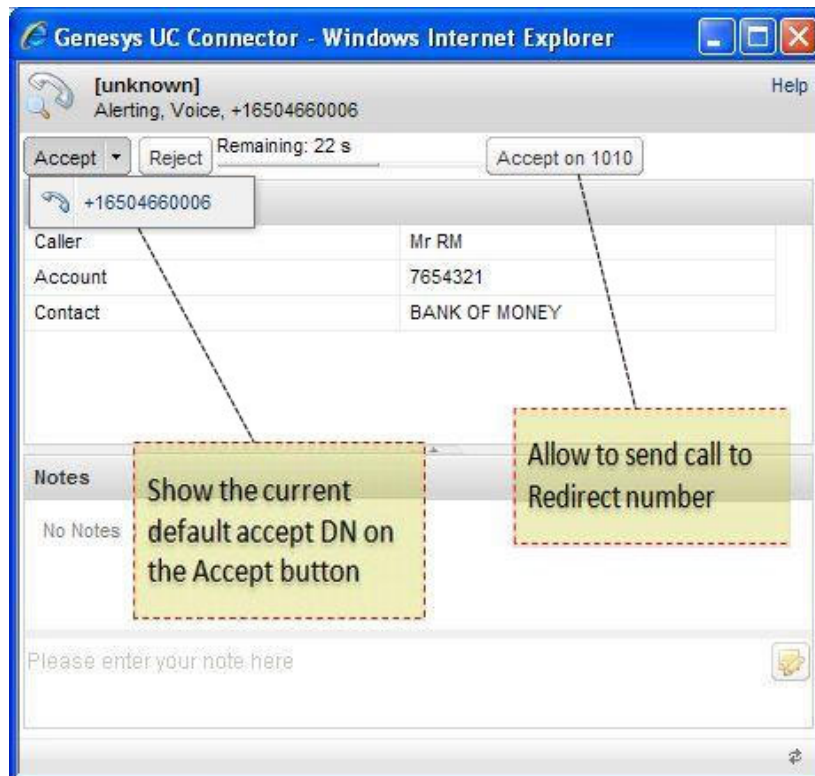
Figure 18: Accepting a preview on an external number.

If the user clicks the external number button, Universal Routing Server (URS) routes the call to the specified redirect number. When the preview is accepted

on an external number, the window is modified to indicate the call is on an external number.
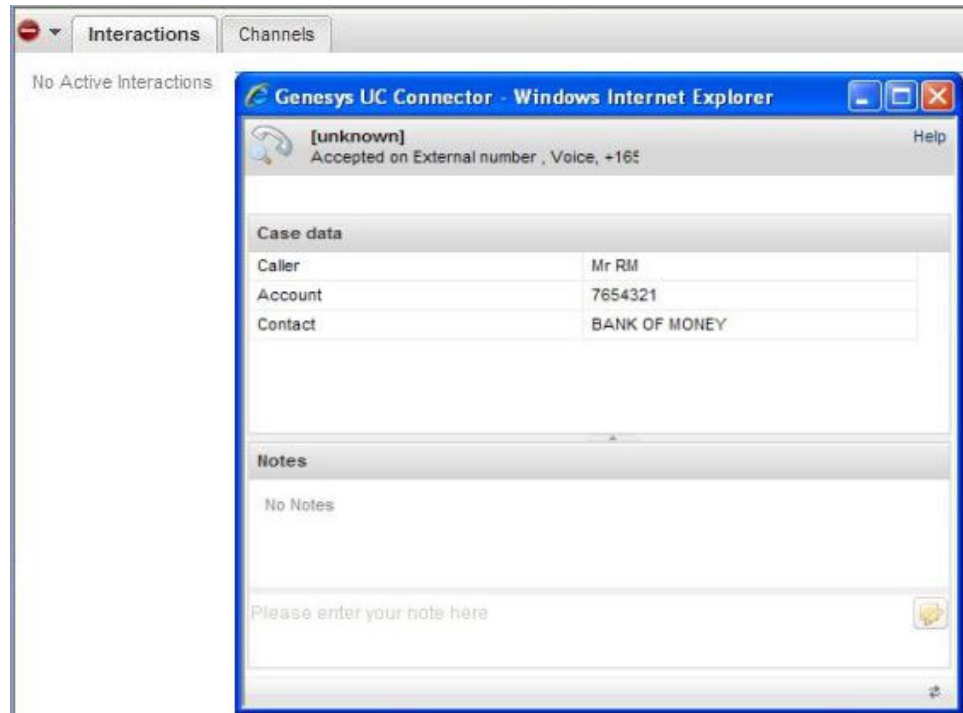
Figure 19: Preview accepted on an external number.

If provisioned in the presence definition XML document, see "How It Works—Customized Knowledge Worker States" on page 49, UC Connector sets the user to the "preview-redirect" auto-state after the redirect. The default target state after the redirect is "dnd". Remote call control is not possible at this point, and call notes (user data) cannot be edited or added.

When the user is ready to take the next call, they must first close the preview window and set the user status to "ready".

Configuring an external number

To configure an external number for preview, the user must select the Setup menu option.
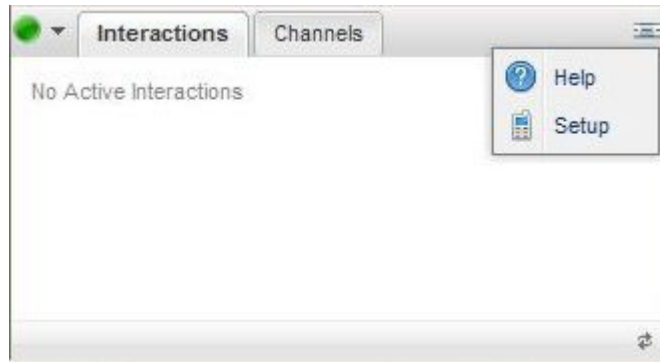
Figure 20: Setup menu option.

This menu option will only appear if the redirect-setup-enabled configuration option is set to true. For configuration details for this feature, see "Enabling a Redirect Number" on .

Selecting the Setup menu option opens the Setup Redirect Number window, where the user can set or change the redirect number.

Figure 21: Setup Redirect Number window.

It is possible for an Administrator to enable and set the redirect number for a person through Genesys Administrator. See "Enabling a Redirect Number" on .

Note: The redirect number is not validated against premium rate, international numbers and so on. It is the responsibility of the Administrator and the telephony platform to validate if the user has permissions to redirect to the requested number.

Genesys

Chapter

3   Supported Integrations

The UC Connector supports Genesys contact center integrations with a variety of third-party Unified Communications (UC) solutions. This chapter describes the differences in supported functionality between the different UC platforms, as well as basic information about the platforms themselves.

- Functional Support, page 57
- Microsoft Office Communication Server, page 58
- IBM Sametime 8.5, page 62

Note: The references to third-party documentation in this section, including any URL or other web references, are subject to change without notice. They are included for your convenience.

Functional Support

Table 7 lists the main differences in functionality between integrations with Microsoft OCS (Lync, OCS 2007 R2) and with IBM Sametime 8.5.

Table 7: Functions Supported by the UC Solution

UCC Function	Microsoft Lync	Microsoft OCS 2007 RN	IBM Sametime
IM through SIP Server-OCS Integration	yes	yes	no
IM through Genesys IM Integration	yes	yes	yes
Presence Monitoring	yes	yes	yes

Table 7: Functions Supported by the UC Solution (Continued)

UCC Function	Microsoft Lync	Microsoft OCS 2007 RN	IBM Sametime
Custom Presence States	yes	yes	no
Voice channels	yes	yes	yes
Overflow	yes	yes	yes
Preview Mechanism	yes	yes	yes
Push on-call status	yes	yes	yes
Custom tab[a]	no	yes	yes
Secure connections (TLS)	yes	yes	yes

a. Custom tab requires direct UC Connector integration with the UC client, so that interactions can be accepted and handled using native UC client capabilities. If the UC platform does not support custom tab integration, interactions are processing using the Knowledge Worker's web browser instead.

Microsoft Office Communication Server

The UC Connector provides the connection between the Genesys environment and the Microsoft OCS (Lync, OCS 2007 R2) solution in the enterprise. For presence, communication between UC Connector and OCS is through MS-PRES. For a complete description of MS-PRES see
`http://msdn.microsoft.com/en-us/library/cc431501%28v=office.12%29.aspx`

The UC Connector does not play an active role in instant messaging scenarios with OCS. These are handled between OCS and SIP Server exclusively, over SIP.

For integration with the Microsoft Office Communicator (MOC) client, the UC Connector uses a custom web application. This web application provides a custom tab in the MOC 2007 R2 client, providing controls over the user's presence and any current interaction. Communication between UC Connector and MOC is through HTTP. The same function is implemented with Lync using the conversation Extensibility window, or directly in a browser window on the user desktop. This is because Lync does not support a custom tab in its user interface.

Supported deployments with Microsoft OCS include:

• "Deployment with Microsoft OCS Standard Edition"
• "Deployment with Microsoft OCS Enterprise Edition"
• "Deployment with Microsoft OCS Enterprise via Edge Server"

Deployment with Microsoft OCS Standard Edition

Figure 22 shows a UC Connector integration with Microsoft OCS Standard Edition.
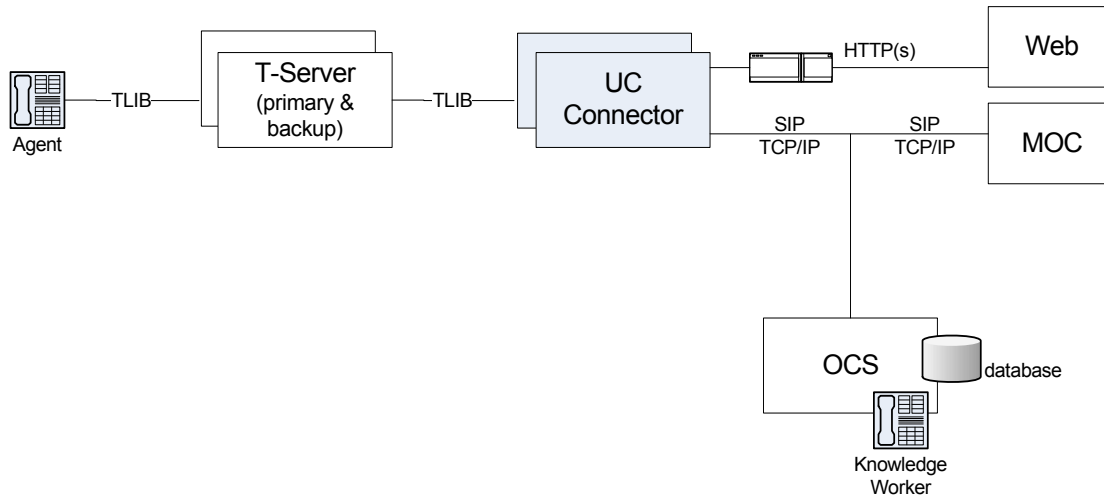
Figure 22: Deployment with Microsoft OCS Standard Edition

Integrations with the Standard Edition of Microsoft OCS require that the main server components, as well as the database for storing user and conference information, are deployed on a single Front End Server. This integration is used for organizations with under 5000 users, and which does not require High Availability through clustering for the OCS part of the system.

Database

For the Standard Edition, the real-time communications (RTC) database must be kept locally on a Microsoft SQL Server Express instance.

Deployment with Microsoft OCS Enterprise Edition

Figure 23 shows a UC Connector integration with Microsoft OCS Enterprise Edition, where OCS is deployed on multiple servers, the database is deployed

on a separate server, and a third-party load balancer is deployed to balance the load across the OCS Front End servers.
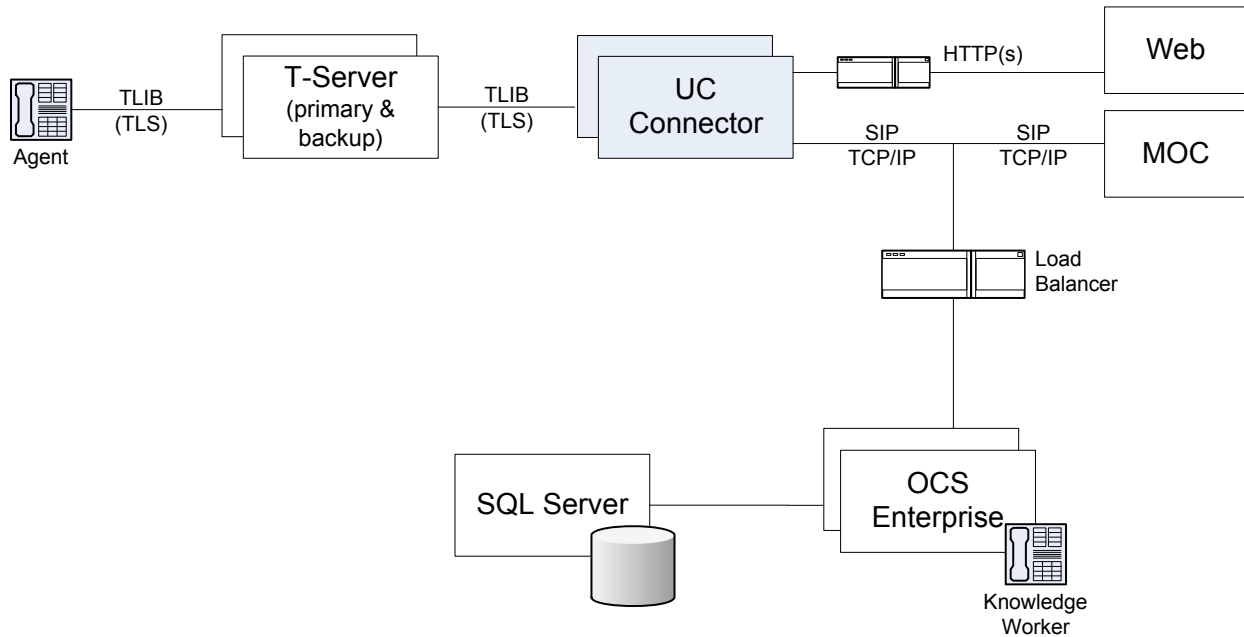
Figure 23: Deployment with Microsoft OCS Enterprise Edition

With the Enterprise Edition, you can separate the server functions from the database functions as a way to increase capacity and availability. This edition is recommended for organizations that require higher availability through clustering of server roles.

Pools

The Enterprise Edition supports multiple pools, with each pool consisting of one or more Front End Servers that provide IM, presence, and conferencing services and are connected to a separate dedicated Microsoft SQL Server back-end database.

Database

The Enterprise Editions enables you to specify a remote database server. This dedicated Microsoft SQL Server back-end database must be located on a computer that is separate from any of the Enterprise Edition servers.

Deployment with Microsoft OCS Enterprise via Edge Server

Figure 24 shows a UC Connector integration with Microsoft OCS Enterprise Edition, where OCS is deployed on multiple servers, the database is located on a separate server, and a Microsoft OCS Edge Server is deployed in front of the primary OCS servers.
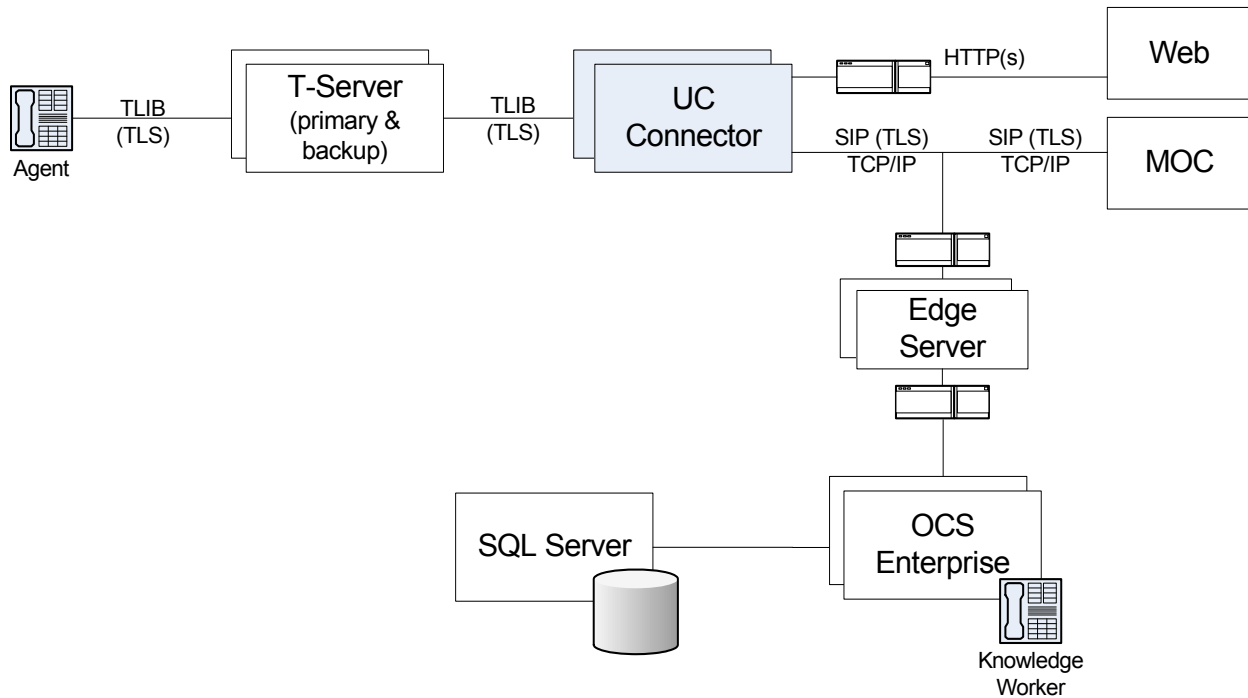
Figure 24: Deployment with Microsoft OCS Enterprise Edition Through Edge Server

About the Edge Server

Installed at the perimeter of the enterprise network where the OCS servers are located, the edge server is used to authorize users from outside of the enterprise firewall before they can access the OCS deployment.

For more information about deployments that use the OCS Edge Server, see the Microsoft documentation for the product:

- `http://www.microsoft.com/downloads/details.aspx?familyid=E9F86F96-AA09`
  `-4DCA-9088-F64B4F01C703&displaylang=en`

Reporting in Microsoft OCS Integrations

When integrated with Microsoft OCS, the four standard Microsoft Office Communicator presence states are mapped to Knowledge Worker-specific `AttributeReason` parameters. These `KW_UC_STATUS` parameters are used to provide Genesys Reporting with additional information about routing requests involving Knowledge Workers.

Table 8 shows the mapping between Microsoft OCS presence states and Genesys `AttributeReason` parameters.

Table 8: Presence Mapping to AttributeReason

Communicator Presence	KW_UC_STATUS
Busy	`RequestAgentNotReady` with `KW_UC_STATUS` of `busy`.
Do Not Disturb	`RequestAgentNotReady` with `KW_UC_STATUS` of `dnd`.
Be Right Back	`RequestAgentNotReady` with `KW_UC_STATUS` of `be-right-back`.
Away	`RequestAgentNotReady` with `KW_UC_STATUS` of `away`.
Available	`RequestAgentReady` with `KW_UC_STATUS` of `ready`.

IBM Sametime 8.5

The UC Connector provides the connection between the Genesys contact center and the IBM Sametime 8.5 platform in the enterprise. For presence, communication between UC Connector and IBM Sametime is through a proprietary IBM protocol, implemented by the Sametime SDK built into the UC Connector application. For integration with the Sametime Connect clients, the UC Connector deployment creates a custom Contact Center plug-in for each Sametime Connect client. This plug-in displays a Contact Center panel in the client, which the Knowledge Worker can use to access information about contact center resources, as well as where a call can be transferred. Communication between UC Connector and the Sametime Connect client is through HTTP.

Deployment with IBM Sametime Entry/Standard Edition

Figure 25 shows a sample UC Connector integration with IBM Sametime Entry or Standard Edition.
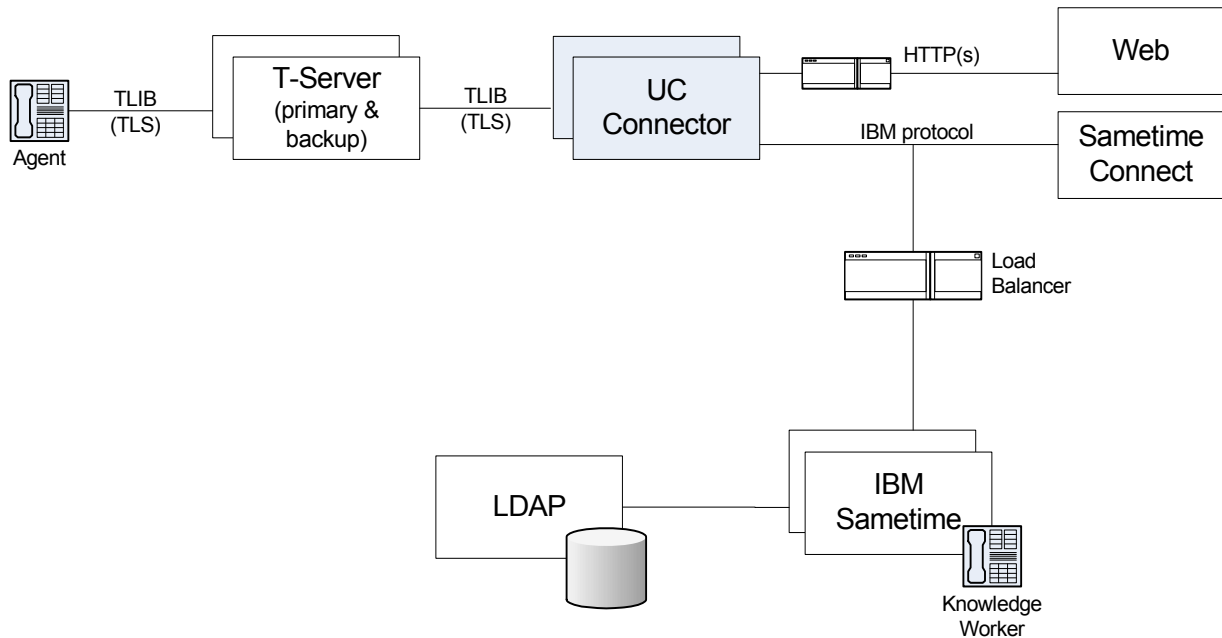
Figure 25: Deployment with IBM Sametime Entry/Standard Edition

Database

Sametime can be configured to operate as a client to a Lightweight Directory Access Protocol (LDAP) server containing an LDAP directory. You can use this LDAP directory as your Knowledge Worker user repository.

Clustering

Multiple Sametime servers can be configured to operate together, in order to support failover and load balancing for large user populations.

Entry Versus Standard Editions

Sametime Entry has less functionality than Standard, but is not limited in terms of scalability. For integrations with UC Connector, there is no practical difference between these two editions. All UC Connector functionality is supported for both Entry and Standard Editions.

Reporting in IBM Sametime 8.5 Integrations

When integrated with IBM Sametime 8.5, the standard Sametime Connect presence states are mapped to Knowledge Worker-specific `AttributeReason` parameters. These `KW_UC_STATUS` parameters are used to provide Genesys Reporting with additional information about routing requests involving Knowledge Workers.

Table 9 shows the mapping between Sametime Connect presence states and Genesys `AttributeReason` parameters.

Table 9: Presence Mapping to AttributeReason

Sametime Connect Presence	KW_UC_STATUS
Away	`RequestAgentNotReady` with `KW_UC_STATUS` of "I am away from the computer."
In a Meeting	`RequestAgentNotReady` with `KW_UC_STATUS` of "I am in a meeting."
Do Not Disturb	`RequestAgentNotReady` with `KW_UC_STATUS` of "Please do not disturb me."
Available	`RequestAgentReady` with `KW_UC_STATUS` of "I am available".

Note: The text for these `KW_UC_STATUS` messages are configurable from the Sametime Connect client. The above messages are the default messages provided by Sametime Connect.

2

UC Connector Deployment

The following chapters describe the prerequisites and procedures that are required to deploy the UC Connector into the Genesys environment, as well as how to integrate the deployment with the third-party Unified Communications (UC) platform on the enterprise side:

- Chapter 4, "Deployment Prerequisites," on
- Chapter 5, "Deploying UC Connector," on
- Chapter 6, "Configuring the Routing Strategies," on
- Chapter 7, "UC Connector High Availability Deployment," on

Deployment Overview

Complete the following tasks to deploy and integrate the UC Connector solution.

Task Summary:
Configuring the UC Connector Solution

Objective	Actions
1. Complete prerequisites.	• Configuring the Baseline Genesys Environment • Installing JAVA • Host Requirements • Deploying the Third-party UC Platform

Task Summary:
Configuring the UC Connector Solution (Continued)

Objective	Actions
2. Deploy the UC Connector.	1. Verify the baseline Genesys configuration. 2. Create Host. 3. Import the application template. 4. Create the UC Connector Application object. 5. Install the UC Connector. 6. Configure the UC Connector section. 7. Configure the Log section. 8. Configure the third-party UC section.
3. Configure the Knowledge Worker.	1. Create KW DNs. 2. Create Agent Logins. 3. Configure Places. 4. Create KW Persons.
4. Integrate with the agent's desktop.	• Integrating with Interaction Workspace
5. Integrate with the third-party UC.	• Integrating with Microsoft Lync Server • Integrating with Microsoft OCS 2007 R2 • Integrating with IBM Sametime 8.5
6. Integrate with Genesys Routing	1. Create a "dummy" Custom Server Application object. 2. Connect UC Connector to Custom Server. 3. Configure routing from contact center to KW. 4. Configure a contact point for the KW.
(optional) Enable Instant Messaging (Genesys IM Integration)	1. Verify the Genesys IM solution. 2. Configure the Knowledge Worker DN.
(optional) Enable Instant Messaging (SIP-OCS Integration)	1. Install and configure SIP Server. 2. Integrate SIP Server with OCS.
(required for Lync) Enable Secure Communication	• Enabling MTLS • Enabling TLS/Kerberos

![Genesys logo]

4 Deployment Prerequisites

Before you begin the UC Connector integration, all baseline components must be installed and configured. For Genesys components, consult the product *Deployment Guides* as outlined in the section below. For the enterprise UC solution-related components, consult the third-party product documentation described in the section below.

This chapter contains the following sections:

Configuring the Baseline Genesys Environment

The following table lists the prerequisite Genesys components, their respective Deployment Guides, as well as key actions that you must complete before starting the integration procedures.

Table 10: Configuring the Baseline Genesys Environment

Component	Key Actions	Documentation
Management Framework 7.5+	A centralized Genesys Management Framework, with all required components, must be installed.	*Framework 8.1 Deployment Guide*
Genesys Administrator	After installing Genesys Administrator, login to the tool: 1. Open a web browser and enter the following URL: `http://<computer name>/wcm/Default.aspx` 2. Enter the following information: • User name • User password • Application (name of the Configuration Server in database) • Host name (Configuration Server host) • Port (Configuration Server port) 3. Click `Log in` **Note:** Genesys Administrator or Configuration Manager can be used to configure UC Connect.	*Framework 8.1 Genesys Administrator Deployment Guide*
Configuration Manager	On a computer installed with Configuration Manager, login to the tool: 1. `Start > Programs > Genesys Solutions > Framework > Configuration Manager > Start Configuration Manager` 2. Enter the following information: • User name • User password • Application (name of Configuration Server in database) • Host name (Configuration Server host) • Port (Configuration Server port) 3. Click `Ok`. **Note:** Genesys Administrator or Configuration Manager can be used to configure UC Connect.	*Framework 8.1 Deployment Guide*

Table 10: Configuring the Baseline Genesys Environment (Continued)

Component	Key Actions	Documentation
T-Server	Required for integration with various switches or PBXs. **Note**: The T-Server and associated switch MUST support emulated-agent functionality for integration with UC Connector.	See the Deployment Guide for your respective T-Server. For more information on which T-Servers support emulated-agent functionality, see Appendix C, "T-Server Compatibility with UC Connector," on page 213.
SIP Server version 8.0.300.00 or later	Required for the following deployments: • Instant Messaging functionality with Microsoft OCS	*Framework 8.1 SIP Server Deployment Guide*
Universal Routing Server	Required for routing to agents and for handling the contact center-to-knowledge worker interaction. Required connections: • Message Server • SIP Server `Application` object (if included) • Any other T-Server Application that is included your deployment. • Stat Server • Custom Server	*Universal Routing 8.1 Deployment Guide* *Universal Routing 8.1 Reference Manual*
Interaction Routing Designer	Required for building the URS routing strategies that control the interaction between contact center and knowledge workers.	*Universal Routing 8.1 Business Process User's Guide* *Universal Routing 8.1 Interaction Routing Designer Help*

Table 10: Configuring the Baseline Genesys Environment (Continued)

Component	Key Actions	Documentation
Interaction Workspace 8.1 (or pre-8.0 Genesys Desktop)	Required for a desktop workspace that the contact center agent uses to initiate interactions with the Knowledge Worker. For Interaction Workspace 8.1, required functionality is included in default deployment. For Genesys Desktop, some additional configuration may be required. See "Integrating with Interaction Workspace" on page 100	*Interaction Workspace 8.1 Deployment Guide*
Stat Server 7.6+	Required for monitoring the availability of agents and emulated agents (knowledge workers) targeted in the routing strategies. Required connections: • Message Server • SIP Server `Application` object (or T-Server)	*Framework 8.1 Stat Server Deployment Guide*

Installing JAVA

UC Connector requires that you install a Java Software Development Kit (JDK). UC Connector support JDK 7.0 (release 1.7.0_06+).

The following table provides basic information about installing the JDK on the host Windows Server.

Task Summary:
Installing JDK 7.0 on the Windows Server

Objective	Key Actions
1. Download the JDK version 1.7.	Download the JDK 7.0 (release 1.7.0_06+) to your host computer (see "Operating System—UC Connector Host"). At publication of this document, JDK 7.0 is available for download here: `http://www.oracle.com/technetwork/java/javase/downloads/index.html`

Task Summary:
Installing JDK 7.0 on the Windows Server (Continued)

Objective	Key Actions
2. Configure Environment Variables.	Make sure the `JAVA_HOME` environment variable points to the directory where you installed the JDK—for example: `c:\jdk1.7.0`

Host Requirements

Operating System—UC Connector Host

For installation of the UC Connector application on the host computer, UC Connector 8.0 currently supports the following operating systems:

- Windows Server 2003, 32-bit and 64-bit
- Windows Server 2008, 32-bit and 64-bit
- Windows Server 2008 R2, 64-bit

Web Browser—Knowledge Worker Host

UC Connector 8.0 integrates with the web browser on the host Knowledge Worker computer. UC Connector currently supports integration with the following web browsers:

- Windows Internet Explorer 8.0, 9.0
- Firefox 3.5 and higher
- Safari 5

Deploying the Third-Party UC platform

The installation and deployment of the third-party UC platform required for a UC Connector integration is beyond the scope of this document. If you are performing an end-to-end deployment (the UC platform is not already installed), the following table details where you can go for more information about setting up the UC platform

Note: The references to third-party documentation in this section, including any URL or other web references, are subject to change without notice. They are included for your convenience.

Task Summary: Deploying the Third-party UC Platform

UC Platform	Key Actions or Info

Microsoft OCS	**Deploying the Server**
	
	The Microsoft OCS Servers must be installed and configured according to the required topology. This may include:
	
	• Load balancers for handling traffic.
	• Edge Servers for authorizing Genesys contact center access to the enterprise UC platform.
	
	For details, consult the following Microsoft product documentation:
	
	Microsoft OCS 2007 R2:
	
	`http://www.microsoft.com/downloads/details.aspx?FamilyID=e9f86f96-aa09-4dca-9088-f64b4f01c703`
	
	Microsoft Lync Server:
	
	`http://technet.microsoft.com/en-ca/library/gg398616.aspx`
	
	Deploying the client (Office Communicator)
	
	Office Communicator must be installed on the Knowledge Worker device(s). For Microsoft OCS 2007 R2, additional configuration for the custom "Contact Center" tab in the client is required. For details, see the Procedure: Modifying the Custom Tab File, on page 108.
	
	Microsoft Lync does not support a custom tab; however, there are some registry keys that must be set in order to enable launching the UC Connector GUI in a browser window from the Lync client. For details, see the Procedure: Modifying the Registry for Microsoft Lync, on page 102.
	
	Secured Mode
	
	For Microsoft Lync Server, secured communication is mandatory. For details see Task Summary: Enabling TLS/Kerberos Secure Communication, on page 140.
	
	Supported Architectures
	
	For information about supported OCS architectures, see "Microsoft Office Communication Server" on page 58.

Task Summary: Deploying the Third-party UC Platform (Continued)

UC Platform	Key Actions or Info
IBM Sametime	The IBM Sametime server architecture and Sametime clients must be installed and configured before you begin the integration with the UC Connector. **IBM Sametime 8.5.x Documentation** The latest version is available at `http://www.ibm.com/developerworks/lotus/documentation/sametime/index.html` **Supported Architectures** For information about supported IBM Sametime architectures, see "IBM Sametime 8.5" on page 62.

Chapter

5 Deploying UC Connector

This chapter describes how to deploy the UC Connector, how to integrate the UC Connector into the rest of the Genesys environment, and how to integrate the UC Connector deployment with the third-party UC solution.

This chapter includes the following sections:

Deploying the UC Connector

The following table provides an overview of the main steps that you must complete in order to deploy the UC Connector into the Genesys environment.

Task Summary: Deploying the UC Connector

Objective	Actions
1. Verify the baseline Genesys configuration.	If the UC Connector is deployed together with a Genesys Contact Center installation, check that the following prerequisite components are deployed: • Management Framework (LCA, Config Server, Message Server) • Universal Routing Server • Stat server • T-Server • SIP Server (for Instant Messaging scenarios) • Interaction Workspace (optional. Other agent desktop clients can also be used) If the UC Connector is used without a Contact Center, install the following components that come with the UC Connector bundle. The components should be installed within the same subnet. Refer to the respective deployment guides for installation information. • Management Framework (LCA, Config Server, Message Server) • Universal Routing Server • Stat server • The appropriate T-Server for the deployed PBX • SIP Server (for Instant Messaging scenarios) For details, see Table 10, "Configuring the Baseline Genesys Environment," on page 68.
2. Create Host.	Create a `Host` object for the computer on which you will later install the UC Connector on (if one has not been created already). **Supported Operating Systems:** • Windows Server 2003 32/64 bit • Windows Server 2008 32/64 bit • Windows Server 2008 R2 64 bit For more information about creating hosts, see the *Framework 8.1 Deployment Guide*.

Task Summary: Deploying the UC Connector (Continued)

Objective	Actions
3. Import the application template.	1. Go to `Environment` > right-click `Application Templates` > `Import Application Template` 2. Select the `UC_Connector_800.apd` template available on the product CD. For more information, see the *Framework 8.1 Deployment Guide*.

4. Create the UC Connector Application object.	1. Create the UC Connector object from the imported .apd template.
	
	2. Add a SIP listening port—typically `5060` (required for integration with OCS only).
	
	3. Add connections to:
	  ◆ Custom Server (see "Integrating with Genesys Routing" on page 119)
	  ◆ T-Server
	  ◆ Stat Server
	
	For more detailed instructions, see Procedure: Creating the UC Connector Application object, on page 79.

5. Install the UC Connector.	On the host computer, launch the `setup.exe` file available on the product CD. **Note**: The installation path must *not* include any spaces. Genesys recommends installing to the default path: `C:\GCTI\UCConnector` **Key Action** • Take note of the HTTP port that you enter in the installation wizard. You will have to specify this port in web page customization later in the deployment. For more detailed instructions see, Procedure: Installing the UC Connector server on the host, on page 81.
6. Configure the UC Connector section.	Use the options in the `UC-Connector` section to enable notes, customize interaction windows, configure timeouts, and other features. None of these options are mandatory (default values are acceptable; some features may not be enabled). For a detailed list of `UC-Connector` options, see Procedure: Configuring UC Connector options. **Note**: DN/Switch-level settings (configured on the KW `Person` object) take precedence.

Task Summary: Deploying the UC Connector (Continued)

Objective	Actions
7. Configure the Log section.	In the `Log` section, configure the log-related options as you would for any other Genesys application. There is one UC Connector-specific log option: · `internal` For more information, see the *Framework 8.1 Deployment Guide*.
8. Configure the third-party UC section.	Depending on your third-party UC platform, configure one of the following sections: · `Microsoft-OCS` · `IBM-Sametime` For details, see Procedure: Configuring UC Connector options. **Note**: DN/Switch-level settings (configured on the KW `Person` object) take precedence.
Optional Customization	
• Enable Help Buttons	You can enable a help button on various UC Connector screens, with links to a customized help file. For details, see "Customizing the Help Buttons" on page 151.
• Set Default Language	You can set the default language for the UC Connector user interface. For details, see "Changing the Default Language" on page 152.
• Enable Automatic Log-in	To automatically log in all Knowledge Workers on UC Connector start up: • In the `uc-connector` section of the UC Connector `Application` object, set the option `user-auto-registration` to `true`. **Note**: This is required for integrations with Microsoft Lync Server.
• Enable Logout Menu	To display the logout menu in the GUI: In the `uc-connector` section of the UC Connector `Application` object, set the option `enable-logout-menu` to `true`.
• Enable Audio on Preview or Ringing	You can enable an audio file to play when the Preview or Ringing pop-up window is displayed. For details, see "Enabling Audio on Preview or Ringing" on page 153

Task Summary: Deploying the UC Connector (Continued)

Objective	Actions
• Configure Hotkeys for Interaction Preview	You can set specific keys to accept or reject a call when the Preview window is displayed. For details, see "Configuring Hotkeys for Interaction Preview" on page 154.
• Customize Agent Sates	You can define custom presence states for the UC Connector Web Client with an XML resource file. For details, see "Customizing Knowledge Worker States" on page 155.
• Enable Number Redirect	You can allow agents to change their own external redirect number and accept preview calls on that number. For details, see "Enabling a Redirect Number" on page 158.

Procedure:
Creating the UC Connector Application object

Prerequisites

- The host object has been created for the machine where you want to install UC Connector. For details, see the "Create Host." section of the table Task Summary: Deploying the UC Connector.

- The `UC_Connector_800.apd` is imported into the `Application Templates` folder. For details, see the "Import the application template." section of the table Task Summary: Deploying the UC Connector

- You are logged into Configuration Manager.

Start of procedure

1. Go to `Environment`, right-click the `Applications` folder, and select `New` > `Application`.

2. In the `Browse` window, select the `UC_Connector_800.apd` that you created in Step 3.

3. On the `General` tab of the new object window, give the UC Connector a `Name`.

4. On the `Server Info` tab, browse to select the `Host` that you created in Step 2.

5. (*Required for Microsoft OCS integrations only*). In addition to the default listening port, click `Add Port` to add an additional SIP listening port for communication with the third-party UC platform.

- `Port ID`—Enter a useful name for this port.
- `Communication Port`—Enter the SIP communication port to be used. Typically `5060` is used for SIP ports.
- `Connection Protocol`—Select `sip` from the drop-down list.
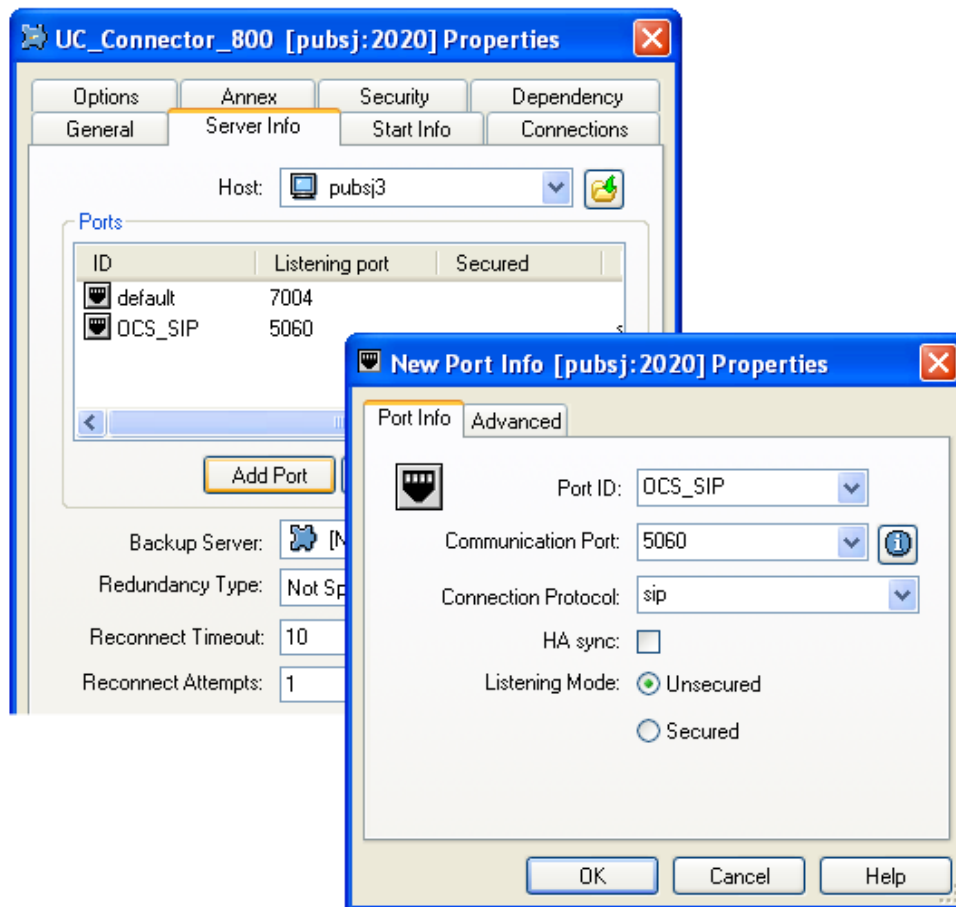
Figure 26: Creating the SIP Port for Microsoft OCS

6. On the `Connections` tab, add connections to:
   - T-Server/SIP Server—To handle agent and call control.
   - Stat Server—To present real time information about resources.
   - Message Server—To handle log events.
   - Custom Server—To handle the Preview Interaction.

      Tip: Custom Server, which links UC Connector to the Universal Routing Server, might not be created yet. You can add the connection later if required.

7. On the `Start Info` tab, type a period (.) in the `Working Directory`, `Command Line`, and `Command Line Arguments` text boxes.

   The information will be filled in automatically when you install UC Connector.

8. Click `OK`.

End of procedure

Next Steps

- Procedure: Installing the UC Connector server on the host

Procedure:
Installing the UC Connector server on the host

Prerequisites

- A Windows Server 2003/2008/2008 R2 computer on which to install the server.
- A UC Connector `Application` object is configured, according to Procedure: Creating the UC Connector Application object.
- JDK 6.0 is installed on the host computer. For details, see Task Summary: Installing JDK 7.0 on the Windows Server, on page 70.
- A valid license file placed on a network-accessible location.

Start of procedure

1. On the UC Connector 8.0 product CD, locate the `setup.exe` file and double-click to start the Genesys Installation Wizard.

> **Tip:** Click the `About` button to see the `read_me` file. This file also provides a link to the product Release Note.

2. Follow the Wizard instructions, clicking `Next` through each of the following pages:

a. `Connection Parameters to the Genesys Configuration Server`—Enter host name, port, user name, and password for Configuration Server.

b. `Select Application`—Select the name of the UC Connector `Application` object that you created in Step 4.

c. `Access to License`—Select one of the following options:
- `License Manager`—You want your server application to use host name and port number parameters to connect to the license server. In this instance, you must enter values for the host and the port of the license server.
- `License File`—You want your server application to retrieve license server information from the license file. Use the `Browse` button to navigate to the license file.

d. `Choose Destination Location`—UC Connector requires that the path to the installation directory has no blank spaces. Genesys recommends accepting the default path:

`C:\GCTI\UCConnector\[CME_Application_Name]`

e. `Select Installed Sun's Java Development Kit (JDK)`—Select the JDK version installed for the UC Connector deployment. This should be the same JDK as the one you installed in "Installing JAVA" on

f. `User Parameters`—Select HTTP or HTTPS. The installer then asks for the HTTP port. Enter the HTTP port to be used for HTTP communication between the UC Connector and the web page for the custom `UC Connector` tab in the UC client.

Notes: Take note of this HTTP port number. You will need to input this later when customizing UC client for integration with UC Connector.

For High Availability deployments, use the Windows NLB Virtual IP address for the Host. Do this for both primary and backup UC Connector instances. You must also use the same port number for both instances. For details, see Chapter 7, "UC Connector High Availability Deployment," on .

---

g. `Unified Communication Server Options`—Select your UC platform type: `Microsoft-OCS` or `IBM-Sametime`.

h. `Unified Communication Server Configuration`—Depending on your UC platform, enter the following parameters:

Table 11: UC Platform-Specific Installation Parameters

UC Platform	Parameter	Value
Microsoft-OCS	Contact	Enter the SIP URI configured in Microsoft OCS for the UC Connector. For OCS, this is the "principal" assigned to the UC Connector application. For example, `sip:ocs-ucc@your-ocs-address.com` This principal represents the UC Connector in Microsoft OCS.
	Registrar-uri	Enter the URI that UC Connector uses to connect to Microsoft OCS. For example, `sip:pool01.your-ocs-address.com`

**Table 11: UC Platform-Specific Installation Parameters
 (Continued)**

UC Platform	Parameter	Value
IBM-Sametime	Server-fqdn	Enter the fully qualified domain name (FQDN) for the IBM Sametime server host computer.
	Server-port	Enter the port number for the IBM Sametime server.

    i. `Ready to Install`—Click `Install` to proceed.

3. In the final `Installation Complete` page, click `Finish`.

End of procedure

Next Steps

- Procedure: Configuring UC Connector options

Procedure:
Configuring UC Connector options

Prerequisites

- You are logged in to Configuration Manager.
- A UC Connector `Application` object, configured according to Procedure: Creating the UC Connector Application object.
- Genesys recommends that you create an account/user to represent the UC Connector in Microsoft OCS—for example, `ocs-ucc`.

Start of procedure

1. Go to `Environment > Applications >` and double-click the UC Connector `Application` object.

2. Go to the `Options` tab.

3. In the `UC-Connector` section, configure the following options.

Table 12: UC Connector—UC-Connector Section

Option Name	Default Value	Description
chat-title	Chat	Enter the name you want to appear in the regular chat window.
chat-consult-title	Consulting Chat	Enter the name you want to appear in the window that appears for consultation chat interactions.
presence-gateway-mode	false	If enabled, UC Connector does not send invite messages to Lync when there is no web client UI connected for the corresponding user, and a new call or preview call is delivered. If false, sending the invite is controlled by the invite-message option.
preview-itx-arrival-timeout	9000	UC Connector waits 9 seconds (9000 ms) after the Preview Notification is accepted for the interaction to arrive.
preview-shortkey-accept	Blank	If blank, the keyboard hotkey is disabled. Otherwise, pressing the configured key accepts the call when the interaction Preview window is displayed.
preview-shortkey-reject	Blank	If blank, the keyboard hotkey is disabled. Otherwise, pressing the configured key rejects the call when the interaction Preview window is displayed.
preview-expiration-timeout	15000	UC Connector waits 15 seconds (15000 ms) for a response from the target of the Preview Notification. If no response, the target user is set to `NotReady`. A countdown timer in the preview window shows how much time is remaining.
login-queue	No default Value	Users logging in through the Custom Tab of the UC client are logged into the specified ACD Queue.

Table 12: UC Connector—UC-Connector Section (Continued)

Option Name	Default Value	Description
dnd-off-timeout	300000	If the user declines the Preview Notification, UC Connector sets the user to `NotReady` for 300000 milliseconds (5 minutes).
enable-logout-menu	false	Specifies whether the logout menu item is displayed in the UC Connector GUI.
enable-preview-reporting	false	Controls whether UC Connector creates records in ICON for the interaction Preview events `Accepted`, `Rejected`, `Timeout`, `Taken`, `Error`, `Cancel`.
preview-state-name	3721, UCC_Preview	Specifies the numeric identifier and key name of the custom state associated with the UC Connector interaction Preview offer for the record in ICON.
presence-gateway-mode	UCC_ConnId,UCC_UserId, UCC_AgentId,UCC_Reason	Specifies the ordered list of key names used for reporting the UC Connector Preview offer parameters in ICON.
userdata-preview-format<n>	title: <UserData_DisplayName>, value: [UserData_Key]	UC Connector displays the value of the configured `UserData_Key` in a field which will have the display name as configured in `UserData_DisplayName`.
userdata-call-format<n>	title: <UserData_DisplayName>, value: [UserData_Key]	UC Connector displays the value of the configured `UserData_Key` in a field which will have the display name as configured in `UserData_DisplayName`.
userdata-contact-format	title: Contact, value: [USER-ID]	UC Connector displays the value of the configured `USER-ID` key under the field `Contact, value:` field. **Note**: This value is not displayed in the interaction, but is required for a valid configuration.
popup-udata-key	Blank	UC Connector shows the call control window if the option value is blank. Otherwise, UC Connector shows the call control window for calls with the specified User Data Key.

Table 12: UC Connector—UC-Connector Section (Continued)

Option Name	Default Value	Description
userdata-note-key	KW_ITX_NOTES	UC Connector includes any agent notes on the interaction in the Interaction window.
userdata-onringing	false	Controls whether the user data specified in `user-data-call_format<n>` is displayed in a ringing interaction on the Knowledge Worker.
userdata-note-onpreview	false	Controls whether user data specified in the `userdata-note-key` is displayed in a preview notification on the Knowledge Worker. **Limitation**: The agent notes for the Preview window cannot contain single quotation (') marks.
userdata-note-onringing	false	Controls whether user data specified in `userdata-note-key` is displayed in a ringing interaction on the Knowledge Worker.
userdata-contact-onpreview	false	Controls whether user data specified in `userdata-contact-format` is displayed in a preview notification on the Knowledge Worker.
userdata-contact-onringing	false	Controls whether the user data specified in `userdata-contact-format` is displayed in a ringing interaction on the Knowledge Worker.
userdata-title	Case data	Specify the heading name to be displayed in an interaction for all the user data specified by the `userdata-call-format<n>` options. **Note**: The text configured here appears in the GUI. If you are localizing the UC Connector language, make sure you modify this option to match the localized language.

Table 12: UC Connector—UC-Connector Section (Continued)

Option Name	Default Value	Description
userdata-note-title	Notes	Specify the heading name to be displayed in an interaction for all the user data specified by the `userdata-note-key` option. **Note**: The text configured here appears in the GUI. If you are localizing the UC Connector language, make sure you modify this option to match the localized language.
itx-window-close-timeout	9000	Enter the length of time you want the interaction to remain open after the Knowledge Worker interaction is released or abandoned.
user-unregister-timeout	300000 (5 minutes)	Enter the length of time, in milliseconds, that UC Connector will wait after Knowledge Worker has closed all browser sessions before it unregisters the Knowledge Worker DN with T-Server/SIP Server.

4. In the log section, configure the following UC Connector-specific log option:

- `internal`—Set this option to the level of message detail you want for the internal UC Connector components.

For other log-related options, see the *Framework 8.1 Deployment Guide*.

5. Depending on which UC platform you are integrating with, configure one of the following sections:
- `Microsoft-OCS`
- `IBM-Sametime`

Then configure the following options (for your UC platform):

Table 13: UC Connector—Third-party Platform Sections

Option Name	Value	Description
Microsoft-OCS Section		
agent-status-logout	1800, 0-2999, 1800[a]	Maps the `Offline` presence status in OCS to the Genesys `Logout` status.
agent-status-notready	6500,9500,12500,15500, 4500-18000[a]	Maps the `Away` presence status in OCS to the Genesys `NotReady` status.

Table 13: UC Connector—Third-party Platform Sections (Continued)

Option Name	Value	Description
agent-status-ready	3500, 3000-4499[a]	Maps the `Online` presence status in OCS to the Genesys `Ready` status.
contact	SIP URI	Enter the user name for the UC Connector as configured in Microsoft OCS (called the UC Connector "principal"). For example, `sip:ocs-ucc@your-domain.com`
registrar-uri	SIP URI	Enter the URI that UC Connector uses to connect with Microsoft OCS. For example, `sip:pool01.your-ocs-address.com`
IBM-Sametime Section		
agent-status-logout	-32768, 0[b]	Maps the presence statuses in IBM Sametime to the Genesys `Logout` status.
agent-status-notready	8,64,96,128,512,16384[b]	Maps the presence statuses in IBM Sametime to the Genesys `NotReady` status.
agent-status-ready	32,544[b]	Maps the presence statuses in IBM Sametime to the Genesys `Ready` status.
channel-type	62	Specifies that the voice channel on IBM Sametime is to be used for this connection.
server-port	1516	Specifies the port number for the IBM Sametime server. The default 1516 is typically used among Sametime components.
server-fqdn	FQDN	Specifies the fully-qualified domain name (FQDN) for the IBM-Sametime host computer.

a. See Table 14 for detail about these values.

b. See Table 15 for detail about these values.

Figure 27 shows a sample configuration for integration with Microsoft OCS.
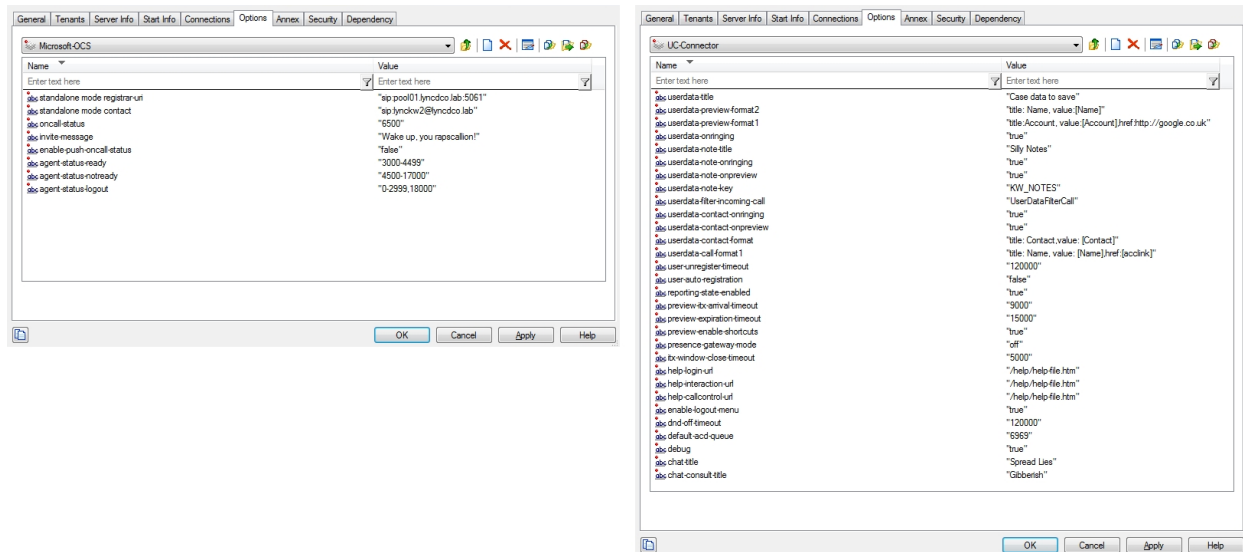
Figure 27: UC Connector Application—Options Tab

6. Click OK.

End of procedure

Next Steps

• This completes the basic deployment of the UC Connector Application object. Next, create the Genesys objects used to represent Knowledge Workers in the Genesys environment. See "Configuring the Knowledge Worker" on page 90.

Additional Reference Info

Table 14 shows the interoperability values for the various presence states used in Microsoft OCS 2007R2.

Note: These values correspond to those described in the Microsoft proprietary MS-PRES protocol. Genesys does not guarantee that these values will remain valid through future product updates. For the latest values, consult third-party Microsoft documentation.

Table 14:
Interoperability Values for OCS Presence States

Interoperability Value	OCS Presence State
0 – 4499	Available
4500 – 5999	Available - Idle
6000 – 7499	Busy
7500 – 8999	Busy - Idle
9000 – 11999	Do No Disturb
12000 – 14999	Be Right Back
15000 – 17999	Away
>= 18000	Offline

Table 15 shows the default status map for IBM Sametime presence states.

Table 15: Default Mapping for IBM Sametime Presence States

Genesys Agent Status	IBM Sametime Presence State
Loggedout	ST_USER_STATUS_UNKNOWN= -32768; ST_USER_STATUS_OFFLINE = 0;
Ready	ST_USER_STATUS_ACTIVE = 32; ST_USER_STATUS_ACTIVE_MOBILE = 544;
NotReady	ST_USER_STATUS_NOT_USING = 64; ST_USER_STATUS_AWAY = 96; ST_USER_STATUS_DND = 128; ST_USER_STATUS_MOBILE = 512; ST_USER_STATUS_DONTCARE = 16384; ST_USER_STATUS_IN_MEETING = 8;

Configuring the Knowledge Worker

The Knowledge Worker must be configured on both the Genesys side and on the third-party UC platform.

On the Genesys side, you configure a `Person` object for each Knowledge Worker, so that the Knowledge Worker is treated by the contact center as an agent, with access to all business rules, routing, and reporting available to a

regular agent. You must also create an `Extension` DN object for each Knowledge Worker who is to be integrated into the contact center.

About Emulated Agent Functionality

For integrations with T-Server, you must configure the Knowledge Workers as "emulated" agents. Consult the Deployment Guide for your respective T-Server for more information about configuring emulated agents.

Configuring Do Not Disturb Status at Login

UC Connector relies on T-Server/SIP Server to determine the Do Not Disturb status of the Knowledge Worker when they login. You can define this status by setting the relevant T-Server/SIP Server options that are used to control whether emulated agents are logged in to the `Ready` or `NotReady` state. Some T-Servers might not support defining this behavior using configuration options. In this case, the Do Not Disturb status of the Knowledge Worker at login cannot be defined. Consult the Deployment Guide for your T-Server for information about how to configure this feature.

After Call Work is Not Supported

UC Connector does not support After Call Work (ACW) time for Knowledge Worker agents. Ensure that AFC in the Agent Login for the Knowledge Worker is disabled (set `wrap-up-time` to `0`).

Task Summary: Configuring the Knowledge Worker

The following table describes the basic steps required to configure the Knowledge Worker on the Genesys side.

Task Summary: Configuring the Knowledge Worker

Step	Action
1. Create KW DNs.	Create an `Extension` DN for each Knowledge Worker you want to integrate into the contact center. For information about configuring Extensions, consult the respective Deployment Guide for your T-Server.
2. Create Agent Logins.	1. In the `Agent Login` folder under the switch, create an `Agent Login` object for each Knowledge Worker. 2. On the `General` tab, enter an agent login ID in the `Code` field. You will link to this agent login when you create the Person for the KW.

Task Summary: Configuring the Knowledge Worker (Continued)

Step	Action
3. Configure Places.	1. Create a `Place` object for each Knowledge Worker. 2. Add a shortcut to the DN you created in Step 1. For a detailed procedure, see Procedure: Creating the Knowledge Worker Place in Genesys Administrator or Procedure: Creating the Knowledge Worker Place in Configuration Manager.

4. Create KW Persons.	Each Knowledge Worker requires a Person to be created in the contact center.
	
	1. For each KW, create a `Person` object.
	
	2. On the `General` tab, define a user name and password (which you will use later to log into the UC Connector interface in the client).
	
	Note: Format depends on how `<userid>` is configured in the custom tab .xml file. See "<userid>" on page 110.
	
	3. In the `Annex` tab, `UC-Connector` section, configure the following option:
	   ◆ `enabled`
	
	4. In the section for third-party UC (`Microsoft-OCS` or `IBM-Sametime`), configure the following options:
	   ◆ `agent-status-ready` (optional)
	   ◆ `agent-status-notready` (optional)
	   ◆ `agent-status-logout` (optional)
	   ◆ `contact`
	
	5. For each Person, on the `Agent Info` tab:
	   ◆ Configure a default Place.
	   ◆ Assign the login ID that you created in Step 2.
	
	For details, see one of the following procedures:
	
	• Procedure: Creating the Knowledge Worker Person in Genesys Administrator
	
	• Procedure: Creating the Knowledge Worker Person in Configuration Manager.

Procedure:
Creating the Knowledge Worker Place in Configuration Manager

Prerequisites

- The Knowledge Worker DNs and Agent Logins are created. For details, see the following sections in the Task Summary: Configuring the Knowledge Worker table:
    - "Create KW DNs."
    - "Create Agent Logins."

Start of procedure

1. Under the tenant, right-click the Places folder, and select New > Place.
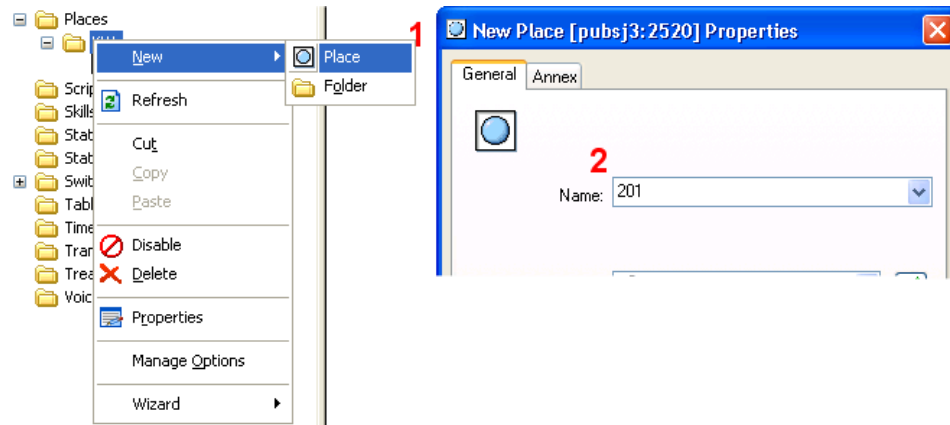
Figure 28: Creating a Genesys Place Object for a Knowledge Worker

2. After you have created the Place, right-click the new object and select New > Shortcut to DN.

3. Browse for the DN that you created in Procedure: Create KW DNs.
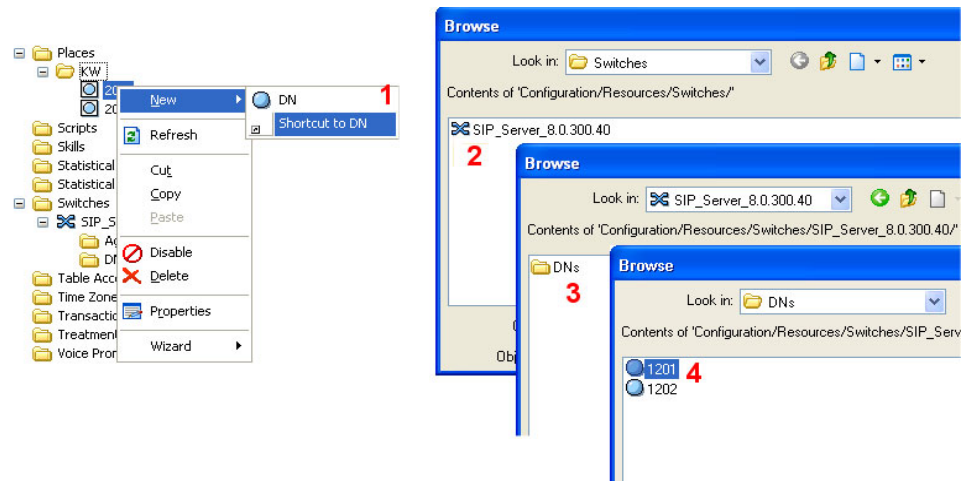
Figure 29: Adding a Shortcut to a Knowledge Worker DN

4. Click OK.

5. Repeat this procedure for every Knowledge Worker that you want to integrate into the contact center.

End of procedure

Next Steps

- Procedure: Creating the Knowledge Worker Person in Configuration Manager

Procedure:
Creating the Knowledge Worker Place in Genesys Administrator

Prerequisites

- The Knowledge Worker DNs and Agent Logins are created. For details, see the following sections in the Task Summary: Configuring the Knowledge Worker table:
  - "Create KW DNs."
  - "Create Agent Logins."

Start of procedure

1. Under Provisioning > Switching > Places, click the New ... button.

2. After you enter a name for the Place, click the Add ... button.

Figure 30: Creating a Genesys Place Object for a Knowledge Worker

3. Double-click your switch.

4. Double-click the DNs folder.

5. Select the DN that you created in Procedure: Create KW DNs. and click OK.
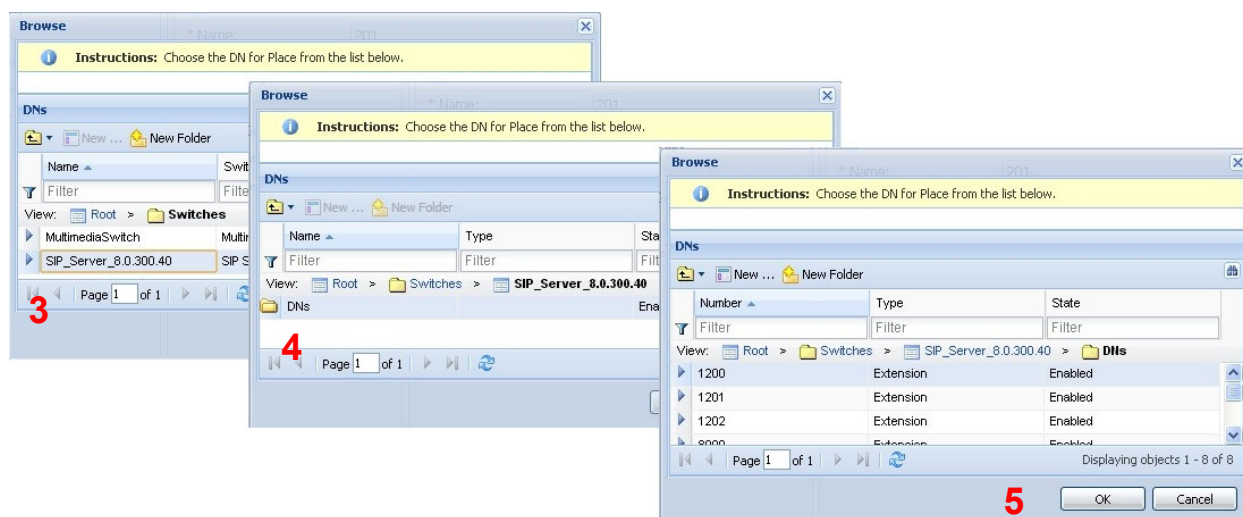
Figure 31: Adding a Knowledge Worker DN

6. Click Save.

7. Repeat this procedure for every Knowledge Worker that you want to integrate into the contact center.

End of procedure

Next Steps

• Procedure: Creating the Knowledge Worker Person in Genesys Administrator

Procedure:
Creating the Knowledge Worker Person in Configuration Manager

Prerequisites

•   An Agent Login is created for each Knowledge Worker. You will link to the ID for this Agent Login when creating the Person.

Start of procedure

1. Under the tenant, right-click the `Persons` folder, and select `New > Person`.

2. Configure the fields on the `General` tab. For the `user name` field, the format depends on how `<userid>` is configured in the custom tab .xml file. See "<userid>" on page 110. For more information about these fields, click Help (or consult the *Framework 8.1 Deployment Guide*).

3. On the `Agent Info` tab, browse and select the `Place` object that you created in Configure Places.

4. Click `Add ID` and browse to select the `Agent Login that` you created in Create Agent Logins.

5. On the `Annex` tab, create a `UC-Connector` section by using the following options.

Table 16: Knowledge Worker Person—UC-Connector Options

Option Name	Value	Description
enabled	true	Enables this Person object for use with UC Connector.

6. (`Annex` tab) Depending on which UC platform you are integrating with, configure one of the following sections:
    ·   `Microsoft-OCS`
    ·   `IBM-Sametime`

    Then configure the following options (for your UC platform):

Table 17: UC Connector—Third-party Platform Sections

Option Name	Value	Description
Microsoft-OCS Section		
agent-status-logout *(optional)*	>= 18000[a]	Maps the `Offline` presence status in OCS to the Genesys `Logout` status.
agent-status-notready *(optional)*	4500-17999[a]	Maps the `Away` presence status in OCS to the Genesys `NotReady` status.

Table 17: UC Connector—Third-party Platform Sections (Continued)

Option Name	Value	Description
agent-status-ready *(optional)*	0-4499[a]	Maps the `Online` presence status in OCS to the Genesys `Ready` status.
contact	SIP URI	Enter the Knowledge Worker sign in name for the UC Connector as configured in Microsoft OCS. For example, `sip:ocs-ucc@your-ocs-address.com`
IBM-Sametime Section		
agent-status-logout *(optional)*	-32768, 0[b]	Maps the presence statuses in IBM Sametime to the Genesys `Logout` status.
agent-status-notready *(optional)*	8,64,96,128,512,16384[b]	Maps the presence statuses in IBM Sametime to the Genesys `NotReady` status.
agent-status-ready *(optional)*	32,544[b]	Maps the presence statuses in IBM Sametime to the Genesys `Ready` status.
contact	LDAP entry	Enter the user name, in LDAP format, as configured in the IBM Sametime environment. For example, `CN=Jack smith/O=City Bank`

a. See Table 14 for details about these values. These take precedence over Application-level options.

b. See Table 15 for details about these values. These take precedence over Application-level options.

7. Click `OK`.

8. Repeat this procedure for each Knowledge Worker that you are integrating into the contact center.

End of procedure

Next Steps

• This completes the configuration of the Knowledge Workers as represented in the Genesys environment.

Next, you must integrate the UC Connector with the agent desktop used in your deployment. For example, you can integrate with Interaction Workspace 8.1. See "Integrating with Interaction Workspace".

Procedure:
Creating the Knowledge Worker Person in Genesys Administrator

Prerequisites

• An Agent Login is created for each Knowledge Worker. You will link to the ID for this Agent Login when creating the Person.

Start of procedure

1. Under `Provisioning > Accounts > Users`, click the `New ...` button.

2. Configure the fields on the `General` tab. For the `user name` field, the format depends on how `<userid>` is configured in the custom tab .xml file. See "<userid>" on page 110. For more information about these fields, click Help (or consult the *Framework 8.1 Deployment Guide*).

3. On the `Agent Info` tab, browse and select the `Place` object that you created in Configure Places.

4. Click `Add` and browse to select the `Agent Login that` you created in Create Agent Logins.

5. Click the `Options` tab and select `Advanced View (Annex)` from the `View` drop down list.

6. Create a `UC-Connector` section by using the following options.

Table 18: Knowledge Worker Person—UC-Connector Options

Option Name	Value	Description
enabled	true	Enables this Person object for use with UC Connector.

7. (`Annex` tab) Depending on which UC platform you are integrating with, configure one of the following sections:
   • `Microsoft-OCS`
   • `IBM-Sametime`

   Then configure the following options (for your UC platform):

Table 19: UC Connector—Third-party Platform Sections

Option Name	Value	Description
Microsoft-OCS Section		
agent-status-logout *(optional)*	>= 18000[a]	Maps the `Offline` presence status in OCS to the Genesys `Logout` status.

Table 19: UC Connector—Third-party Platform Sections (Continued)

Option Name	Value	Description
agent-status-notready *(optional)*	4500-17999[a]	Maps the `Away` presence status in OCS to the Genesys `NotReady` status.
agent-status-ready *(optional)*	0-4499[a]	Maps the `Online` presence status in OCS to the Genesys `Ready` status.
contact	SIP URI	Enter the Knowledge Worker sign in name for the UC Connector as configured in Microsoft OCS. For example, `sip:ocs-ucc@your-ocs-address.com`
IBM-Sametime Section		
agent-status-logout *(optional)*	-32768, 0[b]	Maps the presence statuses in IBM Sametime to the Genesys `Logout` status.
agent-status-notready *(optional)*	8,64,96,128,512,16384[b]	Maps the presence statuses in IBM Sametime to the Genesys `NotReady` status.
agent-status-ready *(optional)*	32,544[b]	Maps the presence statuses in IBM Sametime to the Genesys `Ready` status.
contact	LDAP entry	Enter the user name, in LDAP format, as configured in the IBM Sametime environment. For example, `CN=Jack smith/O=City Bank`

a. See Table 14 for details about these values. These take precedence over Application-level options.

b. See Table 15 for details about these values. These take precedence over Application-level options.

8. Click `Save`.

9. Repeat this procedure for each Knowledge Worker that you are integrating into the contact center.

End of procedure

Next Steps

- This completes the configuration of the Knowledge Workers as represented in the Genesys environment.

  Next, you must integrate the UC Connector with the agent desktop used in your deployment. For example, you can integrate with Interaction Workspace 8.1. See "Integrating with Interaction Workspace".

Integrating with Interaction Workspace

For the purposes of this guide, the agent side of the customer interaction is presented using Interaction Workspace 8.1. Interaction Workspace provides the functionality required to give agents access to experts outside the contact center. This functionality includes:

- Showing a group of Knowledge Workers in the Interaction Workspace Buddy List (the group will appear as a regular agent group).
- Showing an individual Knowledge Worker in the Interaction Workspace Buddy List (the KW will appear as a regular agent).

For more information about using the Buddy List, consult the *Interaction Workspace 8.1 Deployment Guide* and the *Interaction Workspace 8.1 User's Guide*.

If you are using previous versions of Agent Desktop, or a customized agent client using the Genesys SDK, some modifications may be required to achieve the above functionality. Consult the Agent Desktop Deployment Guide for your version of the product, or the Genesys SDK used to create your custom agent client.

Next Steps

- Next, integrate UC Connector with your deployment's UC platform. See "Integrating with the Third-Party UC Platform".

Integrating with the Third-Party UC Platform

Depending on your UC platform, see either of the following:

- "Integrating with Microsoft Lync Server"
- "Integrating with Microsoft OCS 2007 R2"
- "Integrating with IBM Sametime 8.5"

Integrating with Microsoft Lync Server

The following table lists the basic steps required to integrate UC Connector with Microsoft Lync Server, as well as with the Lync client.

Task Summary: Configuring Microsoft Lync Server

Objective	Actions
1. Modify the Windows Registry.	Import the registry file included in the `microsoft-oc-client` folder of the UC Connector installation. • Procedure: Modifying the Registry for Microsoft Lync, on page 102
2. Enable Automated Login.	In the UC Connector `Application` object, go to the `uc-connector` section and configure the following option: • `user-auto-registration` — Set this to true. UC Connector will automatically login all Knowledge Worker DNs on application restart. **Key Notes** • Lync integration does not use "custom tabs" as in previous OCS versions, so individual Knowledge Worker login is not possible. For Lync integrations, this configuration is mandatory.
3. Enable secured mode.	Secured mode—using TLS or MTLS— is mandatory with Microsoft Lync. MTLS is required for push presence status functionality. Complete one of the following tasks: • Task Summary: Enabling MTLS Communication, on page 131 • Task Summary: Enabling TLS/Kerberos Secure Communication, on page 140

Task Summary: Configuring Microsoft Lync Server (Continued)

Objective	Actions
Optional Configuration	
• *(optional)* Enable Push On Call Status.	To enable push-status from the contact center to the Lync client, you can configure the `Microsoft-OCS` section as follows: • Globally in the UC Connector `Application` object. • Individually in the particular Knowledge Worker's corresponding `Person` object. Configuration options for this feature are: · `enable-push-oncall-status` · `oncall-status` **Key Notes** • The Person-level settings take precedence over the Application-level settings. • With Lync, MTLS is required. See Task Summary: Enabling MTLS Communication, on page 131.

Procedure: Modifying the Registry for Microsoft Lync

Purpose: UC Connector integrates with Microsoft Lync Server and client through an extensibility window that opens when an interaction arrives at the Knowledge Worker's desktop. This integration requires that you import to the Windows Registry a registry file included in the UC Connection installation.

Start of procedure

1. On the UC Connector host machine, locate the registry file required for your operating system. Go to the `microsoft-oc-client` folder in the UC Connector installation, and select one of the following registry files:
   - `mocApp` — For OCS 2007 R2 installations.
   - `mocAppLync` —For Lync installations on 32-bit systems.
   - `mocAppLync64` — For Lync installation on 64-bit system.
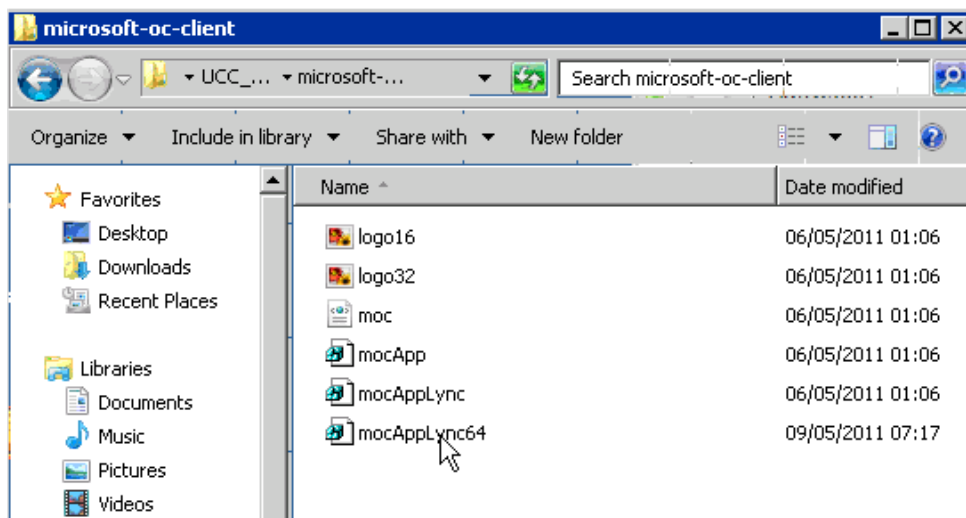
Figure 32: Location of Registry File

2. Open this file in a text editor, to determine the registry location where you will later import this file. For an example, see the following figure. The path you need is highlighted in red.
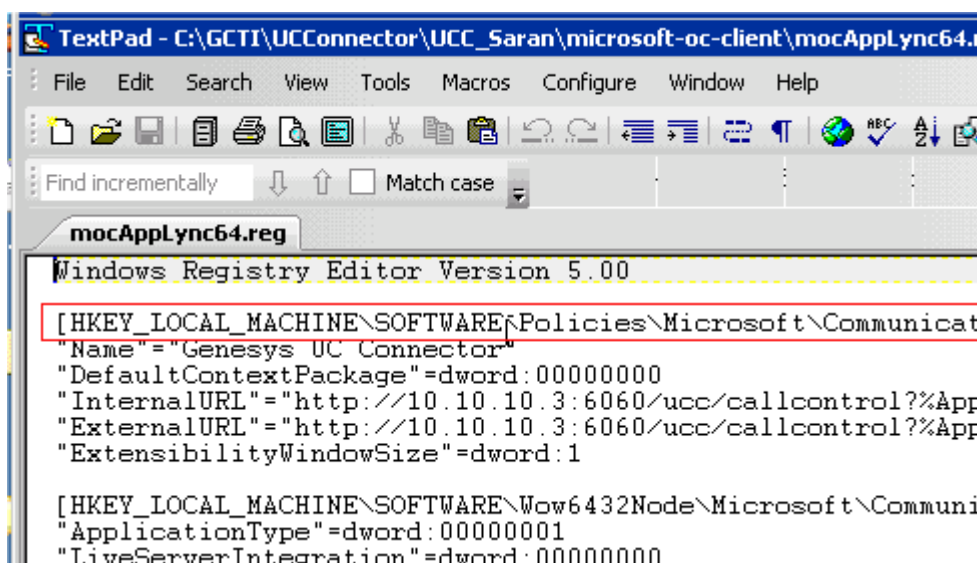
Figure 33: Path to Registry Location

3. On the Knowledge Worker machine (where the Lync client is installed), open the Windows Registry Editor. Click `Start > Run > Regedit`.

4. Navigate to the path that you found in Step 2.

Note: Changes made to one registry location will automatically propagate to the other locations as well; you only need to import the file once.

---

5. Import the registry file. Click `File > Import`.

The `Import Registry File` window opens.

6. Browse to the `microsoft-oc-client` folder and select the registry file for your deployment.
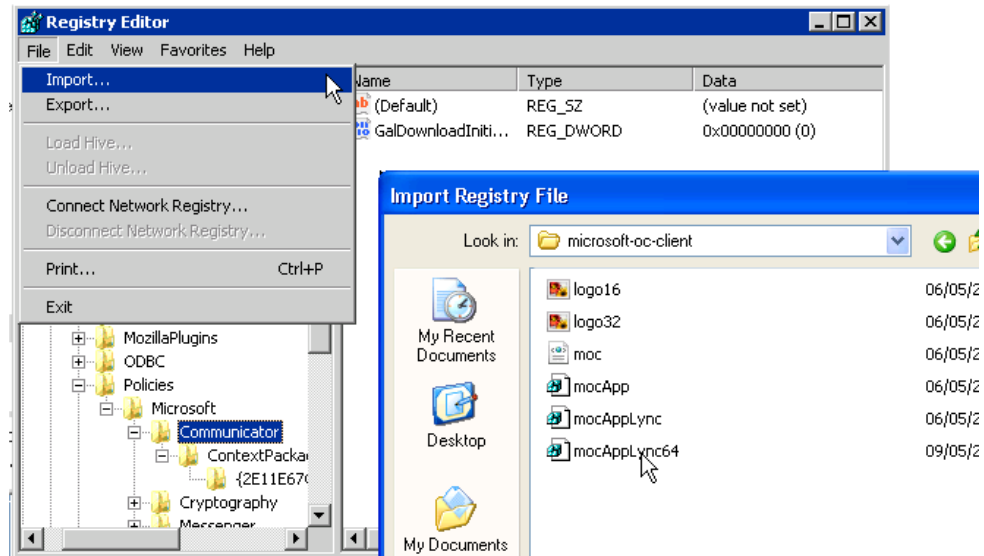
Figure 34: Importing the Registry File

7. Click `Open` to finish the import.

8. Repeat this process for every Knowledge Worker/Lync client machine in your deployment.

End of procedure

Next Steps

1. Set `user-auto-registration` (page 196) to `true`. This enables automated login for all Knowledge Workers. For details, see the section "Enable Automated Login." on page 101 in the Task Summary table.

2. Secured mode is mandatory for integrations with Microsoft Lync.

- For TLS, see Task Summary: Enabling TLS/Kerberos Secure Communication, on page 140.
- For push presence status functionality with Lync, Mutual Transport Layer Security (MTLS) is required. Continue at Task Summary: Enabling MTLS Communication, on page 131.

3. Once secured mode is configured, you must integrate the UC Connector deployment with Genesys Routing. Continue at Task Summary: Integrating with Genesys Routing, on page 119.

Integrating with Microsoft OCS 2007 R2

The following table lists the basic steps required to integrate UC Connector with the third-party UC server (Microsoft OCS 2007 R2), as well as with the UC client (Microsoft Office Communicator).

Task Summary: Configuring Microsoft OCS 2007

Step	Actions
1. Add Authorized Hosts to OCS.	If integrating with Microsoft OCS 2007 R2, you must add the UC Connector host IP address to the list of Trusted Hosts/Servers on OCS. For details, see Procedure: Adding Authorized Hosts to Microsoft OCS 2007 R2, on page 106. **Note**: If the UCC host cannot be considered as trusted by OCS, you may want to enable Kerberos security instead. See "Enabling Secure Communication" on page 129
2. Modify the Custom Tab Definition File.	Modify the Custom Tab Definition File so that it points to the web page for the custom tab. See Procedure: Modifying the Custom Tab File, on page 108.
3. Modify the Windows registry.	**On each Knowledge Worker computer:** In the Registry, go to `HKEY_LOCAL_MACHINE` > `SOFTWARE` > `Policies` > `Microsoft` and add a Communicator key. Within this key specify a new string value called `TabURL`, with the value pointing to the path of the `moc.xml` file. For example, `http://web_host_1/ucc/moc.xml` For details, see Procedure: Modifying the Windows Registry, on page 111.
4. Add Trusted Hosts to Internet Explorer.	**On each Knowledge Worker computer:** In the Trusted Hosts for Internet Explorer: • In non-HA deployments: add the UC Connector host. • In HA deployments, add the Windows NLB virtual IP. • If the Custom Tab Definition File is provided on a web server, add this web server address.

Task Summary: Configuring Microsoft OCS 2007 (Continued)

Step	Actions
Optional Configuration	
• Integrate without using the Custom Tab.	If you do not want to use the custom tab integration, you can skip those steps.
	In this case, UC Connector will not maintain an ongoing web connection to OCS. You need to configure the solution to use the native OCS preview mechanism for incoming interactions:
	1. Import the `mocApp` registry file to the Windows Registry on each Knowledge Worker machine where the MOC installed.
	For more information on how to import the registry file, see the procedure for Microsoft Lync:
	◆ Procedure: Modifying the Registry for Microsoft Lync, on page 102
	2. Set `user-auto-registration` to `true`.
• *(optional)* Enable Push On Call Status.	To enable push-status from the contact center to the MOC client, you can configure the `Microsoft-OCS` section as follows:
	• Globally in the UC Connector `Application` object.
	• Individually in the particular Knowledge Worker's corresponding `Person` object.
	Configuration options for this feature are:
	・ `enable-push-oncall-status`
	・ `oncall-status`
	Note: The Person-level settings take precedence over the Application-level settings.

Procedure:
Adding Authorized Hosts to Microsoft OCS 2007 R2

Note: This procedure involves adding UCC hosts as an Authorized Host in the OCS installation. If the UCC host should not be authorized, you may want to enable Kerberos security instead. For details, see "Enabling Secure Communication" on page 129.

Start of procedure

1. On the host computer where Microsoft OCS 2007 R2 is installed, go to
    `Start > Administrative Tools > Office Communications Server 2007 R2`.

2. In the folder structure for OCS, click through `Enterprise pools` > *your
    configured pool* > `Properties` > `Front End Properties`.
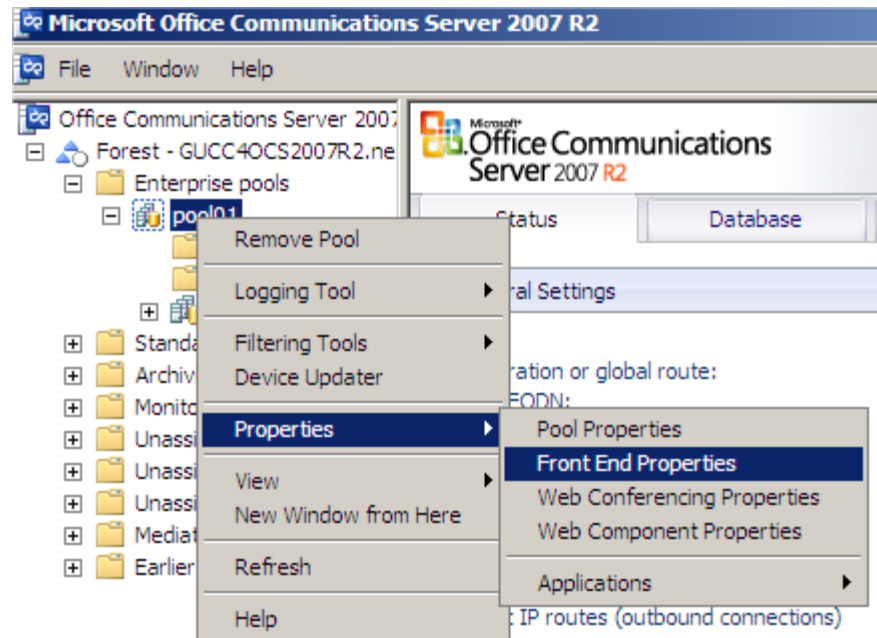
Figure 35: Selecting Front End Properties

3. On the `Host Authorization` tab, add the IP address for the UC Connector
    host, and specify the following:

    *   `Outbound Only`—Select `No` to disable.
    *   `Throttle As Server`—Select `Yes` to enable.
    *   `Treat As Authenticated`—Select `Yes` to enable.

    Note: In HA deployments, add the host where the backup UC Connector
    instance is also running.
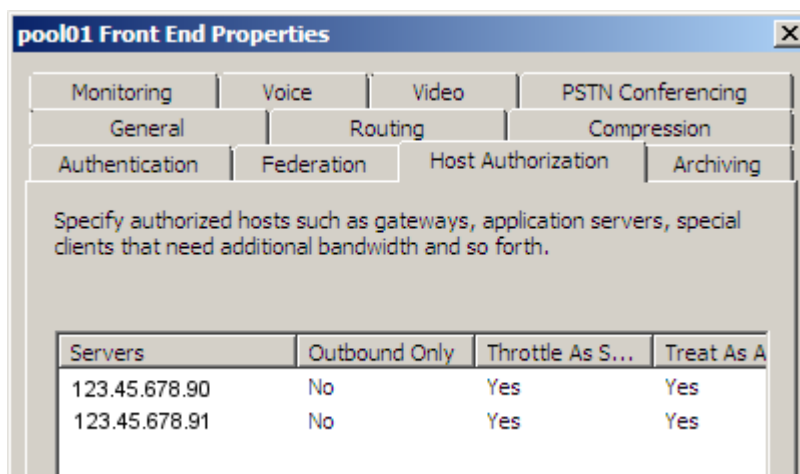
Figure 36: Front End Properties Window

End of procedure

Next Steps

- Procedure: Modifying the Custom Tab File

Procedure:
Modifying the Custom Tab File

The Custom Tab File is used to point Microsoft Office Communicator (MOC) to the main UC Connector Web interface and the graphics files it needs to create the custom UC Connector tab in the Knowledge Worker MOC client.

A sample `moc.xml` file is included in the UC Connector installation package, installed to the following location:
`<UCC_Install_Path>\microsoft-oc-client\moc.xml`

This file includes includes the basic xml parameters required to create a single custom tab. If you are already using a custom tab for your MOC clients, then you will need to modify the existing .xml file, creating a new section for the UC Connector tab. MOC can handle up to 32 custom tabs.

For more information about custom tabs, consult the Microsoft documentation here:
`http://technet.microsoft.com/en-us/library/dd425110(office.13).aspx`

Note: The references to third-party documentation in this section, including any URL or other web references, are subject to change without notice. They are included for your convenience.

Prerequisites

* UC Connector has been installed on the host computer. The installation includes the `moc.xml` file used for the custom UC Connector tab.

* A `.png` graphics file has been placed on a network-accessible location. This graphic file must be either 16x16 or 32x32 pixels in dimension. For your convenience, a sample Genesys logo is included in the UC Connector installation: `<UCC_Install_Path>\microsoft-oc-client\logo.png`.

Start of procedure

1. In the directory where you installed UC Connector, locate the .xml tab definition file. Or create/modify your own. xml file. The following is a sample:

```
<?xml version="1.0" ?>
 <tabdata>
   <tab>
     <image>http://your_webserver_IP/ucc/genesys_logo.png</image>
     <userid>true</userid>
     <name>Contact Center</name>
     <tooltip>Contact Center</tooltip>
<contenturl>http://<UCC_host_FQDN>:<UCC_HTTP_port>/ucc/app/welcome<
/contenturl>
   </tab>
  </tabdata>/
```

The following table describes the key parameters in more detail.

Table 20: Custom Tab XML Definition File

XML tag	Description
<image>	Value: URI
	Specifies the URI to the prerequisite `.png` file used to fill the custom tab
	For example:
	`http://<your-web-server>/mds_icon.png`
	The URI can be in the following format:
	• https: (recommended)
	• http://
	• file:///
	Note: This icon only appears if two or more tabs are configured in the .xml file. However, the image file MUST be included for the tab to appear.

Table 20: Custom Tab XML Definition File (Continued)

XML tag	Description
\<userid\>	Value: `true` • `true`—Microsoft OCS passes the userid from the OCS login to the UC Connector custom tab. It cannot be edited. **Note**: The username configured in the corresponding `Person` object in Configuration Layer must be in the format `sip:<MOC_User_Name>`. For example, if user logs in to OCS with `Joe.Smith@company.com`, then the corresponding Person must be configured with `User Name` set to `sip:Joe.Smith@company.com`. • `false`—The Knowledge Worker will specify the user name when logging into the UC Connector interface on the custom tab. In this case, there are no restrictions on the format for the user name configured in the `Person` object.
\<name\>	Friendly name for the tab. Does not display in the interface.
\<tooltip\>	Displays the text that will show when the mouse pauses over the tab.

\<contenturl\>	Value: URL to the UC Connector web page
	
	 Specify the URL to the web page built into the UC Connector application. Use the following format:
	 `http://<UCC_host_FQDN>:<HTTP_port>/ucc/app/welcome`
	 For example:
	 `http://your-web-server.com:12345/ucc/app/welcome`
	
	 Note: For the HTTP port in this URL, use the port number you assigned to the UC Connector in the installation wizard. See Step 5 on page 77.

2. Place the .xml file in a network-accessible location. This can be a web server, a network folder, or local folder on each Knowledge Worker computer. It is important that this file is accessible from every Knowledge Worker host computer.

End of procedure

Next Steps

• Procedure: Modifying the Windows Registry, on page 111

Procedure:
Modifying the Windows Registry

Start of procedure

1. Using a registry editor (for example, `Run > regedit`), go to either of the following registry keys:

    `HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Communicator`

    `HKEY_CURRENT_USER\Software\Policies\Microsoft\Communicator`

    Note: The `HKEY_LOCAL_MACHINE\SOFTWARE` section takes precedence.

    ---

2. Create a new key called `TabURL`, with the value pointing to the path of the `moc.xml` file location.

    You can use any of the following kinds of locations:
    *   `file:///\\server\share\ucc\tabs.xml`
    *   `file:///c:/ucc/tabs.xml`
    *   `http://web_server/ucc/tabs.xml`

    For example,

    `http://web_host_1/ucc/moc.xml`

    > **Tip:** The web_host must be configured as a Trusted Host in the web browser on the Knowledge Worker computers. See Step 4 on page 105.

3. Do this on the host computer for each Knowledge Worker to be integrated with the contact center.

End of procedure

Next Steps

1. On each Knowledge Worker computer, you must add the UC Connector host to the list of Trusted Hosts in Internet Explorer. For details, see the section "Add Trusted Hosts to Internet Explorer." on page 105 in the Task Summary table.

2. This completes the integration with Microsoft OCS 2007 R2. Next, you must integrate the UC Connector deployment with Genesys Routing. Continue at Task Summary: Integrating with Genesys Routing, on page 119.

Integrating with IBM Sametime 8.5

The following table lists the basic steps required to integrate UC Connector with the third-party UC server (IBM Sametime 8.5), as well as with the UC client (Sametime Connect).

Task Summary: Integrating with IBM Sametime 8.5

Step	Actions
1. Add Authorized Hosts to IBM Sametime.	If integrating with IBM Sametime 8.5, you must add the UC Connector host IP address to the list of trusted IPs in the Sametime configuration. For details, see Procedure: Adding UC Connector host to list of trusted IPs in Sametime, on page 113.
2. Install the Sametime Connect UCC plug-in.	To install the UC Connector plug-in, perform the following steps: 1. Create a website from the following folder, found in the UCC installation directory: `ucc-sametime-client-site` 2. In Sametime Connect, create a new plug-in that points to this website. 3. Configure the `Server URL` for this plug-in to point to the UC Connector host and HTTP port, as specified in the Install Wizard. For details, see Procedure: Installing the UC Connector plug-in for Sametime Connect, on page 114.

Task Summary: Integrating with IBM Sametime 8.5 (Continued)

Step	Actions
(optional) Enable Push On Call Status.	To enable push-status from the contact center to the IBM Sametime client, you can configure the `IBM-Sametime` section as follows: • Globally in the UC Connector `Application` object. • Individually in the particular Knowledge Worker's corresponding `Person` object. Configuration options for this feature are: · `enable-push-oncall-status` · `oncall-status` · `oncall-status-message` · `call-released-status` **Note**: The Person-level settings take precedence over the Application-level settings. For details, see Procedure: Enabling push status from Genesys to Sametime, on page 118.

Procedure:
Adding UC Connector host to list of trusted IPs in Sametime

Start of procedure

1. On the host computer where IBM Sametime is installed, navigate the installation directory to find the `sametime.ini` file. For example, the default installation path would be:

   `C:\Program Files\IBM\Lotus\Domino\sametime.ini`

2. Open the `sametime.ini` file and locate the parameter `VPS_Trusted_IPS`.

3. Add the IP address for the host computer where you installed the UC Connector instance. Use a comma-separated list for multiple IP addresses.

Note: For HA deployments, add both the primary and backup UC Connector hosts.
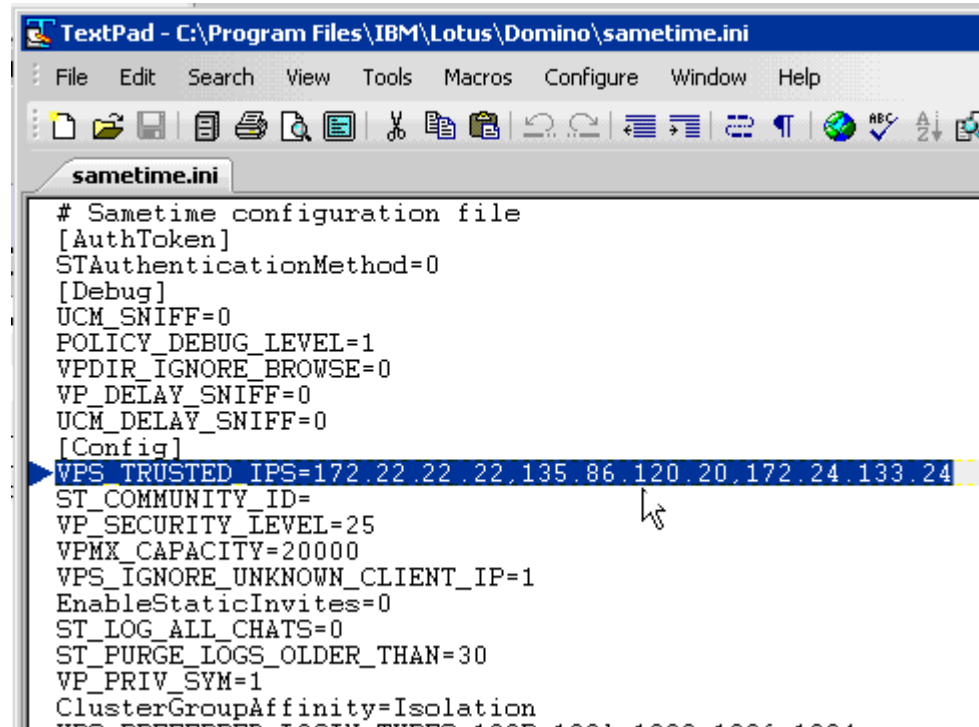
Figure 37: Adding UCC host IP to the sametime.ini File

End of procedure

Next Steps

Procedure:
Installing the UC Connector plug-in for Sametime Connect

Purpose: To enable the custom UC Connector tab in the Sametime Connect client.

UC Connector integrates with Sametime Connect as a plug-in, where the plug-in points to a website created from special ucc-sametime integration files included in the UC Connector installation. This plug-in creates a custom "Contact Center" panel in the Sametime Connect client—the Knowledge Worker uses this panel to access contact center resources, as well as view current contact center interactions.

Prerequisites

• UC Connector is installed on the host computer and you have access to the folder directory where the application is installed.

- You have access to a web server on this host computer. For example, in Windows Server 2008 you can use Microsoft Internet Information Services (IIS) to host the plug-in.

- Obtain the host IP address and HTTP port for UC Connector. This is the information that you entered on the `User Parameters` page of the UCC installation wizard. For more information, see Procedure: Installing the UC Connector server on the host, on page 81 .

Start of procedure

1. Create a website using the content of the `ucc-sametime-client` folder, located in the root directory of the UC Connector installation.

   `C:\<ucc_root>\ucc-sametime-client-site`

   For example, on a Windows 2008 Server host computer, create a Virtual Directory in IIS that points to this `ucc-sametime-client-site` folder.
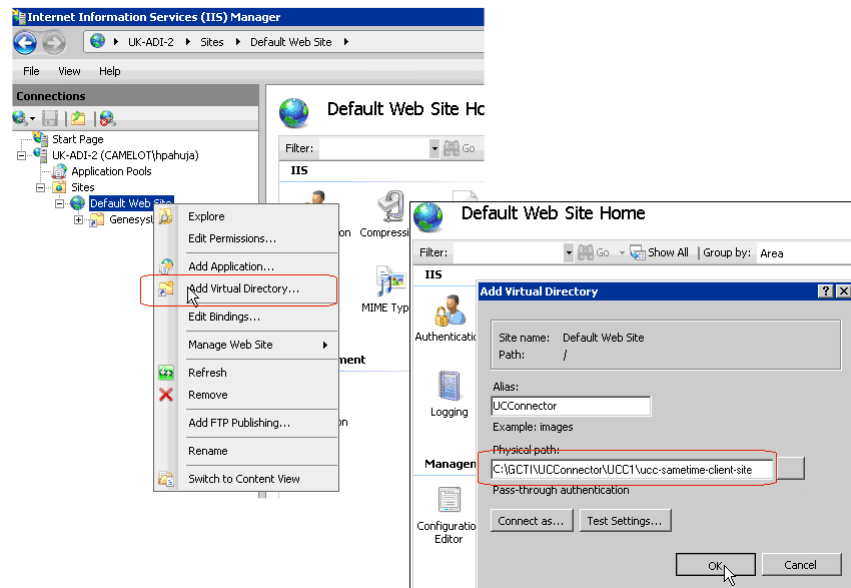
Figure 38: Virtual Directory in ISS for UCC plug-in

To test the website, enter the host computer and the name for this website in your web browser. You should see something like the following:
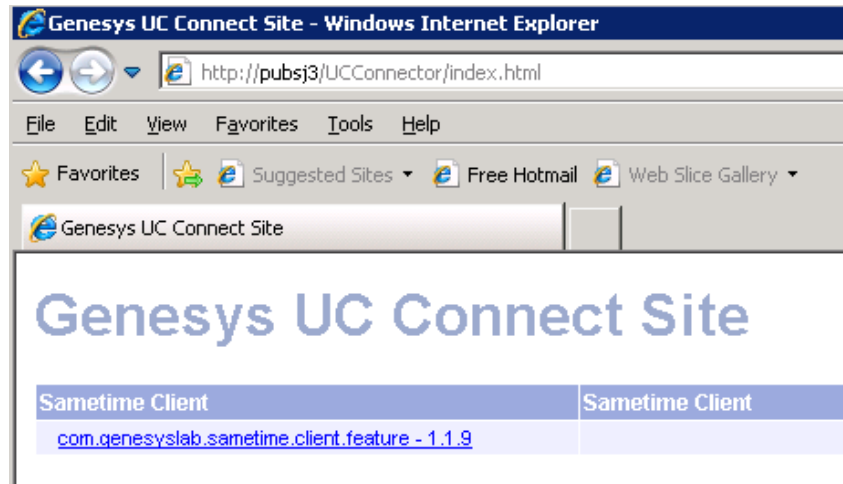
Figure 39: UC Connector Web Page

2. On the Knowledge Worker host computer, open Sametime Connect and
   select `Tools > Plug-ins > Install Plug-ins...`
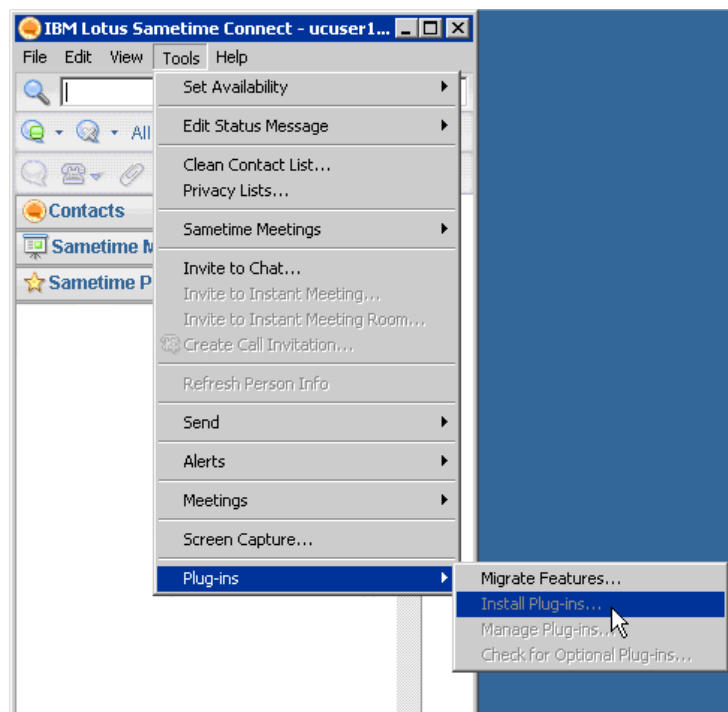
Figure 40: Installing the Plug-In on the KW Host Sametime Connect client

3. In the installation dialog box, select `Search for new features to install`
   and click `Next`.

4. Click `Add Remote Location....`

5. Enter the following information:

- • `Name`—Give the custom contact center panel the name as you want it to appear in the Sametime Connect client. For example, `Genesys UC Connector`.
- • `URL`—Enter the URL to the website that you created in Step 1.

6. Click `OK`, `Finish`, then follow the installation process.

7. Restart Sametime Connect. You should now see the contact center panel at the bottom of the client.
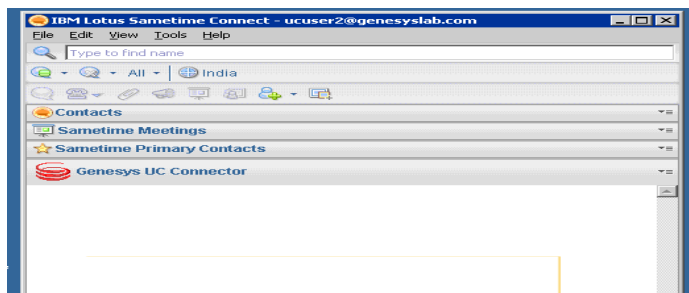
Figure 41: Empty Contact Center Panel

Currently, the custom panel should be empty, because the Sametime Connect client is integrated with the UC Connector website, but not with the UC Connector application itself.

8. Select `File > Preferences > Genesys UC Connector` (or whatever you named your plug-in in Step 5).

9. In the `Server URL` field, enter the host address (FQDN or IP) and HTTP host for the UC Connector application. This information should match the information that you provided on `User Parameters` in the Install Wizard (for details, see Procedure: Installing the UC Connector server on the host, on page 81).

10. Restart Sametime Connect. You should now see the custom panel with login.

11. You must add this plug-in (Steps 2 to Step 5) to the Sametime Connect client for every Knowledge Worker you want to integrate into the contact center.

End of procedure

Next Steps

- • Procedure: Enabling push status from Genesys to Sametime

Procedure:
Enabling push status from Genesys to Sametime

Prerequisites

- The UC Connector has been installed and configured to the default settings as described in "Deploying the UC Connector" on .

Start of procedure

1. You can configure this feature globally for all Knowledge Workers, as well as individually per Knowledge Worker. The individual setting takes precedence:
   - Globally—Open the UC Connector `Application` object, `Options` tab.
   - Individually—Open the `Person` object for the particular Knowledge Worker, `Annex` tab.

2. In the `IBM-Sametime` section, configure the following options:

Table 21: Push Status Option Settings

Option	Setting	Description
enable-push-oncall-status	true	Enables push-status functionality.
oncall-status	128	Pushes the `Do-Not-Disturb` status in Sametime Connect when the Knowledge Worker is on a Genesys call.
oncall-status-message	\<text\>	Enter the message to be shown in Sametime Connect for when the Knowledge Worker is on a Genesys call.
call-released-status	"reset"	Enter the status to be pushed when the Knowledge Worker releases the Genesys call. The default "reset" means the Knowledge Worker will revert to the last status in Sametime Connect before the Genesys call.

3. Click `OK` to save your changes.

End of procedure

Next Steps

- This completes the integration with IBM Sametime. Next, you must integrate the UC Connector deployment with Genesys Routing. Continue at

Integrating with Genesys Routing

Genesys routing is used to handle transfers and conferences both to and from the Knowledge Worker. For agent transfers to the Knowledge Worker, the routing strategy must be designed to include the Preview Interaction, and the `Routing Point` DN—or DNs—should be accessible to the agent desktop. For Knowledge Worker transfers to the contact center, special "contact points" must be created in Genesys. These contact points (`Routing Point` DNs) are then exposed in the UC client Interaction window, so that Knowledge Workers can send calls back to the contact center for further processing. In these strategies, no preview interaction is required.

The following table describes the main steps required to enable Genesys routing to and from Knowledge Workers.

Task Summary: Integrating with Genesys Routing

Objective	Actions
1. Create a "dummy" Custom Server Application object.	1. Import the `Custom_Server_800.apd` application template. 2. Create the Custom Server `Application` object with the same host as UC Connector. **Key Rules** • Do not install the Custom Server `.exe` file. Only the dummy application is required in the Configuration Layer. This is because an instance of the Custom Server is included with the UC Connector executable. • Create one dummy Custom Server for each instance of UC Connector. • For HA deployments, deploy one dummy Custom Server for each HA UC Connector pair. For more information, see "Configuring HA for Custom Server" on page 179. For details, see Procedure: Creating the Custom Server Application object.

Task Summary: Integrating with Genesys Routing (Continued)

Objective	Actions
2. Connect UC Connector to Custom Server	Add connections to Custom Server in both UC Connector and URS `Application` objects: • On the `Connections` tab of the `Application` object, `Add` and browse for the Custom Server object you created in Step 1. • Do this for both UC Connector and URS.

3. Configure a contact point for the KW.	1. Configure a `Routing Point` DN as the contact point that Knowledge Workers can use to transfer/conference interactions back to the contact center. Add the following options to the `Annex` tab:
	
	  ♦ `enabled`—Set this option to `true`.
	
	  ♦ `display-name`—Set this option to the name that will be displayed in the Interaction window. For example, `Contact Center`.
	
	  ♦ `attribute<n>`—Set this option to the statistics you want to make available for this contact point.
	
	For details, see Procedure: Configuring contact points.
	
	2. Create a routing strategy that handles agent selection in the contact center, and load it on this DN. Preview Interactions for transfers to agents are not required.
	
	Key Rules
	
	• Create a separate `Routing Point` DN for each contact point that you want to appear in the Interaction window.

4. Configure routing from contact center to KW.	• Configure Routing Points for contact center agent transfers to the Knowledge Worker. • Create and load the routing strategies that will direct transferred calls to the Knowledge Worker. **Sample Strategies** For sample strategies that you can import or use as a model for your own strategies, see the following: • Routing to a Particular Knowledge Worker • Routing with Round-Robin Selection • Routing with Broadcast Preview

Procedure:
Creating the Custom Server Application object

Prerequisites

- You have the Universal Routing product CD or downloaded IP ready.
- URS is installed and configured according to the procedures described in the *Universal Routing 8.0 Deployment Guide*.

Note: This procedure provides an overview of the main steps required to install and configure Custom Server. For more detailed procedures, see the *Universal Routing 8.0 Deployment Guide*.

Start of procedure

1. Import the Custom Server `Application` template.

   Go to the `Applications Template` folder. Import the `Custom_Server_800.apd` Application Template from the Universal Routing CD.

2. Create the Custom Server `Application` object.

   Go to the `Applications` folder and create a new Custom Server Application object based on the template you imported in Step 1.

3. On the `General` tab, specify the Application name.

4. On the `Server info` tab, specify the following:
   - For non-HA Deployments—Add the same host that is as used by UC Connector.
   - For HA deployments—Add the host and port for the Windows NLB virtual IP cluster.

5. Click `OK` to save.

6. Verify that both UC Connector and URS are both connected to the Custom Server application.

End of procedure

Next Steps

1. Connect custom server to UC Connector. For details, see the "Connect UC Connector to Custom Server" section in the task summary table.

2. Next, continue at Procedure: Configuring contact points.

Procedure:
Configuring contact points

Purpose: To create the `Routing Point` DNs exposed in the Knowledge Worker Interaction window for transfer or conferences to the contact center.

Start of procedure

1. Under a configured `Switch` object, select the `DNs` folder. In the `File` menu, select `New > DN` to create a new DN object of the type `Routing Point`.
2. On the `Annex` tab, create a `UC-Connector` section with the following options.

Table 22: Contact Point DN—UC-Connector Section

Option	Description

`attribute<n>`	**Value**: `format: <display text>: %s, statistic:<object>, <ObjectType>, <TenantName>, <StatType>, <TimeProfile>, <StatServerName>`
	
	Specifies the statistic to be displayed when the user places their cursor over the contact point in their Interaction window.
	
	For example, to display the number of calls waiting on a contact point (Routing Point 1111) as follows,
	
	`Calls Waiting: <#of calls>`
	
	configure the value for this option as follows,
	
	`format: Calls waiting: %s, statistic:1111@YourSwitch, RoutePoint, YourTenantName, CurrNumberWaitingCalls, Default`
	
	Key Actions:
	• All fields are required for UC Connector to properly subscribe to get the `CurrNumberWaiting` statistic from Stat Server.
	• You can enter the \<n\> variable in the option name as 1, 2, 3, and so on, or leave it empty.
	• The parameter `<StatServerName>` is only required if UC Connector connects to more than one Stat Server.
	
	For details, see the `attribute<n>` option description on page 205.

Table 22: Contact Point DN—UC-Connector Section (Continued)

Option	Description
`display-name`	**Value**: `<required>` Enter a unique descriptive name for this contact point, as you want it to appear in the Interaction window. For details, see the option description `display-name` on page 206.
`enabled`	**Value**: `true` Enables the contact to be used by the UC Connector. For details, see the option description `enabled` on page 206.

Figure 42 shows a sample configured `Routing Point` DN.
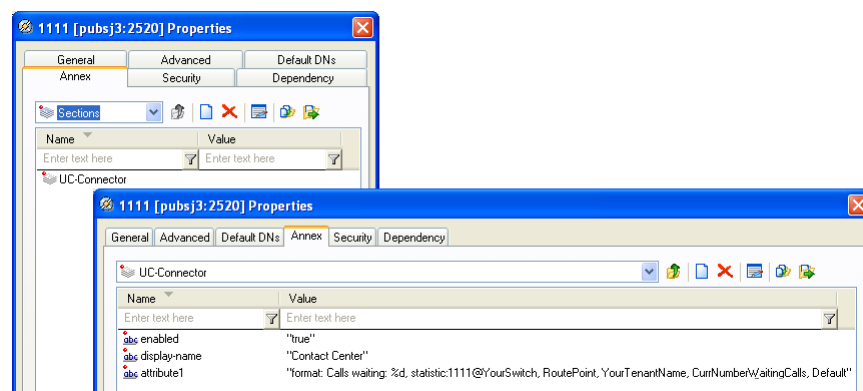
Figure 42: Contact Point Configuration—Routing Point DN

3. Click `OK`.

4. Repeat this procedure (creating a separate `Routing Point` DN) for each contact point that you want to appear in the Interaction window.

End of procedure

Next Steps

1. Create a routing strategy that handles agent selection in the contact center, and load it on this DN. Preview Interactions for transfers to agents are not required. For general information about designing routing strategies, see the following Universal Routing documents:

  ◆ *Universal Routing 8.0 Strategy Samples*
  ◆ *Universal Routing 8.0 Reference Manual*

2. If you are deploying a voice-only UC Connector solution, continue at Chapter 6, "Configuring the Routing Strategies," on page 161.

Optional • If you want to include Instant Messaging interactions in the deployment, continue at one of the following (depending on how IM is going to be provided):

-
-

Optional • If you want to enable secure communication in the deployment (and have not already done so), continue at

Mandatory • Secured mode is mandatory for integrations with Microsoft Lync Server.

- For TLS, see
- For push presence status functionality with Lync, Mutual Transport Layer Security (MTLS) is required. Continue at

Enabling Instant Messaging

Instant Messaging (IM) functionality is available through either of the following configurations:

- Integration with Genesys Instant Messaging (IM).
- SIP Server integration with Microsoft Lync and OCS 2007 R2

Pros and Cons

The following table describes the advantages and disadvantages of the supported IM configurations.

Table 23: IM Configurations—Pros and Cons

Genesys IM Integration	SIP Server-OCS Integration
• Platform-independent. Can be used regardless of UC platform.	• Limited to OCS deployments.
• Inbound IM only. The Knowledge Worker can only accept incoming IM sessions. They cannot initiate new IM sessions with the contact center.	• Inbound and outbound IM. The Knowledge Worker can both accept incoming IM sessions, as well as initiate new IM sessions with the contact center.

Chapter 5: Deploying UC Connector

Enabling Instant Messaging

Table 23: IM Configurations—Pros and Cons (Continued)

Genesys IM Integration	SIP Server-OCS Integration
• Simplified configuration. You can configure a single DN to support both voice and IM interactions. **Note:** You can still choose to keep the interactions on separate DNs— for example, to separate voice and chat traffic across two separate switches.	• Multiple DNs required. OCS-SIP Server integration requires a separate Knowledge Worker DN for voice and chat. It also requires a special DN to handle the direct connection between SIP Server and the UC platform.
• UCC-controlled IM window.	• OCS-controlled IM window
• T-Library With UC Connector, chat through Genesys IM is controlled by the T-Library interface. This allows UC Connector to manage the IM session.	• SIP Chat is provided through the SIP interface.

Deployment Guide

125

Enabling Genesys IM

The following table shows the main steps required to integrate the UC Connector SIP Server with OCS for IM functionality, in a UC Connector deployment.

Task Summary: Integrating UC Connector with Genesys IM

Objective	Actions
1. Verify the Genesys IM solution.	The Genesys IM Solution uses many of the same prerequisite Genesys components as those used by UC Connector: • Genesys Stat Server • Universal Routing • Interaction Workspace or customized desktop These components should already be installed and configured as part of the prerequisites for deploying UC Connector. Genesys IM also requires: • SIP Server **For more information, see:** • *Genesys 7.6 Instant Messaging Solution Guide* • *Framework 8.1 SIP Server Deployment Guide*
2. Configure the Knowledge Worker DN.	Configure the Knowledge Worker `Extension` DNs with the following options: • `multimedia`—Set this option to `true` to allow IM interactions. • `sip-signaling-chat`—Set this option to `none` so that UC Connector handles the IM interaction. • `voice`—(Optional) Set this option to `false` for DNs that will only provide chat functionality. (for example, if you are separating chat and voice traffic across different switches). **Sample DNs** • Single DN for Both Chat and Voice • Separate Chat-Only DN

Task Summary: Integrating UC Connector with Genesys IM (Continued)

Objective	Actions
3. Customize the Chat window (optional).	You can customize the name for the Chat window. In the UC Connector Application, on the `Options` tab, `UC-Connector` section, you can configure the following options: • `chat-title`—Enter the name you want to appear in the regular chat window. • `chat-consult-title`—Enter the name you want to appear in the window that appears for consultation chat interactions.

Single DN for Both Chat and Voice

You can modify an existing Knowledge Worker voice DN to also handle the chat interaction. Or you can create a separate DN that handles chat only.

Figure 43 shows a DN that handles SIP voice plus SIP instant messaging (the `sip-signaling-chat` option is set to `none`).
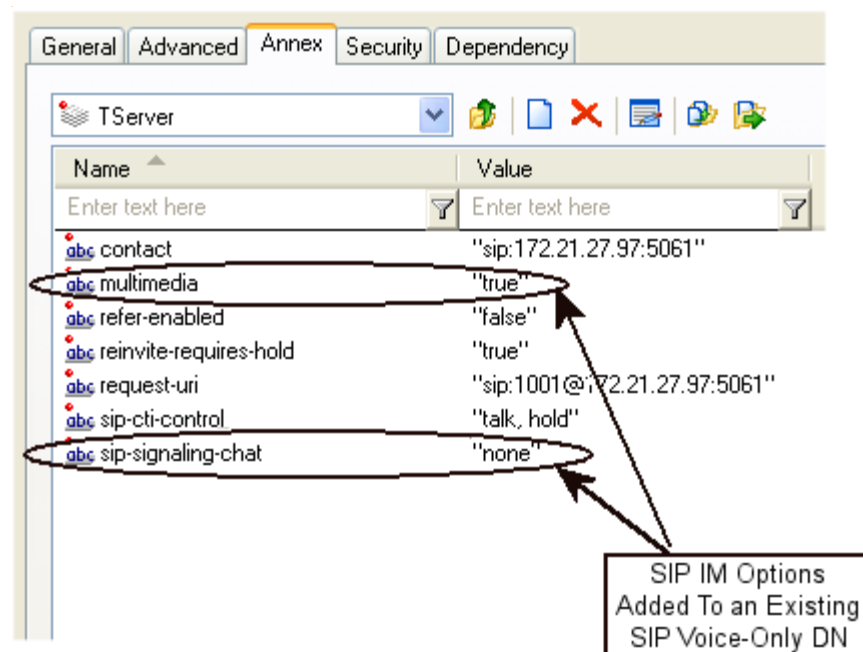
Figure 43: Sample of Options on a DN Handling SIP Voice and SIP IM

Separate Chat-Only DN

Figure 44 shows what this same DN object's options might look like if you intended for it to handle SIP instant messaging only (the `sip-signaling-chat` option is set to `none,` and the `voice` option is set to `false`).
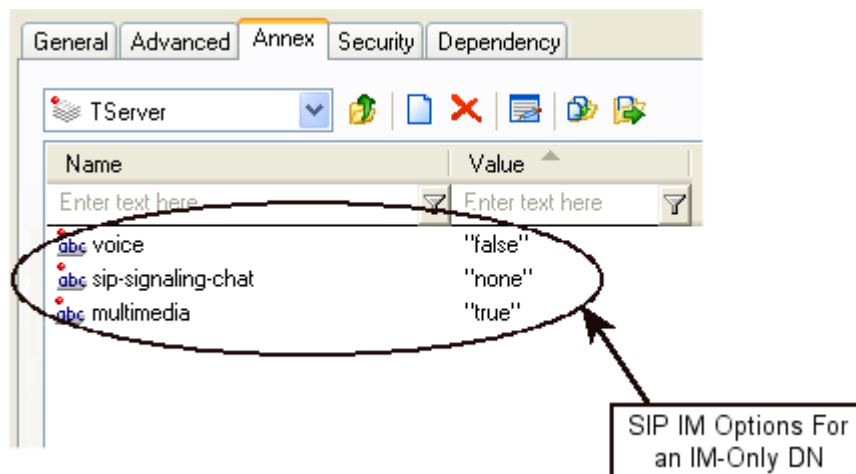
Figure 44: Sample of Options for a DN That Handles IM Only

Next Steps

- If you do not require secure communication in the deployment, continue at Chapter 6, "Configuring the Routing Strategies," on page 161.

- If secure communication is required, continue at one of the following:
  - Task Summary: Enabling MTLS Communication, on page 131.
  - Task Summary: Enabling TLS/Kerberos Secure Communication, on page 140.

Enabling IM through SIP Server-OCS

The following table shows the main steps required to integrate SIP Server with OCS for IM functionality, in a UC Connector deployment.

Task Summary: Enabling IM with Microsoft OCS

Objective	Actions
1. Install and configure SIP Server.	SIP Server is mandatory for Instant Messaging. If SIP Server is not already part of your deployment, install and configure SIP Server according to the procedures described in the *Framework 8.1 SIP Server Deployment Guide*. **Note:** No special configuration in the SIP Server `Application` object is required for IM.

Task Summary: Enabling IM with Microsoft OCS (Continued)

Objective	Actions
2. Integrate SIP Server with OCS.	Complete all the steps for Presence Integration with Microsoft Office Communications Server 2007, as described in the *Framework 8.1 SIP Server Deployment Guide*. These steps include: • Creating a Routing Point and adding a user for it in OCS. Used for presence integration. • Configuring a `Trunk` DN for OCS. • Configuring `Extension` DNs for each Knowledge Worker, to be used for IM interactions only. **Note**: Do not configure the `ocs-dn` option in the `Extension` DN. Presence is handled through UC Connector.
3. Customize the Chat window (optional).	You can customize the name for the Chat window. In the UC Connector Application, on the `Options` tab, `UC-Connector` section, you can configure the following options: • `chat-title`—Enter the name you want to appear in the regular chat window. • `chat-consult-title`—Enter the name you want to appear in the window that appears for consultation chat interactions.

Next Steps

• If you do not require secure communication in the deployment, continue at Chapter 6, "Configuring the Routing Strategies," on .

• If secure communication is required, continue at one of the following:
   ◆
   ◆ .

Enabling Secure Communication

UC Connector supports the Kerberos protocol for establishing secure connections—using simple Transport Layer Security (TLS) or Mutual Transport Layer Security (MTLS)—between the UC Connector application and Microsoft OCS. Secured mode is required for integrations with Microsoft Lync Server. MTSL is required for push presence status with Lync.

About TLS/Kerberos Security

If configured for TLS/Kerberos secure communication, when UC Connector registers with Microsoft OCS (by sending a SIP `REGISTER` request), OCS will use Kerberos authentication procedures to send sip `401 Unauthorized` or sip `407 proxy authentication required` in the following cases:

- UCC is using regular TCP connection, and the UC Connector host IP address has *not* been added to the Trusted Host list.

- UCC is configured to use TLS connection.

To configure TLS/Kerberos, see Task Summary: Enabling TLS/Kerberos Secure Communication, on page 140.

About MTLS

If configured for Mutual Transport Layer Security (MTLS), a shared trusted Certificate Authority (CA) on both the UC Connector host and the Lync deployment are used to establish secure communication. The certificates prove the identity of each server to the other.

To configure MTLS, see Task Summary: Enabling MTLS Communication, on page 131.

Enabling MTLS

Complete the following steps to enable MTLS between UC Connector and OCS.

Note: These procedures apply specifically to Microsoft Lync Server. The exact steps may differ slightly for integrations with Microsoft OCS 2007 R2.

Task Summary: Enabling MTLS Communication

Step	Actions
1. Add UCC host as trusted host/application in Lync	In the Lync `Topology Builder`, add the UC Connector host machine as a trusted application: • Enter FQDN for the UCC host. • Select `Single computer pool`. For a detailed procedure, see Procedure: Adding UCC Host as Trusted Host in Lync. • **Note**: For OCS 2007, skip this step.
2. Generate client certificate.	• Generate a client certificate using internal Certificate Authorities (CA). For a detailed procedure, see Procedure: Generating the Client Certificate.
3. Generate server certificate.	Generate the server certificate using one of two methods: • Procedure: Generating the Server Certificate using Lync Management Shell • Procedure: Generating the Server Certificate using Microsoft Management Console or CA Web Access A third method can be used to generate a certificate, but it must only be used in an isolated lab: • Procedure: Using the Lync Front End Server Certificate **Note:** The first method, using Lync Management Shell, is preferred.
4. Add certificate file to UCC installation.	Copy the exported file to a logical location on the machine where you installed the UC Connector server.

Task Summary: Enabling MTLS Communication (Continued)

Step	Actions
5. Configure secure SIP port in UCC.	In the UCC application, add the SIP port 5061 for security-enabled communication with Microsoft OCS. For a detailed procedure, see Procedure: Configuring a secure SIP port.

6. Modify Command Line Argument.	You must add the following parameters to the UC Connector command line argument:
	
	`-cert_store_file <path to .pfx file on UCC host>`
	
	`-cert_store_pass <password generated in Step 1>`
	
	`-cert_store_type pkcs12`
	
	For a detailed procedure, see Procedure: Modifying Command Line Arguments for MTLS.

Procedure:
Adding UCC Host as Trusted Host in Lync

Note: For OCS 2007 R2, you can skip this step. This requirement should already be completed as part of the basic integration. For details, see Procedure: Adding Authorized Hosts to Microsoft OCS 2007 R2, on page 106

Start of procedure

1. On the computer where Lync Server is installed, click `Start > All Programs > Microsoft Lync Server`, and then click `Lync Server Topology Builder`.

2. Right-click `Trusted application servers` and click `New Trusted Application Pool`.

3. Follow the wizard to create a trusted application pool for the UC Connector host machine.
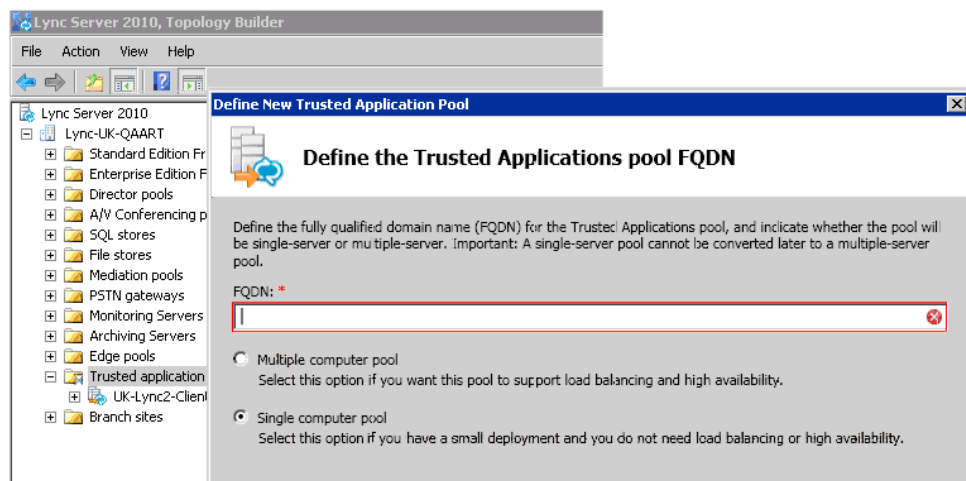
Figure 45: Trusted Applications pool FQDN

4. Enter the Fully Qualified Domain Name (FQDN) for the UC Connector host machine.

5. Select `Single computer pool.`

6. Click `Next` to continue the wizard.

End of procedure

Next Steps

Procedure: Generating the Client Certificate

Procedure: Generating the Client Certificate

Purpose:

To generate a regular client/user certificate used to trust servers in the domain, such as the Lync Front End server(s). This is the same type of certificate that is installed on a user's workstation to start a Lync client and to connect to the Lync server using TLS connectivity. It is not necessary to export private keys for this certificate or have private keys exportable.

Start of procedure

1. Request the certificate through Certification Authority (CA) Web Access
   `https://[server_name]/certsrv`

2. Select 'Download a CA certificate, certificate chain, or CRL'

3. Select 'Download CA certificate'

4. Save the certificate as "DER encoded binary X.509 (.CER)." For example, `CompanyA_Certificate.cer`

End of procedure

Next Steps

- On the host computer where Genesys is installed, open the Microsoft Management Console (MMC) or Internet Explorer to retrieve the certificate.
- Continue to one of the following:
  - Procedure: Generating the Server Certificate using Lync Management Shell
  - Procedure: Generating the Server Certificate using Microsoft Management Console or CA Web Access

Procedure:
Generating the Server Certificate using Lync Management Shell

Purpose: To generate a server certificate. This is the same type of certificate required by any server belonging to a Lync infrastructure (A/V MCU, Edge Server, Mediation Server).

Start of procedure

1. On the host computer where the Lync Front End Server is installed, open the Microsoft Lync Shell and type

```
Request-CsCertificate -New -Type Default -FriendlyName
"GenesysServerCertificate" -CA
"labdc01.companya.com\companya-LABDC01-CA" -ComputerFQDN
[server_name]-Verbose
```

Note: The [server_name] must match the FQDN of the host where UC Connector is running.

```
This will request the certificate through Lync. If
authorized/granted, it will be installed on the Certificate Store
(Personal) of the host where the request was issued.
```

2. Open the Microsoft Management Console (MMC):

a. Click `Start` > `Run`.

b. Type `MMC` and click `Ok`.

3. Add the certificates snap-in:

a. Go to `File` > `Add/Remove Snap-In`.

        b. Click `Add`.

        c. Select the `Certificates Snap-In` and click `Add`.

        d. Select `Local Computer` and click `Finish`.

4. Find the Genesys Server certificate that you want to export:

        a. Under the `Certificates` tree, locate your domain certificate; for example this could be in the `Personal` folder.

        b. Click `Certificates`.

        c. Right-click the certificate you want to export, select `All Tasks >Advanced Operations > Export`.

5. Follow the wizard to export the certificate to a .pfx file ("Personal Information Exchange - PKCS #12 (.PFX)").

        a. Choose `'Yes, export the private key'`.

        b. Choose `'Include all certificates in certificate path if possible'`.

> **Tip:** Do NOT select `'Delete Private key'`.

        c. Enter a password (take note of it). (Example: mnopqr)

        d. Select a location to save the file, then click `Finish` (Example: GenesysServer_Certficate.pfx)

6. When you get the message "`The export was successful`", click `OK`.

End of procedure

Next Steps

*   Place the exported file in a logical location on the UC Connector host machine.
*   After the certificate is moved to the UCC host, continue at Procedure: Configuring a secure SIP port.

Procedure:
Generating the Server Certificate using Microsoft Management Console or CA Web Access

Purpose: To generate a certificate for the host running UC Connector with the Microsoft Management Console or Certification Authority (CA) Web Access.

Note: Note that such a certificate template may not exist by default at the Certification Authority level (certificate template including Server Authentication as enhanced key usage and allowing Private keys to be exported). If operational policies permit it, a copy of the "Web Server" certificate template can be made, adding permission to export Private keys. This can be achieved on the Certification Authority host running the client tool "certtmpl.msc ".

Start of procedure

1. On the host computer where Genesys is installed, request the certificate through CA Web Access:

   `https://[server_name]/certsrv`

2. Select `Request a certificate`

3. Select `Advanced certificate request`

4. Select `Create and submit a request to this CA`
   - Type - Select a `Server Template with Private Keys exportable`
   - (NDLR: custom Server template with Private Keys exportable)
   - Name: `demosrv.genesyslab.com` (Subject)
   - New keyset: Microsoft RSA, Key Size 2048, Mark Keys as exportable
   - Friendly Name: (Example: GenesysServerCertificate)

5. Export the certificate and save it into a .pfx file (Example: `GenesysServer_Certficate.pfx`). [password - Example: `mnopqr`)]

End of procedure

Next Steps

- Place the exported file in a logical location on the UC Connector host machine.

- After the certificate is moved to the UCC host, continue at Procedure: Configuring a secure SIP port.

Procedure:
Using the Lync Front End Server Certificate

Purpose: To generate a server certificate for use in a lab environment.

Start of procedure

1. On the host computer where Lync Front End Server is installed, open the Microsoft Management Console (MMC):

   a. Click `Start` > `Run`.

   b. Type `MMC` and click `Ok`.

2. Add the certificates snap-in:

   a. Go to `File` > `Add/Remove Snap-In`.

   b. Click `Add`.

   c. Select the `Certificates Snap-In` and click `Add`.

   d. Select `Computer Account` and click `Finish`.

   e. Select `Local Computer` and click `Finish`.

3. Find the Domain certificate that you want to export.

   a. Under the `Certificates` tree, locate your domain certificate, for example in the `Personal` folder.

   b. Click `Certificates`.

   c. Right-click the certificate you want to export, select `All Tasks` > `Advanced Operations` > `Request New Certificate with the Same Key`.
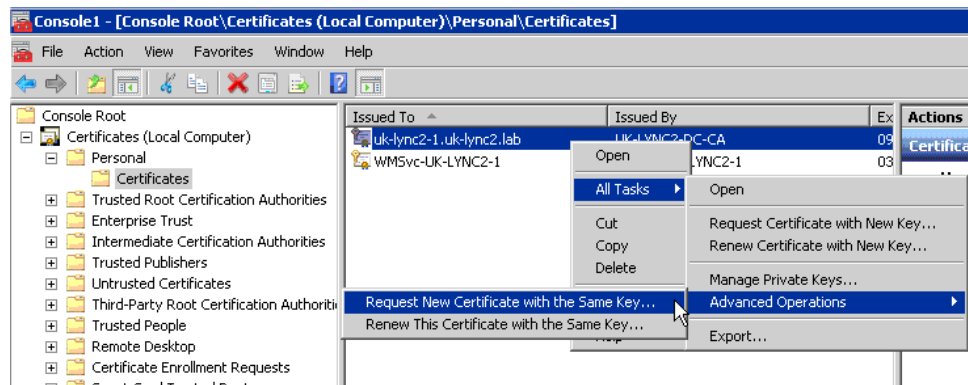
Figure 46: Requesting a new certificate

4. Follow the wizard to export the certificate to a .pfx file.

   a. Choose `'Yes, export the private key'`.

   b. Choose `'Include all certificates in certificate path if possible'`.

   > **Tip:** Do NOT select `'Delete Private key'`.

     c.   Enter a password (take note of it).

     d.   Select a location to save the file, then click `Finish`.

5. When you get the message "`The export was successful`", click `OK`.

End of procedure

Next Steps

- Place the exported file in a logical location on the UC Connector host machine.

- After the certificate is moved to the UCC host, continue at Procedure: Configuring a secure SIP port.

Procedure:
Modifying Command Line Arguments for MTLS

Prerequisites

- You noted the password that you created in Step 3 .

- You noted the file location where you placed the `.pfx` file.

Start of procedure

1. Go to `Start Info` tab > `Command Line Arguments` in the UC Connector application.

2. Add the following parameters to the existing command line argument:

`-cert_store_<path to .pfx file on UCC host>`

`-cert_store_pass <password generated in Step 3>`

`-cert_store_type pkcs12`

`-key_store_file <path to the Java keystore file>` — This file contains the collection of CA certificates trusted by the application process (trust store). If a trust store location is not specified, the SunJSSE implementation uses a keystore file in the following locations (in order):

     i. $JAVA_HOME/lib/security/jssecacerts

     ii. $JAVA_HOME/lib/security/cacerts

`-key_store_pass <password to unlock the keystore file>`

`-ket_store_type jks`
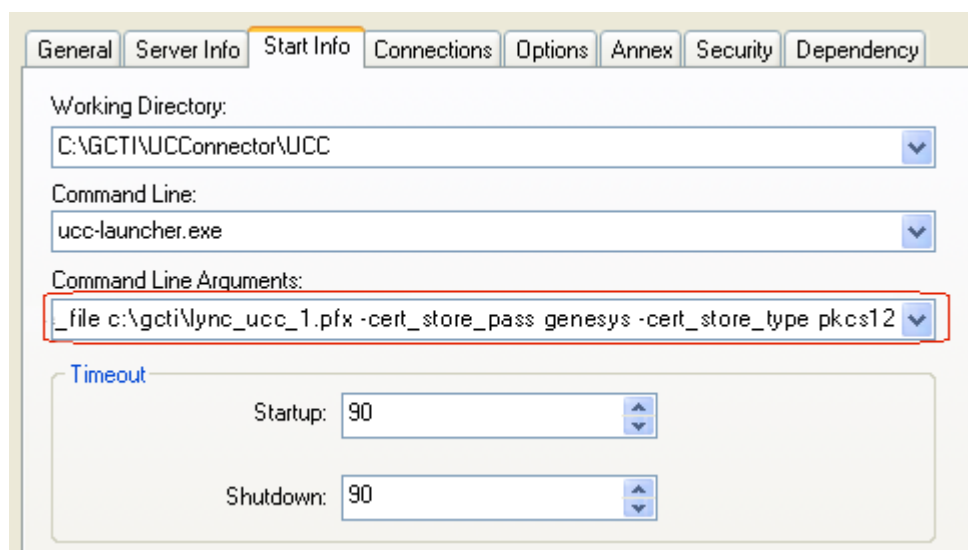
Figure 47: Modifying the Command Line Parameters for MTLS

3. Click OK to save.

4. If you are planning on starting UC Connector from the batch file, you must also modify the `startup.bat` file with the certificate parameters.

   For example, the following `startup.bat` is appended with these sample certificate values:

   ```
   -cert_store_file c:\gcti\lync_ucc_1.pfx
   -cert_store_pass genesys
   -cert_store_type pkcs12
   ```

   ```
   @echo off

   rem ------------------------------------------------------------------------------------
   rem Copyright (C) 2011 Genesys Telecommunications Laboratories, Inc.
   rem
   rem startServer.bat file for UC Connector, version 8.0.100.12
   rem ------------------------------------------------------------------------------------

   @TITLE UC Connector v. 8.0.100.12: Application UCC_Saran
   ucc-launcher.exe -host 10.10.10.0 -port 2020 -app UCC_Saran -l 7260@135.80.170.
   c:\gcti\lync_ucc_1.pfx -cert_store_pass genesys -cert_store_type  pkcs12
   ```

End of procedure

Next Steps

• If you came to this task from the Lync procedures, you might still need to integrate with Genesys Routing. If so, continue at

- Otherwise, configure the routing strategies used to deliver interactions to the Knowledge Worker. See Chapter 6, "Configuring the Routing Strategies," on page 161.

Enabling TLS/Kerberos

Complete the following steps to enable TLS/Kerberos authentication between UCC and OCS.

Note: These procedures apply specifically to Microsoft OCS 2007 R2. The exact steps may differ slightly for integrations with Microsoft Lync Server.

Task Summary: Enabling TLS/Kerberos Secure Communication

Objective	Actions
1. Configure OCS server for Kerberos.	In `Front End properties` window, select Kerberos as the authentication protocol. For a detailed procedure, see Procedure: Enabling Kerberos security in Microsoft OCS.
2. Configure OCS users for Kerberos.	Open the Active Directory in the host server (Windows). 1. Open the `User Properties` window for the user that represents UC Connector in OCS. 2. Go to the `Account` tab and select `Use kerberos DES encryption types`. This gives the 168 bit encryption type (triple DES) that is used by Java Kerberos. For a detailed procedure, see Procedure: Configuring Kerberos security in Active Directory.
3. Configure the UC Connector application for Kerberos.	In the `Microsoft-OCS` section, create a new option `password`, and enter a password for the user that represents UC Connector in OCS. For a detailed procedure, see Procedure: Creating a password for Kerberos security
4. Export the OCS host's trusted certificate.	Use Internet Explorer to export a trusted certificate (.cer) file. For a detailed procedure, see Procedure: Exporting the trusted certificate from the OCS host.

Task Summary: Enabling TLS/Kerberos Secure Communication (Continued)

Objective	Actions
5. Add certificate file to the UCC installation.	1. Copy certificate file to JDK. 2. Use the keytool command to create the keystore file for UC Connector. 3. Add the keystore file to the UCC install directory. For a detailed procedure, see Procedure: Adding the certificate to the UC Connector installation.
6. Create the Kerberos configuration file.	1. Configure the `krb5.conf` file. 2. Place the file in the default location `<ucc_root>/etc`. For a detailed procedure, see Procedure: Creating the configuration file for Kerberos security.
7. Configure secure SIP port in UCC.	Add the SIP port `5061` for security-enabled communication with Microsoft OCS. For a detailed procedure, see Procedure: Configuring a secure SIP port.

8. Modify Command Line Arguments.	You must add the following parameters to the UC Connector command line argument:
	`-krb_conf_file <path to krb5.conf file>`
	`-cert_store_pass <password generated in step 5>`
	`-cert_store_file <path to keystore file moved in step 5>`
	For a detailed procedure, see Procedure: Modifying Command Line Parameters for TLS.

Procedure:
Enabling Kerberos security in Microsoft OCS

Start of procedure

1. On the host computer where Microsoft OCS is installed, go to `Start > Administrative Tools > Office Communications Server 2007 R2`.

2. In the folder structure for OCS, click through `Enterprise pools` > *your configured pool* > `Properties` > `Front End Properties`.

3. On the `Authentication` tab, select either of the following choices for authentication protocol:
   • `Both NTLM and Kerberos`

- Kerberos

Note: The NTLM protocol is not supported.
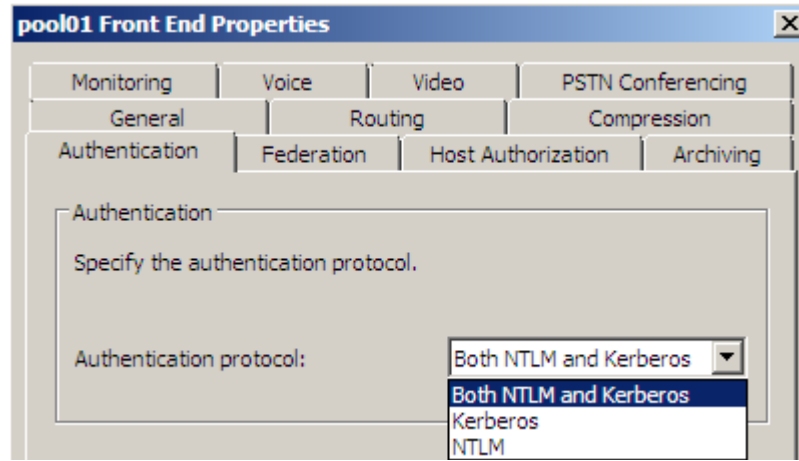
Figure 48: Selecting Kerberos Protocol

End of procedure

Next Steps

Procedure:
Configuring Kerberos security in Active Directory

Purpose: To configure Kerberos security for the user that represents the UC Connector environment in the Microsoft OCS deployment.

Start of procedure

1. Access the user properties in Active Directory.

2. Locate the OCS user that represents your UC Connector environment, right-click this user, and then select Properties.

3. On the Account tab, under the Account options field, select Use Kerberos DES encryption for this account.
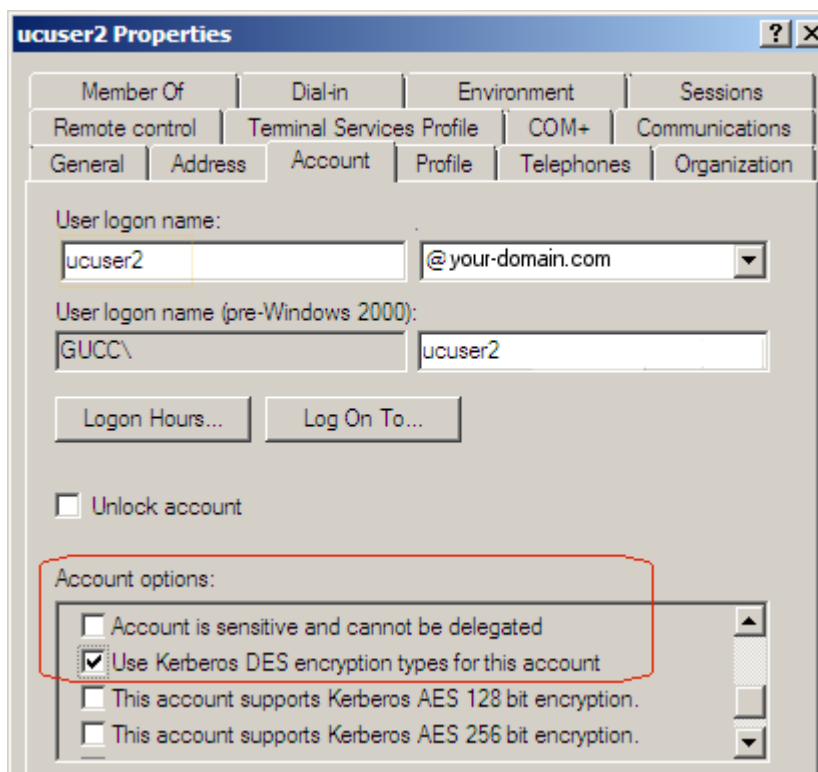
Figure 49: Configure OCS User In Active Directory

End of procedure

Next Steps

*   Procedure: Creating a password for Kerberos security

Procedure:
Creating a password for Kerberos security

Prerequisites

*   You are logged in to Configuration Manager.
*   A UC Connector `Application` object, which as been configured according to Procedure: Creating the UC Connector Application object.
*   An account/user that represents the UC Connector has been created in Microsoft OCS. For example, `ocs-ucc`.
*   You will need the password configured for this user in Active Directory. If you do not know the password, you might have to reset it. Right-click the user in Active Directory and select `reset password`.

Start of procedure

1. Go to `Environment` > `Applications` and double-click the UC Connector `Application` object.

2. Go to the `Options` tab.

3. In the `Microsoft-OCS` section, configure the following option.

Table 24: UC Connector—Password Option

Option Name	Value	Description
password	String	Set this option to the password configured for the OCS user in Active Directory. This is your Kerberos password, required for Kerberos authentication between the components.

End of procedure

Next Steps

• Procedure: Exporting the trusted certificate from the OCS host

---

Procedure:
Exporting the trusted certificate from the OCS host

Prerequisites

• A valid certificate to authenticate the OCS server has been installed as part of the domain configuration. This can be obtained using internal Certificate Authorities (CA), a domain configuration utility, or from a third party (for example, Verisign), according to your security policy.

Start of procedure

1. On the host computer for Microsoft OCS, open Internet Explorer.

2. Go to `Tools` > `Internet Options` > `Content` > `Certificates` and select the `Trusted Root Certification Authorities` tab.

3. Highlight the certificate that has been issued to this host computer, then click `Export`.
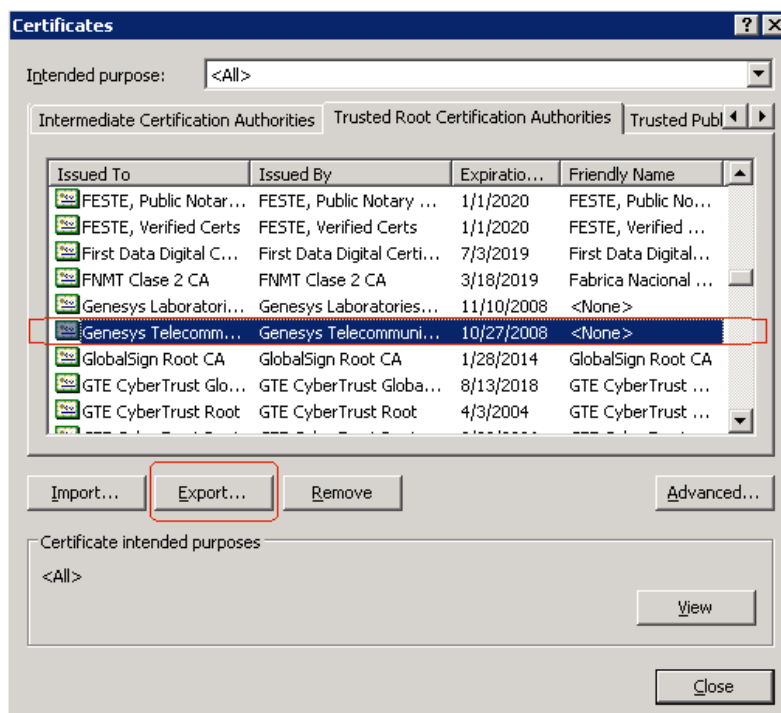
Figure 50: Exporting the Trusted Certificate

4. When you are prompted to do so during the export operation, under `Export format`, select `DER Encoded Binary x.509(*.cer)`.

5. When asked to enter a name for the certificate, enter any useful name. There are no mandatory formats. For example, if you enter `UCC_certificate`, the export operation will create a file called `UCC_certficate.cer`.

End of procedure

Next Steps

• Procedure: Adding the certificate to the UC Connector installation

---

Procedure:
Adding the certificate to the UC Connector installation

Prerequisites

• The trusted certificate has been exported from the Microsoft OCS host computer.

Start of procedure

1. Copy the certificate that you created in Procedure: Exporting the trusted certificate from the OCS host.

2. Place this certificate in the `JDK\bin` directory of your prerequisite JDK installation.

3. From the `JDK\bin` directory, run the following command:

    ```
    keytool -import -alias "certificate_name" -file <certifcatefile.cer>
    -keystore <output_file.jks>
    ```

    For example,

    ```
    keytool -import -alias "ucc-cert" -file UCC_Certificate.cer
    -keystore UCC_store.jks
    ```

    Note: Take note of the password that was generated during the keytool process. You will add this later to the `Command Line Parameters` of the UC Connector application.

Table 25: Keytool Command Parameters

parameter	description
-alias	Enter an alias for the certificate. It can be anything; there are no restrictions.
-file	Enter the file name of the exported certificate
-keystore	Enter the name of the file that will be created as a result of running this keytool command.

4. Place this file in a logical location. For example:

    ```
    <ucc_root>\etc\MYSTORE.jks
    ```

    Note: Take note of the location where you save this file. You will need to add a parameter for this path to the `Command Line Arguments` of the UC Connector application.

End of procedure

Next Steps

• Procedure: Creating the configuration file for Kerberos security

Procedure:
Creating the configuration file for Kerberos security

Start of procedure

1. If Kerberos is not already configured for your environment, on the UC Connector host computer, navigate to the `etc` folder in the installation directory and open the sample `krb5.conf` file.

   For example, the default path to the `etc` folder would be:

   `C:\GCTI\UCConnector\<your_UC_Connector>\etc`

2. Modify the `krb5.conf` file with the following information (change the text in **bold** to match your environment):

```
[libdefaults]
   default_realm = YOUR-OCS-DOMAIN.COM
#    default_checksum = rsa-md5
[realms]
     YOUR-OCS-DOMAIN.COM = {
        kdc = SERVER1.YOUR-OCS-DOMAIN.COM
     }
[domain_realm]
   .your-ocs-domain.com = YOUR-OCS-DOMAIN.COM
```

   The following table provides more information about the parameters used in this file.

Table 26: Kerberos Configuration File—Parameters

Parameter	Description
default_realm	Set this option to the OCS domain as per the UCC setup.
	This is used in cases where a user in Active Directory is configured without a specified domain. For example, in cases where clients are connecting from computers that are not part of the domain.
[realms]	This is a list of all the domain names included in the OCS environment.
kdc=	Set this option to the FQDN or IP address for the Key Distribution Center (KDC), typically the same computer hosting the domain or domain controller.
[domain_realm]	Use this to map domains to realms in which Kerberos authentication is running (typically used in multi-domain environments).
	In our sample, the realm `.your-ocs-domain.com` is mapped to the domain `YOUR-OCS-DOMAIN.COM`.

3. Save the `krb5.conf` file.

> **Note:** Take note of the location where you save this file. You will need to add a parameter for this path to the `Command Line Arguments` of the UC Connector application.

End of procedure

Next Steps

- Procedure: Configuring a secure SIP port

Procedure:
Configuring a secure SIP port

Purpose: To configure the UC Connector `Application` object to connect to Microsoft OCS using the secure port.

By default, the Microsoft OCS installation uses port `5060` as the unsecure SIP port, and port `5061` as the secure port. For secure communication, you must configure UC Connector to use this secure port `5061` instead for SIP traffic.

Start of procedure

1. In the UC Connector `Application` object, on the `Server Info` tab, add a new port for SIP communication with OCS, using the port number `5061`.
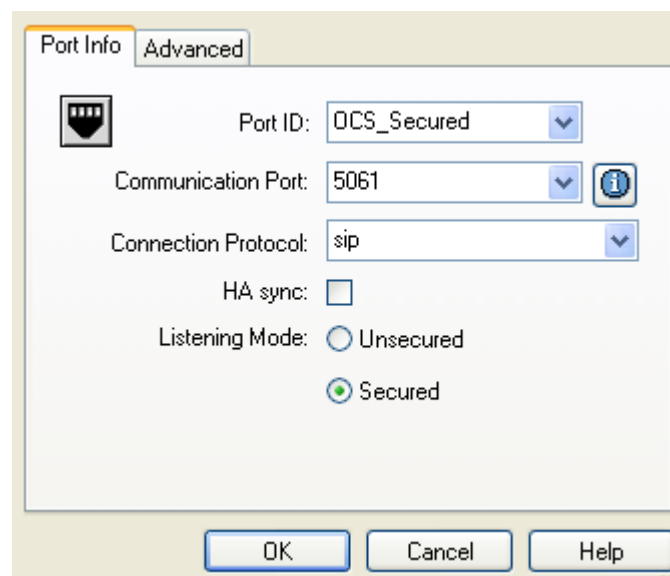
Figure 51: Adding Secure SIP Port to UC Connector

2. On adding this port, you may see the following `Warning`. For this configuration, you can ignore this warning. Click `Yes` to continue.
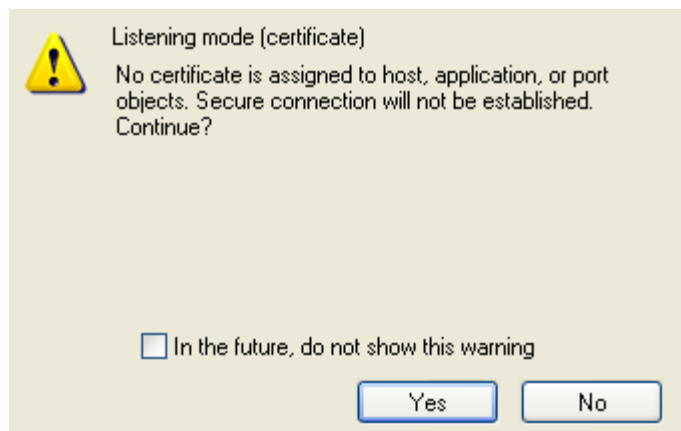
Figure 52: No certificate warning

End of procedure

Next Steps

- For TLS/Kerberos, continue at Procedure: Modifying Command Line Parameters for TLS.

- For MTLS, continue at Procedure: Modifying Command Line Arguments for MTLS. Or go back to "Modify Command Line Argument." in the Task Summary: Enabling MTLS Communication, on page 131.

Procedure:
Modifying Command Line Parameters for TLS

Prerequisites

- You noted the password that you created in Procedure: Adding the certificate to the UC Connector installation, on page 145.

- You noted the file location where you placed the keystore file in Procedure: Adding the certificate to the UC Connector installation, on page 145.

Start of procedure

1. Go to `Start Info` tab > `Command Line Arguments` in the UC Connector application.

2. Add the following parameters to the existing command line argument:

`-krb_conf_file <path to krb5.conf file>`

`-cert_store_pass <password generated in Step 5>`

`-cert_store_file <path to keystore file moved in Step 5>`
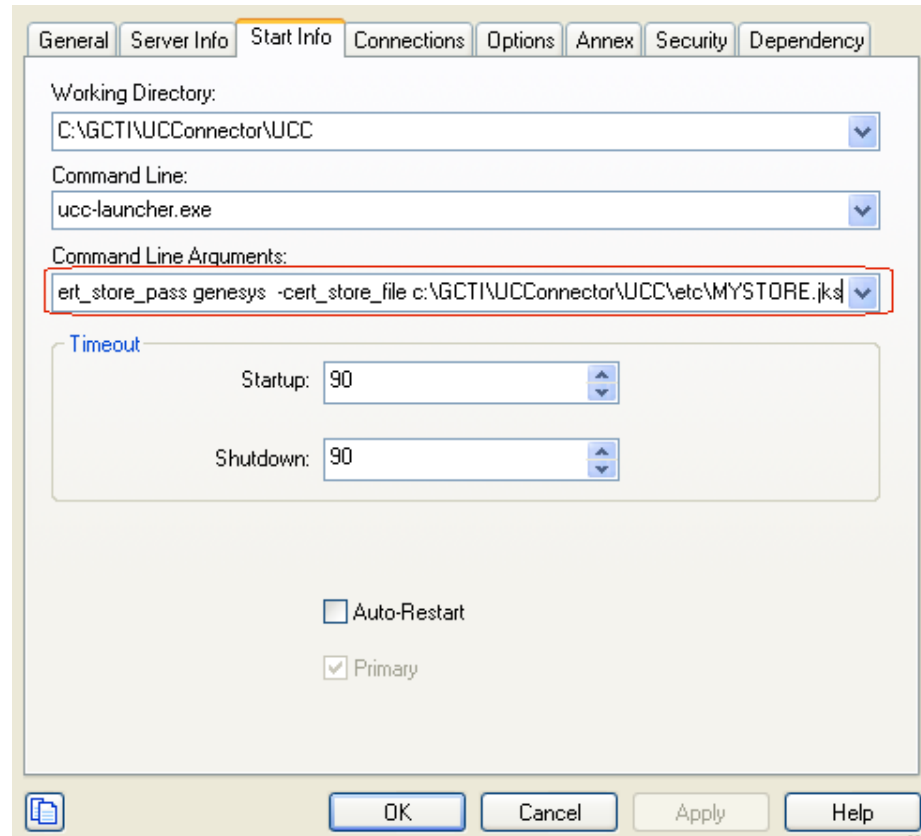
Figure 53: Modifying the Command Line Parameters

3. Click `OK` to save.

End of procedure

Next Steps

- Configure the routing strategies used to deliver interactions to the Knowledge Worker. See Chapter 6, "Configuring the Routing Strategies," on page 161.

- If you came to this task from the Lync procedures, you might still need to integrate with Genesys Routing. If so, continue at Task Summary: Integrating with Genesys Routing, on page 119.

Customizing UC Connector

UC Connector supports the following customization:

- Help Button — Enable the Help buttons on various UC Connector client windows.

- Default Language — Modify the language used in the UC Connector user interface.

- Audio — Enable an audio file to play when UC Connector displays a Preview or Ringing pop-up window.

- Hotkeys — Configure hotkeys to control accepting or rejecting a call when UC Connector displays the Preview window.

- Knowledge Worker States — Customize the "agent states" displayed in the UC Connector Web Client drop-down menu.

- Redirect Number — Enable a Knowledge Worker or Administrator to define an external telephone number to recieve interactions.

- After Call Work — Enable a Knowledge Worker to enter the After Call Work state.

Customizing the Help Buttons

The following table describes the steps required to enable an active Help button in the various UC Connector client windows. For general information about this feature, see "How It Works—Customized Help" on .

Task Summary: Customizing the Help Button

Objective	Related Procedure and Action
• Customize the Help Buttons.	In the `UC-Connector` section of the UC Connector `Application` object, configure any of the following: • `help-login-url` — Enter the path the Help file for the Login screen. For example, the path to the sample help file is: • /help/login.html • `help-interaction-url` — Enter the path to the Interaction window Help file. For example, `/help/interation.html` • `help-callcontrol-url` — This configuration applies to both the Preview and Interaction windows. For example, `/help/callcontrol.html` **Key Notes** • Sample help files are included on the product CD. You can find them under the `documentation/help` folder. • You can move these sample help files to a network-accessible location (point the help-url options to this location). Or you can create help files of your own. • By default, these options are not configured. You can enable any or all of these buttons. Only if enabled will a particular Help button appear. • For external help files, use a fully qualified URL. For example, `http://www.companyhelp.com`

Changing the Default Language

The following table describes the steps required to modify the language used by the UC Connector client interface. For general information about this feature, see "How It Works—Customized Languages" on .

Task Summary: Changing the Default Language

Objective	Related Procedure and Action
• Change the default language.	In the `UC-Connector` section of the UC Connector `Application` object, configure the following: • `locale` — Enter a two-character language code. For example, the default language (English) uses the code `en`. For a list of supported languages and character codes, see "Supported Languages" on page 48.

Enabling Audio on Preview or Ringing

The following table describes the steps required to enable an audio file to play, in a loop, when UC Connector displays a Preview or Ringing pop-up window..

Task Summary: Enabling audio on preview or ringing

Objective	Related Procedure and Action
• Enable audio when a preview or ringing pop-up is displayed.	In the `UC-Connector` section of the UC Connector `Application` object, configure one of the following: • `audio-on-preview`—Enter the location of the audio file. For example, a file path to the UC Connector installation or a URL to some other network-accessible location. • `audio-on-ring`—Enter the location of the audio file. For example, a file path to the UC Connector installation or a URL to some other network-accessible location. **Key Notes** • To specify the location of a file in the UC Connector installation, place the file in the `UCC-install/webapps` directory or any of its subdirectories. For example, if an audio file 'ring.mp3' is placed in `UCC-install/webapps/audio/`, then the value for the `audio-on-ring` or `audio-on-preview` option should be `/audio/ring.mp3`. • These options are only applicable when UC Connector is used in non-gateway mode (`presence-gateway-mode` is set to `false`). • This feature supports .mp3, .ogg and .wav formats, but not all audio formats are supported by all browsers. HTML5 is used to play the audio file, which is not supported by Internet Explorer 8. For information on audio formats and browser support as of the time this document was last updated, see Table 27.

Table 27: Audio Formats and Browser Support

Browser	MP3	Wav	Ogg
Internet Explorer 8	No	Yes	No
Internet Explorer 9+	Yes	No	No
Firefox 4.0+	No	Yes	Yes
Google Chrome 6+	Yes	Yes	Yes
Apple Safari 5+	Yes	Yes	No

Configuring Hotkeys for Interaction Preview

The following table describes the steps required to configure hotkeys to control accepting or rejecting a call UC Connector displays the Preview window.

Task Summary: Enabling audio on preview or ringing

Objective	Related Procedure and Action
• Configure hotkeys for interaction Preview.	In the `UC-Connector` section of the UC Connector `Application` object, configure the following: • `preview-shortkey-accept` — Set to an alpha-numeric or ASCII number format to represent the key used to accept the call when the Preview window is displayed. • `preview-shortkey-reject` — Set to an alpha-numeric or ASCII number format to represent the key used to reject the call when the Preview window is displayed. **Key Notes** • The user is still able to click the corresponding buttons in the Preview window. • The Preview window must be in focus for the hotkeys to function. • ASCII decimal number format — The string must start with a hash (#) as the first character: ◆ `#32` Only a single digit can be processed with this ASCII deimal number format. • Alpha-numeric format — The number of characters should not exceed 255, and the first character '`#`' should be specified as '`##`'.The string can be a combination of uppercase and lowercase characters representing a single keystroke: ◆ `QqWwEeRrTt`

Customizing Knowledge Worker States

The following table describes the steps required to configure customized Knowledge Worker states and display customizable menus. For general information about this feature, see "How It Works—Customized Knowledge Worker States" on page 49.

Task Summary: Customizing Knowledge Workerstates

Objectives	Related Procedure and Action
1. Modify or override the presence definition document.	The UC Connector application is initially installed with a default presence definition document called presence.xml file. This file contains the two states available in releases prior to 8.0.300: • Set Do Not Disturb On • Set Do Not Disturb Off You can modify this file or create your own XML file to override the default. See Presence definition document for details.
2. Enable customized Knowledge Worker states.	In the `UC-Connector` section of the UC Connector `Application` object, configure the following: • presence-location — Set to the location of the presence.xml file.

Presence definition document

You can provide an XML file with presence definitions that can override the default definitions. This file is recognized as a resource file and its location is defined with the `presence-location` option.

The default presence definition document is called `presence.xml`. This file represents the custom states seen by the agent in releases prior to 8.0.300, which provides backwards compatibility. See Appendix, "Presence Definition Document Examples," on page 215 for the default `presence.xml` file and other examples.

Requirements and restrictions

XML Document If the namespace of the loaded XML definition is not matched, UC Connector considers the XML definition to be invalid. If the XML definition is invalid, UC Connector ignores it and continues to use the previous valid definition.

UC Connector considers an XML document to be valid if at least one state definition is valid.

UC Connector only uses the first root element with the name "presence" for all state definitions.

State ID   The State ID must not contain whitespace characters and it must be unique. If there are multiple states with the same ID, only the first definition is used.

The following predefined names cannot be used with the state:

- predefLogoutMenu
- predefLoggedOut
- predefOutOfService
- predefReady
- predefPartReady
- predefNotReady
- predefDND

Icons   There are two types of icons used in the web client:

- Main icons to represent agent status in the agent status indicator on the main panel of the web client. These icons must be 16 by 16 pixels. The icons can be specified as a local or URL location, or the ID of a standard icon can be provided. Table 28 lists predefined icon ID URIs you can use:

Table 28: Predefined icon ID URIs

Icon ID URI	Icon description
tag:ucc.genesyslab.com,2013:icons/status/ready	Round green shape
tag:ucc.genesyslab.com,2013:icons/status/partready	Round part green and part orange shape with vertical divider
tag:ucc.genesyslab.com,2013:icons/status/notready	Round red shape
tag:ucc.genesyslab.com,2013:icons/status/dnd	Red "Stop" pictogram
tag:ucc.genesyslab.com,2013:icons/status/logout	Red "Out" pictogram
tag:ucc.genesyslab.com,2013:icons/status/oos	White cross on grey background

- Badge icons to represent agent status in the agent status indicator Channel view (icons). These icons must be 10 by 10 pixels. UC Connector will use the location of the main icon to define the location for the bade icon. For example, if a main icon location is defined as /icons/iconName.png, then UC Connector will use the location /icons/iconName-badge.png for the badge icon.

UC Connector does not check the supplied images; any image is valid as long as it is supported by the client browser.

Display Text   The display text entries in the XML definition must follow these guidlines:

- The entry must have an element "stateText" in order to present the status text in agent status indicators.

- The entry must have an element named "menuText" to make it available for execution. If the element is missing, UC Connector will not add an entry in the drop-down menu of the web client.

- UC Connector only considers an entry valid if it has either a "menuText" or "stateText" element.

- Entries can be defined for any language, but only languages currently supported by UC Connector will be used. See "List of Supported Languages" on page 48.

- If an entry for default language is missing, UC Connector does not display the corresponding state in an unsupported language.

Genesys Agent State definitions

UC Connector processes the definitions for states in the order in which they are defined in the XML document. UC Connector uses the first found state that matches an event reported by T-Server to represent the agent state in the in the agent state indicators. For example, if the XML definition has an entry defining the state NotReady with no reasons, followed by a definition for the state Not Reay with reason, then UC Connector uses the first entry.

UC Connector ignores a work mode definition for an element "ready" with a value of "true". If the value is "false", UC Connector will accept the following work modes:

- manual

- acw

- legal

- auxwork

- away

- back

- nodisconnect

Note: The work modes might not be supported by T-Server.

Auto-state definition

UC Connector only accepts the following auto-state event definitions:

- preview-reject

- preview-timeout

- preview-redirect

UC Connector checks if the attributes "onevent" and "postevent" refer to an existing state definition. If no match is found, the value is ignored.

For each of the auto-states missing in the custom-defined XML document, UC Connector uses a predefined auto-state definition:

- When activated, the default auto-state "preview-reject" invokes the default DND state and the default Ready state when the timeout defined in the dnd-off-timeout option expires.

- When activated, the default auto-state "preview-timeout" invokes the default NotReady state.
- When activated, the default auto-state "preview-redirect" invokes the default DND state.

Note: UC Connector supports "postevent" only for the auto-state "preview-reject".

Enabling a Redirect Number

The following table describes the steps required for either a Knowledge Worker or an Administrator to enable an external redirect number. Enabling this feature allows Knowledge Workers to accept preview calls at the specified number. For general information about this feature, see "How It Works— External Number Redirect" on .

Task Summary: Enabling a redirect number

Objective	Related Procedure and Action
• Enable the redirect setup.	To allow all users to set and enable their own redirect number using the `Setup` menu option in the UC Connector web client: • In the `UC-Connector` section of the UC Connector `Application` object, set the option redirect-setup-enabled to `true`. To allow a specific user to set and enable his or her own redirect number: • In the `Persons > Annex > UC-Connector` section, set the option redirect-setup-enabled to true. This option overwrites the value of the application-level redirect-setup-enabled option. To set and enable the redirect number for a Person: • In the `Persons > Annex > UC-Connector` section, set the redirect-number option to the number. • In the `Persons > Annex > UC-Connector` section, set the redirect-enabled option to true.

Enabling After Call Work

When an agent enters the After Call Work state, the agent's presence state is preserved in Genesys until the agent uses the Lync client menu to change state, or the After Call Work timer expires. The agent's presence state is also propagated to the Lync server so that the agent's unavailability is reflected in

the corresponding Lync presence, with a configurable presence status and note values.

When the agent exits the After Call Work state (either automatically or manually), the agent's Lync presence state is set back to a value that is preserved from the Lync presence update. The agent's Genesys state is also updated with the corresponding value.

The following table describes the steps required to configure the presence and note values for the After Call Work and Legal Guard states.

Task Summary:

Objective	Related Procedure and Action
• Enable After Call Work.	In the `Microsoft-OCS` section of the UC Connector `Application` object, configure the following: • `presence-acw-note` — Enter the note UC Connector uses when an agent enters the After Call Work State. • `presence-acw-status` — Enter a positive integer between 1 and 18500.
• Enable Legal Guard.	In the `Microsoft-OCS` section of the UC Connector `Application` object, configure the following: • `presence-lg-note` — Enter the note UC Connector uses when an agent exits the After Call Work state and enters the Legal Guard state. • `presence-lg-status` — Enter a positive integer between 1 and 18500.

![Genesys logo]

6 Configuring the Routing Strategies

This chapter describes a few sample Universal Routing Server (URS) routing strategies, as configured in Interaction Routing Designer (IRD), which can be used to deliver interactions from an agent to the Knowledge Worker—including negotiation of the PreviewInteraction protocol for delivering the Preview Notification.

Note: This section does not provide detailed step-by-step instructions for designing routing strategies, but instead gives samples and key information that you can work from. If you need detailed information about using IRD, see the *Universal Routing 8.0 Deployment Guide*.

This chapter includes the following sections:

About the Key Routing Blocks

The following table gives an overview of the main blocks that are used in the configuration of the sample routing strategies described in this chapter.

Table 29: About the Key Routing Blocks

Routing Block	Key Actions
Attach data block	Used for adding UserData for the call, which will include any key-value pairs (KVPs) sent by the agent. For example, a KVP that: • Adds agent Notes on the interaction. • Specifies a particular KW to target for this call. **Limitation**: For agent notes that will appear in the Preview window, the text cannot contain a single (') mark. URS does not process this quotation mark as regular text.
SelectDN block	Used for building the expression that will choose the DN. • Point to the Stat Server instance installed in your environment.

Table 29: About the Key Routing Blocks (Continued)

Routing Block	Key Actions
BlockDN	Used for environments with multiple URS instances. BlockDN makes the Target DN selected by this strategy unavailable for the duration specified in this block, so that other URS instances do not attempt to route interactions to this DN.
Database blocks	These are the key blocks for the Preview Interaction. When configuring this block, you will: • Connect the strategy to the Custom Server module built into the UC Connector. • Specify the `PreviewInteraction` protocol message to be sent to the Custom Server. For example, `PreviewInterationRequest`, which asks Custom Server to initiate the preview. • Specify how to process the results that come back from the Custom Server. **Sample Screenshots** For some sample screenshots of the Database Wizard, see "PreviewInteractionRequest Database Block" on page 163. **About The Protocol Formats** For a list of formats to be used with the various PreviewInteraction protocol messages, see "About the Preview Interaction Protocol" on page 165.

PreviewInteractionRequest Database Block

The following figures show sample configuration screens from the database wizard for the `PreviewInteractionRequest` Database Block.

1. Select Custom Server
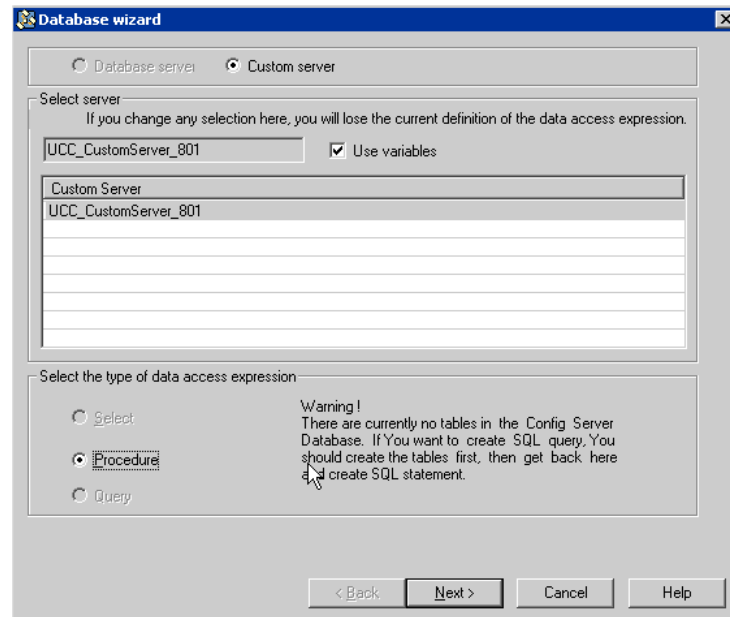
Figure 54: Database Wizard—Selecting Custom Server

2. Configuring the PreviewInteractionRequest Message
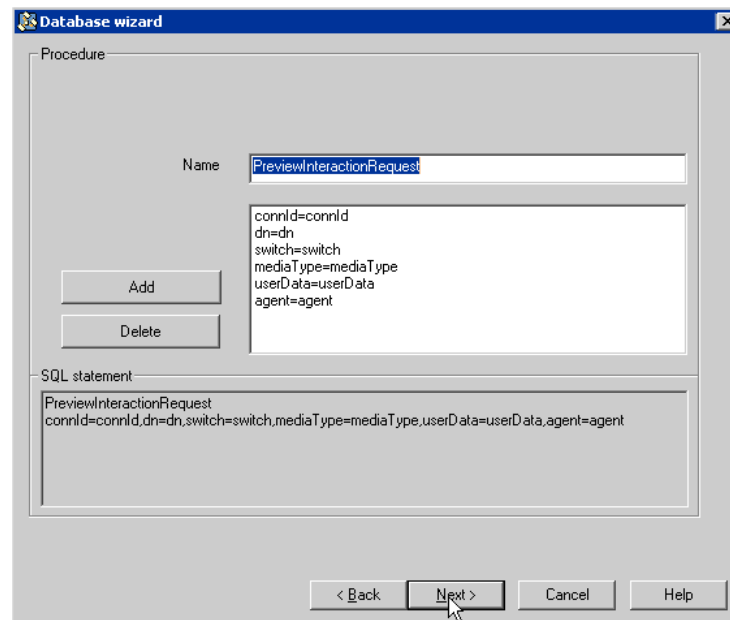
Figure 55: Database Wizard—Configuring the PreviewInteractionRequest

3. Assigning Values to the Results from Custom Server
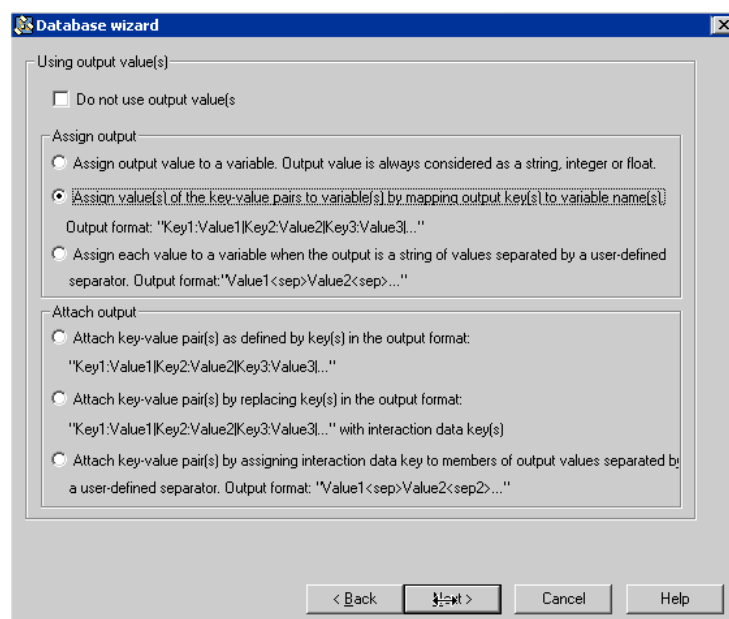
Figure 56: Database Wizard—Assigning Values to Custom Server Results

4. Mapping KVPs to Variables
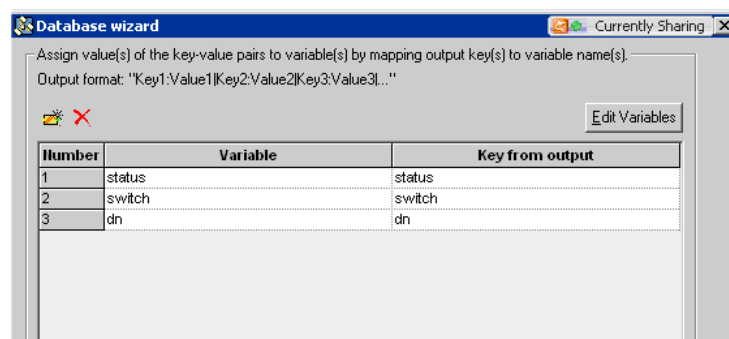
Figure 57: Database Wizard—Mapping KVPs to Variables

About the Preview Interaction Protocol

Table 30 shows the various `PreviewInteraction` protocol messages and the formats that go along with them.

Table 30: Preview Interaction Protocol Messages

Message	Format
PreviewIneractionRequest Initiates the Preview Interaction with the targeted Knowledge Worker.	agent connId dn mediaType switch userData
PreviewInteractionCancel Indicates to the Knowledge Worker that the interaction is canceled.	Contact ThisDN ConnID OtherDN MediaType UserData Message Status StatusMessage
PreviewInteractionMultiple Initiates the Preview Interaction with the Knowledge Worker group. This acts like a broadcast to all the agents in the group.	connId – Used as a key for this interaction, and added to any ICO reporting messages. mediaType – Media type as assigned by TServer/SipServer userData – Used for display, this may include the filter third-party window key. targets – A list of agents found from a source e.g Stat Server. The default delimiter "|" should be replaced with "^".

Routing to a Particular Knowledge Worker

Figure 58 shows a sample strategy that routes a call to a particular Knowledge Worker selected by the agent.
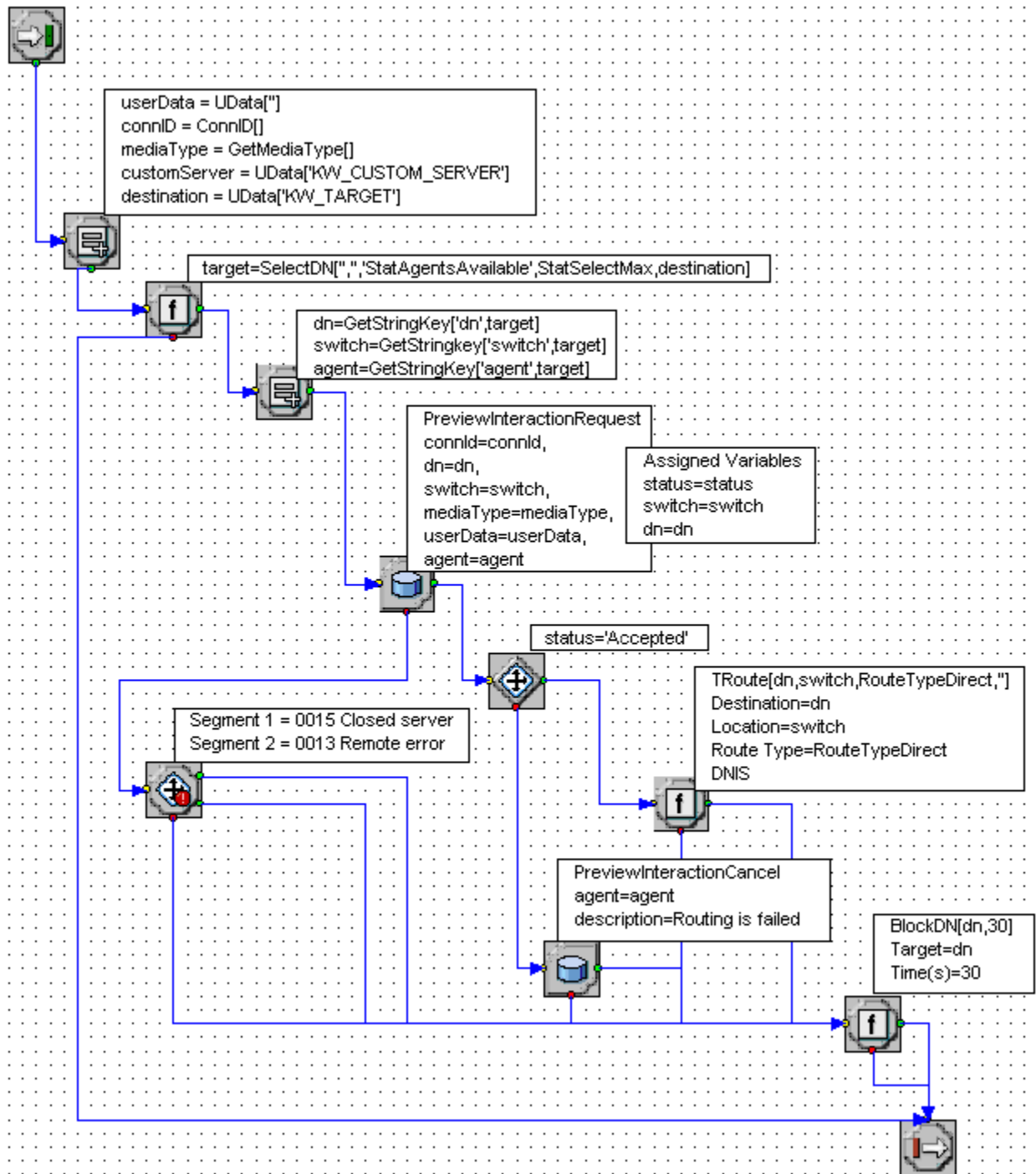
Figure 58: Contact Center to Particular Knowledge Worker

For details about the call flow that this strategy supports, see "Contact Center to Knowledge Worker" on page 31.

Routing with Round-Robin Selection

Figure 59 shows a sample strategy that routes a call to an available Knowledge Worker using a round-robin selection method.
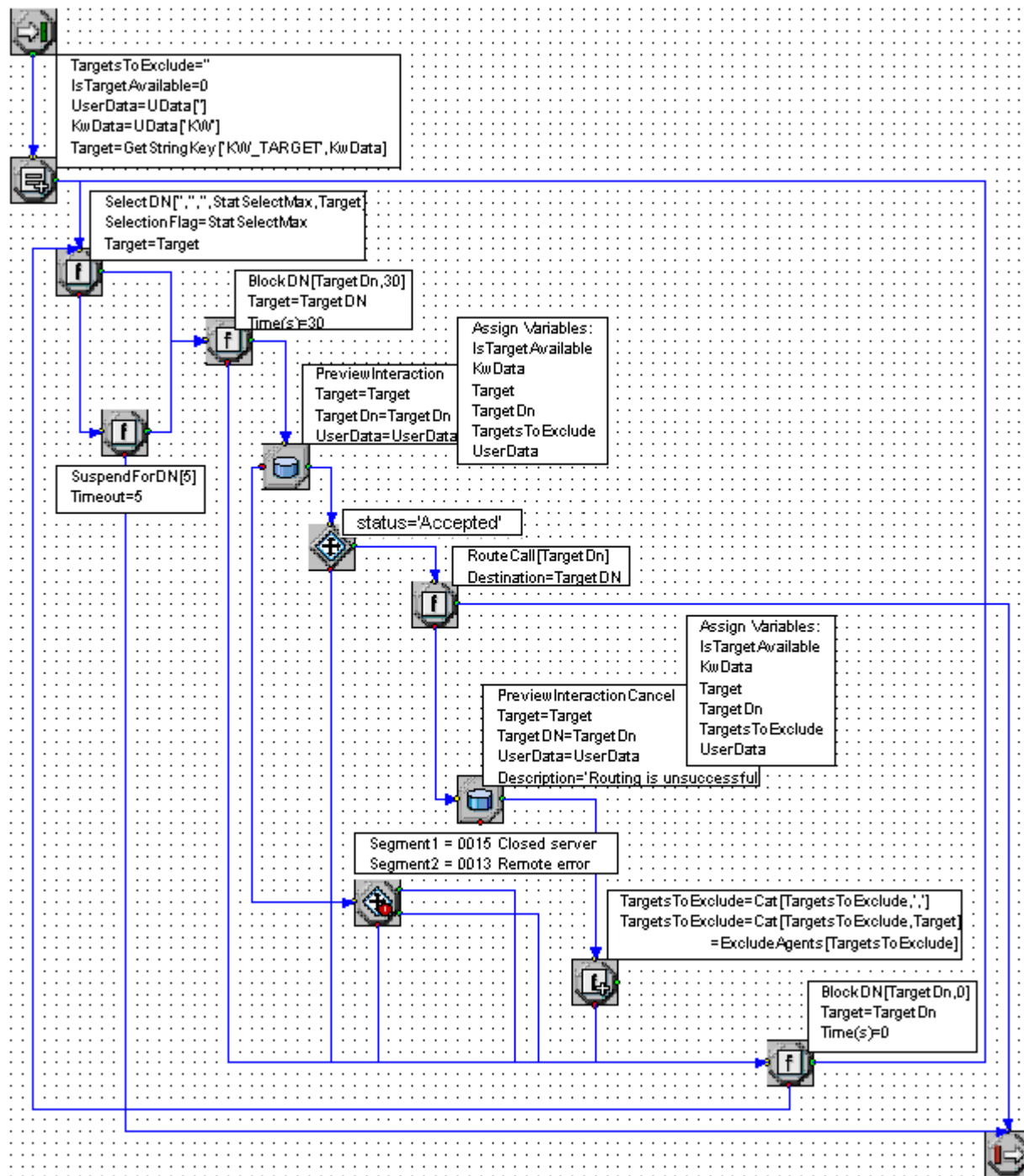
Figure 59: Round Robin Knowledge Worker Selection

For details about the call flow that this strategy supports, see "Contact Center to Knowledge Worker" on page 31.

Routing with Broadcast Preview

Figure 60 shows a sample strategy that broadcasts preview notifications simultaneous to a set of Knowledge Workers.
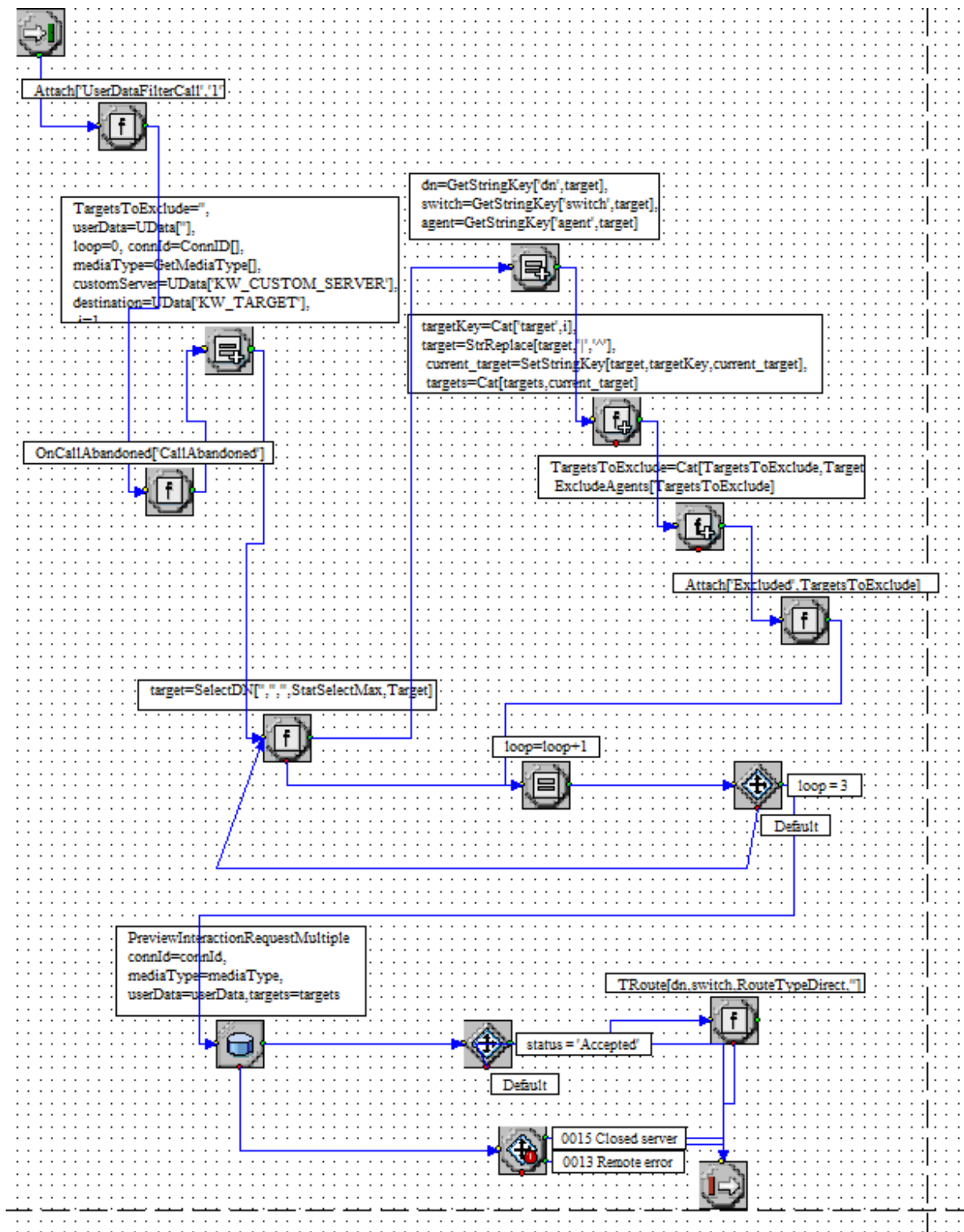
Figure 60: Broadcast Preview for Knowledge Worker Selection Strategy

Chapter

7

UC Connector High Availability Deployment

This chapter describes the steps required to deploy primary and backup UC Connector instances in a High Availability (HA) configuration, using a virtual IP-based architecture provided by Windows Network Load Balancer (NLB).

This chapter includes the following sections:

- About HA Through Windows NLB, page 171
- How The Switchover Works, page 172
- Deploying HA Instances of UC Connector, page 173

About HA Through Windows NLB

Windows Network Load Balancer (NLB) is used to provide high-availability for the following connections:

- The interaction web page on the Knowledge Worker desktop—the custom UC Connector tab—and the UC Connector web port.
- Universal Routing Server (URS) and the UC Connector Custom Server port.

The ports used for both of these connections must be switched over as part of the Window NLB mechanism, in order to ensure that both the Preview Notification method and the custom UC Connector tab continue to operate after a switchover from primary to backup instance of UC Connector.

When configured for HA, the HTTP messages from the UC platform are sent to the Windows NLB cluster—using the virtual IP (VIP) address—and the Windows NLB cluster then delivers this traffic to the individual UC Connector instance, according to its unique IP address.

Figure 61 shows an overview of how HA-enabled UC Connector instances can be deployed.
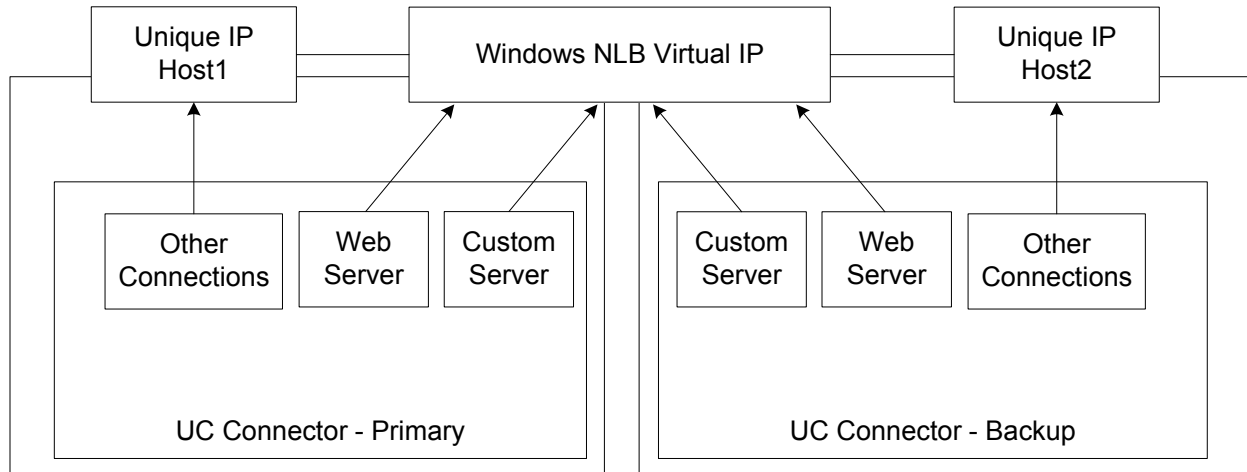
Figure 61: UC Connector in an HA Deployment

While web communication with the UC platform uses a single Virtual IP address to communicate with UC Connector, Genesys Management and Configuration Layer components and T-Library clients use the unique IP address for communication with the UC Connector and the Local Control Agent (LCA) installed at each UC Connector host.

When a failure occurs, Genesys performs a switchover of the failed primary UC Connector instance to the backup instance. The Genesys Management Layer uses a Windows NLB utility (`wlbs.exe` or `nlb.exe`) to enable and disable the web ports used by UC Connector. To start this utility, special control scripts (.bat files) are required. These scripts are triggered when alarm conditions in the UC Connector log events are generated as the UC Connector instances switch modes from primary to backup.

How The Switchover Works

The following steps describe a primary UC Connector failure workflow for a Windows NLB cluster configuration. The figure above represents the end state of the workflow.

1. The primary UC Connector (UC Connector 1) fails.

2. The LCA detects the primary UC Connector application failure and reports it to the SCS.

3. Through the LCA, SCS instructs the backup UC Connector (UC Connector 2) to go into primary mode.

4. When backup UC Connector (UC Connector 2) goes into primary mode, a log event occurs, which indicates that the backup UC Connector has changed to primary mode. The log event triggers an associated alarm condition.

5. The alarm condition triggers associated alarm reaction scripts.

6. The alarm reaction scripts trigger the associated VIP control script application objects.

The control scripts run Windows NLB utilities that disable the Web and Custom Server ports on the primary UC Connector and enable the web port on the backup UC Connector.

Deploying HA Instances of UC Connector

Task Summary: Deploying HA UC Connector Instances Using Windows NLB

The following table provides an overview of the tasks you must complete to deploy a highly available configuration of UC Connector in a Windows NLB cluster environment.

Task Summary: Windows NLB cluster HA Deployment

Step	Related Procedures and Information
Ensure that your system meets the deployment pre-requisites.	"Windows NLB Cluster HA Deployment Prerequisites" on page 174
Configure Windows Network Load Balancing (NLB) Parameters.	Use the Windows NLB Configuration Manager to configure load balancing parameters, as described in Procedure: Configuring Windows NLB cluster parameters, on page 175.

Task Summary: Windows NLB cluster HA Deployment (Continued)

Step	Related Procedures and Information
UC Connector HA Deployment	Complete the following procedures to deploy your UC Connector HA configuration: 1. Procedure: Configuring the primary UC Connector (Windows NLB cluster), on page 176. 2. Procedure: Configuring the Backup UC Connector (Windows NLB cluster), on page 178. 3. Procedure: Configuring HA for Custom Server, on page 179 4. Procedure: Creating Virtual IP Interface control scripts (Windows NLB cluster). Virtual IP (VIP) interface control scripts are used to enable and disable UC Connector ports when UC Connectors change modes. 5. Procedure: Creating application objects for VIP control scripts (Windows NLB cluster), on page 182. Application objects allow the VIP control scripts to be run as applications. 6. Procedure: Creating alarm reaction scripts (Windows NLB cluster), on page 183. Alarm reaction scripts are called when HA-related alarm conditions are activated. 7. Procedure: Creating alarm conditions (Windows NLB cluster), on page 184. When an HA-related log event occurs, such as a log event that records when a UC Connector changes from primary to backup mode, Alarm Conditions are activated. 8. Procedure: Testing alarm conditions (Windows NLB cluster), on page 185.

Windows NLB Cluster HA Deployment Prerequisites

- Two separate physical host computers, one for the primary UC Connector and one for the backup UC Connector.

- Software requirements:
  - UC Connector must be installed and configured on both host computers.
  - A Local Control Agent (LCA) must be installed and configured on both host computers.
  - A Message Server must be installed.

- Networking requirements:
    - A name resolution method such as Domain Name System (DNS), DNS dynamic update protocol, or Windows Internet Name Service (WINS) is required.
    - Both host computers must be members of the same domain.
    - A domain-level account that is a member of the local `Administrators` group is required on each host computer. A dedicated account is recommended.
    - Each host computer must have a unique NetBIOS name.
    - A static IP address is required for each of the network interfaces on both host computers.

Note: Server clustering does not support IP addresses assigned through DHCP.

-
    - A dedicated network switch or separate VLAN for cluster adapters is recommended to reduce switch flooding that may be caused by Windows Network Load Balancing.
    - Access to a domain controller is required. If the cluster service is unable to authenticate the user account used to start the service, the cluster may fail. It is recommended that the domain controller be on the same Local Area Network (LAN) as the cluster to ensure availability.
    - Each node must have at least two network adapters; one for the connection to the public network and the other for the node-to-node private cluster network.
    - A dedicated private network adapter is required for HCL certification.
    - All nodes must have two physically independent LANs or Virtual LANs for public and private communication.
    - If you are using fault-tolerant network cards or network adapter teaming, verify that firmware and drivers are up to date and check with your network adapter manufacturer for Windows NLB cluster compatibility.

Windows NLB Cluster HA Deployment Procedures

Procedure:
Configuring Windows NLB cluster parameters

Purpose: To configure Windows Network Load Balancing (NLB) parameters required for a UC Connector HA deployment.

Start of procedure

1. Open the Microsoft Network Load Balancing Manager tool.

2. Select a cluster host and open the `Cluster Properties` window.

3. On the `Cluster Parameters` tab, select the `Cluster operation mode`. You can choose `Unicast` (default) or `Multicast` mode. For information about Windows NLB Unicast and Multicast modes, refer to your Microsoft Windows Server documentation.

4. Click the `Port Rules` tab.

   a. Specify a `Port range` that includes the port that you will assign as the web port.

   b. In the `Protocols` section, select `Both` (for both UDP and TCP).

   c. In the `Filtering mode` section, select `Multiple host` and set `Affinity` to `None` or `Single`.

   d. Set `Load weight` to `Equal`.

5. Click the `Host Parameters` tab. In the `Initial host state` section, set the `Default state` to `Stopped`.

For more information about Windows NLB cluster parameters, refer to your Microsoft Windows Server documentation.

End of procedure

Next Steps

- Procedure: Configuring the primary UC Connector (Windows NLB cluster)

Procedure:
Configuring the primary UC Connector (Windows NLB cluster)

Purpose: To configure the primary UC Connector `Application` object for high availability.

Start of procedure

1. Stop the UC Connector service on the primary and backup hosts. Genesys UC Connector services can be stopped using the Windows `Services` dialog box.

2. Change the HTTP host to the virtual IP address for the Windows NLB cluster. In the UC Connector `Application` object, go to the `Start Info` tab and modify the `Command Line Arguments` as follows:

   `-ucc_host <Virtual_IP_address_of_NLB_cluster>`

3. Make sure that the HTTP port is one that can be shared on both primary and backup UC Connector hosts. To check the primary HTTP port, go to the `Command Line Arguments` and take note of the port number specified by the following parameter:

`-http_port <shared_port_number>`

Note: Modifying the `Command Line Arguments` is suggested for enabling HA on *existing* UC Connector instances only. If you are deploying new instances of UC Connector for HA, you can specify these `Host` and `Port` parameters in the `User Parameters` page of the Installation wizard. The same rules apply:

- Set `ucc_host` for both primary and backup to the same virtual IP address of the Windows NLB cluster.
- Set `http_port` for both primary and backup to the same value.

---

4. Open Configuration Manager.

5. Select the `Applications` folder and right click the UC Connector `Application` object that you want to configure as the primary UC Connector. Select `Properties`.

6. Click the `Options` tab, `Log` section.

   a. Set the `standard` option to "`network`".

   b. Set the `verbose` option to "`all`".

Note: Setting logging options is required for this UC Connector HA configuration. HA-related log events pass through the Message Server to activate alarm conditions and reaction scripts necessary for managing failover between the primary and backup instances of UC Connector.

---

   c. Click `Apply` to save the configuration changes.

7. Click the `Server Info` tab.

   a. Set the `Redundancy Type` to `Warm Standby`.

   b. For the `Backup Server` option, select the UC Connector `Application` object you want to use as the backup UC Connector. If necessary, browse to locate the backup UC Connector `Application` object.

   c. Click `Apply` to save the configuration changes.

8. Click the `Start Info` tab.

   a. Select `Auto-Restart`.

   b. Click `Apply` to save the configuration changes.

9. Click the `Connections` tab, and then click `Add` to create a connection to the Message Server.

10. Click `Apply` and `OK` to save the configuration changes.

End of procedure

Next Steps

-

Procedure:
Configuring the Backup UC Connector (Windows NLB cluster)

Purpose: To configure the backup UC Connector `Application` object for high availability.

Start of procedure

1. Stop both primary and backup UC Connectors if they are running. You can stop the UC Connector service using the Windows `Services` dialog.

2. Change the HTTP host to the virtual IP address for the Windows NLB cluster. In the UC Connector `Application` object, go to the Start Info tab and modify the Command Line Arguments as follows:

`-ucc_host to ucc_host <Virtual_IP_address_of_NLB_cluster>`

3. Assign the same HTTP port as used in the primary UC Connector host. Go to the Command Line Arguments and enter the shared port number in the following parameter:

`-ucc_port <shared_port_number>`

> **Note:** Modifying the `Command Line Arguments` is suggested for enabling HA on *existing* UC Connector instances only. If you are deploying new UC Connectors for HA, you can specify these `Host` and `Port` parameters in the `User Parameters` page of the Installation wizard. The same rules apply:
> - Set `ucc_host` for both primary and backup to the same virtual IP address of the NLB cluster.
> - Set `http_port` for both primary and backup to the same value.

4. Open Configuration Manager.

5. Select the `Applications` folder and right click on the UC Connector application object that you want to configure as the backup UC Connector.

6. Click the `Start Info` tab.

　　a. Select `Auto-Restart`.

　　b. Click `Apply` to save the configuration changes.

7. Click the `Options` tab, `Log` section.

　　a. Set the `standard` option to "`network`".

UC Connector 8.0

b. Set the `verbose` option to "`all`".

Note: Setting `Log` options is required for this UC Connector HA configuration. HA related log events pass through the Message Server to activate alarm conditions and reaction scripts necessary for managing failover between the primary and backup UC Connectors.

---

c. Click `Apply` to save the configuration changes.

8. Click `Apply` and `OK` to save the configuration changes.

End of procedure

Next Steps

• Procedure: Configuring HA for Custom Server

Procedure:
Configuring HA for Custom Server

Purpose: To complete the configuration steps required to support integration of Custom Server with the Windows NLB virtual IP address.

Custom Server does not need to be deployed in an HA pair. However, the Custom Server application must be configured on a host created for the virtual IP address.

Start of procedure

1. In Configuration Manager, create a new Host object, specifying the IP address that you configured as the virtual IP for the NLB cluster.

2. Create a new Custom Server `Application` object, specifying this virtual IP-based Genesys host. For details about creating this object, see Procedure: Creating the Custom Server Application object, on page 121.

End of procedure

• Procedure: Creating Virtual IP Interface control scripts (Windows NLB cluster)

Procedure:
Creating Virtual IP Interface control scripts (Windows NLB cluster)

Purpose: To create Virtual IP (VIP) control scripts for each of the UC Connectors. Each UC Connector host requires VIP control scripts to enable or disable the Virtual IP (VIP) interface on the host computer when the role of the UC Connector changes. The scripts are used to enabled the VIP interface on the host where the UC Connector is in primary mode and disabled the VIP interface on the host where the UC Connector is in backup mode.

In this procedure, you will create the following four VIP Control Scripts:

- uc_connector_prime_up.bat: Enables the VIP interface on the primary host.

- uc_connector_prime_down.bat: Disables the VIP interface on the primary host

- uc_connector_backup_up.bat: Enables the VIP interface on the backup host

- uc_connector_backup_down.bat: Disables the VIP interface on the backup host

---

Notes: You can use the script names listed above or you can specify your own script names.

If you get security-related error messages for these scripts, you may need to add a password parameter to the wlbs.exe commands. For example, add /PASSW <your_password> to the command:

```
wlbs.exe enable 5060 123.45.68.90:2 /PASSW yourpass123
```

Start of procedure

1. On the primary UC Connector host, create a batch file named `uc_connector_prime_up.bat` and input the following commands:

```
@title Enable Virtual IP Control Script
@echo ************** Primary VIP Enabled ************** >> vip1.log
@echo %time% >> vip1.log
wlbs.exe start uccluster:1 >> vip1.log
wlbs.exe enable <your_web_port> uccluster:1 >> vip1.log
wlbs.exe enable <your_Custom_Server_port> uccluster:1 >> vip1.log
wlbs.exe disable <your_web_port> uccluster:2 >> vip1.log
wlbs.exe enable <your_Custom_Server_port> uccluster:2 >> vip1.log
exit
```

2. On the primary UC Connector host, create a batch file named
   `uc_connector_prime_down.bat` and input the following commands:

```
@title Disable Virtual IP Control Script
@echo *************** Primary VIP Disabled ************ >> vip1.log
@echo %time% >> vip1.log
wlbs.exe disable <your_web_port> uccluster:1 >> vip1.log
wlbs.exe disable <your_Custom_Server_port> uccluster:1 >> vip1.log
ping ·n 2 127.0.0.1
exit
```

3. On the backup UC Connector host, create a batch file named
   `uc_connector_backup_up.bat` and input the following commands:

```
@title Enable Virtual IP Control Script
@echo ************** Backup VIP Enabled *************** >> vip2.log
@echo %time% >> vip2.log
wlbs.exe start uccluster:2 >> vip2.log
wlbs.exe enable <your_web_port> uccluster:2 >> vip2.log
wlbs.exe enable <your_Custum_Server_port> uccluster:2 >> vip2.log
wlbs.exe disable <your_web_port> uccluster:1 >> vip2.log
wlbs.exe disable <your_Customer_Server_port> uccluster:1 >> vip2.log
exit
```

4. On the backup UC Connector host, create a batch file named
   `uc_connector_backup_down.bat` and input the following commands:

```
@title Disable Virtual IP Control Script
@echo ***************** Backup VIP Disabled *********** >> vip2.log
@echo %time% >> vip2.log
wlbs.exe disable <your_web_port> uccluster:2 >> vip2.log
wlbs.exe disable <your_Custom_Server_port> uccluster:2 >> vip2.log
ping ·n 2 127.0.0.1
exit
```

> **Note:** The scripts above include commands to log script execution. The
> logs are created in the directory where the script is located.

End of procedure

Next Steps

- Procedure: Creating application objects for VIP control scripts (Windows
  NLB cluster)

Procedure:
Creating application objects for VIP control scripts (Windows NLB cluster)

Purpose: To create the four "Third Party Server" application objects listed below; one for each of the VIP control scripts created in Procedure: Creating Virtual IP Interface control scripts (Windows NLB cluster).

- uc_connector_Prime_Up
- uc_connector_Prime_Down
- uc_connector_Backup_Up
- uc_connector_Backup_Down

Creating application objects for the VIP control scripts allows the scripts to be run as applications within the Genesys framework.

Prerequisites

- The `Third Party Server` template must already exist in the `Application Templates` folder. If not, right-click this folder, select `Import Application Template`, and import the `Third Party Server` template from your Management Framework CD.

Start of procedure

1. In Configuration Manager, select `Environment` > `Applications`.

2. Right click and select `New` > `Application`.

3. Select the `Third Party Server` template from the `Application Templates` folder and click `OK`.

4. On the `General` tab, enter a name for the application object.

   Note: You can use the application object names listed above or you can specify your own.

5. Select the `Server Info` tab.

   a. Select the host name of the UC Connector where the corresponding VIP control script is located.

   b. If necessary, specify a valid communication port number using the `Edit Port` option.

   Note: This port will not be used. However, because of the way the application works, the port may have to be specified in order to save the application.

6. Select the `Start Info` tab.

    a. Set the `Working Directory` to the location of the control script and enter name of the script in the `Command Line` field.

    b. If you are configuring an application object that disables a VIP interface (`uc_connector_Prime_Down` and `uc_connector_Backup_Down`), set the `Timeout Startup` value to `8`.

7. Repeat the steps in this procedure to create application objects for each of the four VIP control scripts.

End of procedure

Next Steps

- Procedure: Creating alarm reaction scripts (Windows NLB cluster)

Procedure:
Creating alarm reaction scripts (Windows NLB cluster)

Purpose: To create alarm reaction scripts for HA-related alarm conditions. When an HA-related alarm condition occurs, the associated alarm reaction script is run. Alarm reaction scripts are configured to call the application objects you created in Procedure: Creating application objects for VIP control scripts (Windows NLB cluster).

Start of procedure

1. Open Configuration Manager.

2. Select `Resources` > `Scripts`.

3. Right click and select `New` > `Script`.

4. Create four scripts, one for each of the applications objects you created in Procedure: Creating application objects for VIP control scripts (Windows NLB cluster). Select `Alarm Reaction` as the `Script Type`. For example, create the following four Alarm Reaction scripts:
- AR_Script_Prime_Up
- AR_Script_Prime_Down
- AR_Script_Backup_Up
- AR_Script_Backup_Down

5. For each of the Alarm Reaction scripts, use the Alarm Reaction Wizard to configure the `Alarm Reaction Type`.

    a. Select an Alarm Reaction script and right-click to open the `Alarm Reaction Wizard` (select `Wizard` > `Configure`).

    b. In the Alarm Reaction Wizard, click `Next`.

   c. In the Alarm Reaction Type dialog, select `Start a specified application` and click `Next`.

   d. Browse to select the corresponding application object. For example, for the `AR_Script_Prime_Up` Alarm Reaction script, select the `uc_connector_Prime_Up` Third Party Server application object.

   e. Repeat the previous steps to configure each of the Alarm Reaction scripts you created in Step 4.

End of procedure

Next Steps

- Procedure: Creating alarm conditions (Windows NLB cluster)

Procedure: Creating alarm conditions (Windows NLB cluster)

Purpose: Alarm Conditions are required to handle log events that occur when a UC Connector changes its mode from primary to backup or backup to primary. When you create the Alarm Conditions, you configure them to trigger the alarm reaction scripts you created in Procedure: Creating alarm reaction scripts (Windows NLB cluster).

Four alarm conditions are required for your HA configuration, two for the primary UC Connector application and two for the backup. Refer to the procedure that follows to create the alarm conditions required for your configuration.

Table 31: Alarm Conditions for Warm Standby

Name	Log Event ID	Application	Reaction Script
`ALRM_Primary_down_4560`	4560	`<Primary UC Connector>`	`AR_Script_Prime_Down`
`ALRM_Primary_up_4562`	4562	`<Primary UC Connector>`	`AR_Script_Prime_Up`
`ALRM_Backup_down_4560`	4560	`<Backup UC Connector>`	`AR_Script_Backup_Down`
`ALRM_Backup_up_4562`	4562	`<Backup UC Connector>>`	`AR_Script_Backup_Up`

Start of procedure

1. Open Configuration Manager.

2. Navigate to the `Environment > Alarm Conditions` folder.

3. Right click and select `New > Alarm Condition` to open the `New Alarm Condition Properties` dialog.

4. On the `General` tab:
    - Enter a `Name` for the `Alarm Condition`.
    - Optionally, enter a description.
    - For the `Category` value, select `Critical`.
    - Set Cancel Timeout to 3.

5. On the `Detect Event` tab:
    - Set the `Log Event ID`.
    - Set the `Selection Mode` to `Select By Application`.
    - For the Application Name field, click the folder icon to browse for the UC Connector Application object. If you are creating an Alarm Condition for the primary UC Connector, select the primary UC Connector application object. If you are creating an Alarm Condition for the backup UC Connector, select the backup UC Connector application object.

6. Click `OK`.

7. On the `Reaction Scripts` tab, add the alarm reaction script as defined according to the table at the beginning of this procedure.

8. Repeat the steps in this procedure to create each of the four Alarm Conditions for your hot or warm standby configuration.

End of procedure

Next Steps

-

Procedure:
Testing alarm conditions (Windows NLB cluster)

Purpose: To verify that the alarm conditions work as expected.

Start of procedure

1. Open the Solution Control Interface (SCI).

2. Under `Alarm Conditions`, select the "ALRM_Primary_4561" Alarm Condition, right click, and click, `Test`. The "ALRM_Primary_4561" Alarm Condition indicates that the primary UC Connector is in backup mode which triggers the alarm reaction scripts that disable the Virtual IP interface at the primary UC Connector and disable the VIP interface at the backup UC Connector.

3. Use an wlbs `queryport <your_web_port>` or `<your_Custom_Server_port>` command to verify that the Virtual IP interface is active on the backup UC Connector and that the Virtual IP interface is inactive on the primary UC Connector.

End of procedure

![Genesys]

Part

3

Appendixes

The following chapters describe additional reference information for your UC Connector deployment:

UC Connector 8.0

![Genesys logo]

A Configuration Options

This appendix describes the configuration options that are modified during the procedures included in this guide. Options are organized according to component type, and include the following:

UC Connector Application Options

You must configure the sections `UC-Connector`, `License`, `Log`, and either `Microsoft-OCS` or `IBM-Sametime`.

UC-Connector Section

audio-on-preview

Default Value: No default value
Valid Values: URL or file name
Changes Take Effect: Immediately

Specifies the location of the audio file that will be played when the Preview pop-up window is displayed.

audio-on-ring

Default Value: No default value
Valid Values: URL or file name
Changes Take Effect: Immediately

Specifies the location of the audio file that will be played when the Ringing pop-up window is displayed.

Note: The `audio-on-ring` and `audio-on-preview` options are only applicable when UC Connector is used in non-gateway mode (`presence-gateway-mode` is set to `false`).

chat-title

Default Value: `Chat`
Valid Values: Any string
Changes Take Effect: Immediately

Specifies the name used for the Chat window. By default, the window uses the name `Chat`. You can use any other name for this section by changing the value of this option.

chat-consult-title

Default Value: `Consulting Chat`
Valid Values: Any string
Changes Take Effect: Immediately

Specifies the name used for the Chat window that opens for consultation chat interactions. By default, the window uses the name `Consulting Chat`. You can use any other name for this section by changing the value of this option.

presence-location

Default Value: blank
Valid Values: string format, URL or file location
Changes Take Effect: Immediately

Specifies the location of the XML presence configuration file.

Notes: The configuration file is only reloaded when the option changes.

The default definitions will be used if the option is empty.

If the XML is not valid, the current definition in use will remain in effect.

If the resource directory location is specified, then all the files the `presence.xml` file requires must be in the same directory.

login-queue

Default Value: Blank
Valid Values: A valid DN
Changes Take Effect: Immediately

Specifies the value used to populate the agent login request parameter `queue`. SIP Server can use this value for default routing in case of URS failure. The

effects of this parameter depend on the T-Server and other solution components. See the deployment guide for your T-Server/SIP Server for information on whether it supports this feature.

dnd-off-timeout

Default Value: 300000
Valid Values: Any positive integer
Changes Take Effect: Immediately

Specifies the duration, in milliseconds, that UC Connector will set an agent to Do-Not-Disturb (DND) if the Knowledge Worker rejects the Preview Notification.

Note: If UC Connector receives an agent state change event from SIP Server/T-Server (`EventAgentReady`, `EventAgentNotReady`, or `EventAgentLogout`) while the timer is active, then the timer will be cancelled.

enable-logout-menu

Default Value: `false`
Valid Values: `true, false`
Changes Take Effect: Immediately

Specifies whether the log out menu item is displayed in the UC Connector GUI for the particular user, if no UC system configuration annexes are present in the `Person` configuration in CME.

help-callcontrol-url

Default Value: No default value
Valid Values: a valid path to an .html file
Changes Take Effect: Immediately

Specifies the URL path to the `.html` help file that appears when the user clicks the `Help` button on the Call Control or Preview windows. For example, if you copy the sample Call Control Help file from the CD to the UC Connector installation folder, enter a path something like the following:

`help/callcontrol.html`

If you do not configure this option, the Help button does not appear.

If you are hosting your help file externally, use a fully qualified URL as the value for this option. For example,

`http://www.companyhelp.com`

help-interaction-url

Default Value: No default value
Valid Values: a valid path to an .html file
Changes Take Effect: Immediately

Specifies the URL path to the `.html` help file that appears when the user clicks the `Help` button on the Interaction window. For example, if you copy the sample Help file from the CD to the UC Connector installation folder, enter a path something like the following:

`help/interaction.html`

If you do not configure this option, the Help button does not appear.

If you are hosting your help file externally, use a fully qualified URL as the value for this option. For example,

`http://www.companyhelp.com`

help-login-url

Default Value: No default value
Valid Values: a valid path to an .html file
Changes Take Effect: Immediately

Specifies the URL path to the .html help file that appears when the user clicks the Help button on the Login screen. For example, if you copy the sample Call Control Help file from the CD to the UC Connector installation folder, enter a path something like the following:

`help/login.html`

If you do not configure this option, the Help button does not appear.

If you are hosting your help file externally, use a fully qualified URL as the value for this option. For example,

`http://www.companyhelp.com`

itx-window-close-timeout

Default Value: `9000`
Valid Values: Any positive integer
Changes Take Effect: Immediately

Specifies the length of time you want the interaction to remain open after the Knowledge Worker interaction is released or abandoned.

gla-call-match-window

Default Value: `4000`
Valid Values: `2000—1500`
Changes Take Effect: Immediately

Specifies the time window, in milliseconds, in which a T-Lib call is matched against a Lync call reported by GLA. Lync and T-Lib call events do not have a common reference and can only be matched by co-incidence in time.

gla-kpl-time

Default Value: `30`
Valid Values: `4`—integer greater than the value of `gla-kpl-response-time`
Changes Take Effect: Immediately

The interval, in seconds, between GLA keep alive messages being sent to UC Connector.

gla-kpl-response-time

Default Value: `4`
Valid Values: `3`—integer less than the value of `gla-kpl-time`
Changes Take Effect: Immediately

The expected time, in seconds, for UC Connector to respond to the keep alive messages sent by GLA.

locale

Default Value: `en`
Valid Values:A two-character country code
Changes Take Effect: Immediately
Related Feature: "How It Works—Customized Languages" on page 47

Specifies the default language to be used in the UC Connector user interface, if not otherwise specified by the integrated web browser. By default, UC Connector uses English (`en`) for all the labels and buttons in the user interface. To set UC Connector to a different default language, enter one of the supported two-character country codes listed in Table 6, "Supported Languages," on page 48.

If you set this option to `default`, then the language of the local operating system where the UC Connector is running will be used. If you do not configure this option at all, then English is used as the absolute default language.

presence-gateway-mode

Default Value: `false`
Valid Values: `true, false`
Changes Take Effect: Immediately

Specifies whether UC Connector prevents sending invite messages to Lync when there is no web client UI connected for the corresponding user, and a new call or preview call is delivered. If `true,` UC Connector does not send SIP "INVITE" to Lync; otherwise, sending the invite is controlled by the invite-message option.

preview-info-keys

Default Value: `UCC_ConnId, UCC_UserId, UCC_AgentId, UCC_Reason`
Valid Values: A comma-separated list of four identifiers:
<keyname>,<keyname>,<keyname>,<keyname>
Changes Take Effect: Immediately
Related Feature: "How It Works—Reporting Events" on page 42

Specifies the ordered list of key names used for reporting the UC Connector Preview offer parameters.

Note: The `preview-info-keys` values must correspond to the keys provided in the ICON option `AgentUserFields`.

---

preview-state-name

Default Value: `3721,UCC_Preview`
Valid Values: A string in the format <number>,<keyname>
Changes Take Effect: Immediately
Related Feature: "How It Works—Reporting Events" on

Specifies the numeric identifier and the key name of the custom state associated with the UC Connector Preview offer.

Note: The `preview-state-name` values must correspond to the value of the ICON option `AgentRecordUserTypes`.

---

preview-shortkey-accept

Default Value: Blank
Valid Values: A string in alpha-numeric or ASCII decimal number format
Changes Take Effect: After Restart

The hotkey used to accept a call when the Preview window is displayed. If left blank, the shortcut key is disabled.For details on the ASCII and alpha-numeric formats, see Configuring Hotkeys for Interaction Preview.

Note: The Preview window must be in focus for the hotkey to function.

---

preview-shortkey-reject

Default Value: Blank
Valid Values: A string in alpha-numeric or ASCII decimal number format
Changes Take Effect: After Restart

The hotkey used to reject a call when the Preview window is displayed. If left blank, the shortcut key is disabled. For details on the ASCII and alpha-numeric formats, see Configuring Hotkeys for Interaction Preview.

Note: The Preview window must be in focus for the hotkey to function.

---

user-unregister-timeout

Default Value: `60000` (1 minute)
Valid Values: Any positive integer
Changes Take Effect: Immediately

Specifies the length of time, in milliseconds, that UC Connector will wait after Knowledge Worker has closed all browser sessions before it unregisters the Knowledge Worker DN with T-Server/SIP Server.

preview-itx-arrival-timeout

Default Value: `9000`
Valid Values: Any positive integer
Changes Take Effect: Immediately

Specifies the length of time, in milliseconds, that the UC Connector will wait for the interaction to arrive after the Preview Notification was accepted by the Knowledge Worker.

preview-expiration-timeout

Default Value: `15000`
Valid Values: Any positive integer
Changes Take Effect: Immediately

Specifies the length of time, in milliseconds, that the UC Connector will wait for the Knowledge Worker response (accept or reject) to the Preview Notification.

A countdown timer in the preview window shows how much time is remaining. If the timer runs out, the call is returned to the URS routing strategy.

redirect-setup-enabled

Default Value: `false`
Valid Values: `true`, `false`
Changes Take Effect: Immediately

This global option enables all users to set and enable a redirect number. This value can be overwritten by the Person-level redirect-setup-enabled option.

enable-preview-reporting

Default Value: `false`
Valid Values: `true`, `false`
Changes Take Effect: Immediately
Related Feature: "How It Works—Reporting Events" on

When set to `true`, UC Connector creates records in ICON for the interaction Preview events accept, reject, timeout, or withdrawn due to another user in the group accepting the interaction.

Note: Enabling this option will cause extra network traffic from UC Connector to T-Server/SIP Server, so you should check your network limits.

---

user-auto-registration

Default Value: `false`
Valid Values: `true`, `false`
Changes Take Effect: At application restart

Enables automated DN registration of all Knowledge Workers configured in the Genesys environment, at the moment the UC Connector application starts. If set to `true`, UC Connector will begin monitoring all user DNs and their presence on the UC platform automatically, without requiring individual Knowledge Workers to log in to Genesys. For example, in integrations with Microsoft Lync Server, no custom tab is available for Knowledge Worker log in; in this case, this option must be set to `true`.

Note: The auto-registration will fail if the Knowledge Worker is currently on a call. If the registration does fail, UC Connector will try again after the DN is no longer busy or (in the case of some other login failure) when the Knowledge Worker's UC presence changes.

---

userdata-preview-format<n>

Default Value: `title: <UserData_DisplayName>, value: [UserData_Key]`
Valid Values: `title: <UserData_DisplayName>, value: [UserData_Key]`
Changes Take Effect: Immediately

Specifies which UserData key-value pair will be displayed in the Preview window, as well as how it will be displayed. For example, to display the UserData kvp `custname` with the title `Customer Name` in the Interaction window, configure the option as follows:

`title: Customer Name, value: [custname]`

You can name the `<n>` variable in this option in numerical order (1, 2, 3) as you add more kvps, or you can leave it empty.

userdata-call-format<n>

Default Value: `title: <UserData_DisplayName>, value: [UserData_Key]`
Valid Values: `title: <UserData_DisplayName>, value: [UserData_Key]`
Changes Take Effect: Immediately

Specifies which `UserData` key-value pair will be displayed in the Interaction window, as well as how it will be displayed. For example, to display the UserData kvp `cust-account` with the title Account in the Interaction window, configure the option as follows:

`title: Account, value: [cust-account]`

You can name the `<n>` variable in this option in numerical order (1, 2, 3) as you add more kvps, or you can leave it empty.

userdata-contact-format

Default Value: `title: Contact, value: [USER-ID]`

Valid Values: `title: Contact, value: [USER-ID]`
Changes Take Effect: Immediately

Specifies whether the Customer field will be displayed in the Interaction window

If this option is enabled (with valid values), UC Connector checks the `UserData` attribute for the key name `customer`. If found, UC Connector includes the value for this key in the `Contact` field of the Preview Notification and Interaction windows.

For example, if the key-value pair `customer=Bank of Nova Scotia` is found in the `UserData`, then UC Connector will display `Customer Bank of Nova Scotia` in the Preview Notification and Interaction windows.

userdata-note-key

Default Value: `KW_ITX_NOTES`
Valid Values: Any text
Changes Take Effect: Immediately

Specifies the key-name to be used for the notes taken by by the Knowledge Worker during the interaction, and returned to the contact center as user data.

By default, the key name for Know edge Worker notes in the user data is `KW_ITX_NOTES`.

userdata-onringing

Default Value: `false`
Valid Values: `true, false`
Changes Take Effect: Immediately

Specifies whether UC Connector will display UserData when the call to the Knowledge Worker is in the ringing state.

userdata-note-onpreview

Default Value: `false`
Valid Values: `true, false`
Changes Take Effect: Immediately

Specifies whether UC Connector will display the UserData `Notes` (configured in `userdata-note-key`) when the Preview Interaction is initially presented to the Knowledge Worker.

Limitation: For agent notes that will appear in the Preview window, the text cannot contain a single (') mark. URS does not process this quotation mark as regular text, but as a text delimiter. In other words, any text after this quotation park will not be included in the Preview window.

userdata-note-onringing

Default Value: `false`
Valid Values: `true, false`

Changes Take Effect: Immediately

Specifies whether UC Connector will display the UserData `Notes` (configured in `userdata-note-key`) when the call to the Knowledge Worker is in the ringing state.

userdata-contact-onpreview

Default Value: `false`
Valid Values: `true`, `false`
Changes Take Effect: Immediately

Specifies whether UC Connector will display the contact information for the caller in the Preview Notification sent to the Knowledge Worker.

userdata-contact-onringing

Default Value: `false`
Valid Values: `true`, `false`
Changes Take Effect: Immediately

Specifies whether UC Connector will display the contact information for the caller when the call to the Knowledge Worker is in the ringing state.

userdata-title

Default Value: `Case data`
Valid Values: Any text
Changes Take Effect: Immediately

Specifies the title for the user data that is displayed in the Knowledge Worker Call Control or Preview window. By default, this part of the client UI uses the title `Case data`. You can use any other name for this section by changing the value of this option.

userdata-note-title

Default Value: `Notes`
Valid Values: Any text
Changes Take Effect: Immediately

Specifies the title for the section of the UC client interface where the Knowledge Worker can input notes about the interaction. By default, this area is named `Notes`. You can use any other name for this section by changing the value of this option.

popup-udata-key

Default Value: Blank
Valid Values: A string representing a User Data Key
Changes Take Effect: Immediately

If the value is blank, UC Connector will display the third-party call control window to the user for all arriving calls. If the option value contains a User

Data Key, UC Connector will only display the call control window for calls with the specified User Data Key present in the User Data of the call.

Note: Calls delivered using the Preview window will not be suppressed, regardless of user data filer. Clicking on transaction line in the main UC Connector window will bring up the third-party call control window regardless of user data filter.

Microsoft-OCS Section

This section must be created using the correct syntax for the supported UC platform you are integrating with: `Microsoft-OCS`.

presence-acw-note

Default Value: `Working after call`
Valid Values: Any string
Changes Take Effect: Immediately

Specifies the presence note that UC Connector uses when an agent enters the After Call Work state. For Lync voice terminals, UC Connector uses this value for the presence update to Lync when the agent enters the After Call Work phase.

Note: The ampersand (&) and less than (<) symbols should not be used in the string value. Microsoft Lync 2010 does not accept presence updates if the `presence-acw-note` option contains an ampersand (&) or less than (<) symbol. It rejects the presence update from UC Connector and the Lync user will not be set into the state specified by the presence-acw-status option.

---

presence-acw-status

Default Value: `6500`
Valid Values: Any valid `MS-PRES` protocol interoperability code (see Table 14 on page 90)
Changes Take Effect: Immediately

Specifies the presence state that UC Connector uses when an agent enters the After Call Work state. For Lync voice terminals, UC Connector uses this value for the presence update to Lync when an agent enters the After Call Work phase.

presence-lg-note

Default Value: `Statutory pause`
Valid Values: Any string
Changes Take Effect: Immediately

Specifies the presence note that UC Connector uses when an agent exits the After Call Work state and enters the Legal Guard state. For Lync voice terminals, UC Connector uses this value for the presence update to Lync when an agent exits the After Call Work state and enters the Legal Guard state.

Note: The ampersand (&) and less than (<) symbols should not be used in the string value. Microsoft Lync 2010 does not accept presence updates if the value of the `presence-lg-note` option contains an ampersand (&) or less than (<) symbol. It rejects the presence update from UC Connector and the Lync user will not be set into the state specified by the presence-lg-status option.

presence-lg-status

Default Value: `6500`
Valid Values: Any valid `MS-PRES` protocol interoperability code (see Table 14 on page 90)
Changes Take Effect: Immediately

Specifies the presence state that UC Connector uses when an agent exits the After Call Work state and enters the Legal Guard state. For Lync voice terminals, UC Connector uses this value for the presence update to Lync when an agent exits the After Call Work state and enters the Legal Guard state.

agent-onnote-tstatus<x>

Default Value: No default value
Valid Values: `<tserver status>`, `<note>`
Changes Take Effect: Immediately

Specifies the text for the triggering note, as well as the corresponding Genesys status. If enabled, when the Knowledge Worker enters a note that matches the text configured in this option, UC Connector will change the status for this Knowledge Worker to the specified Genesys agent state. To configure this option, enter the value `<tserver status>`, `<note>` in the following pattern:

<tserver status>	Enter the Genesys status that the note will trigger. It can be any of the following Genesys states: ready, not ready, logout.
<note>	Enter the text for the triggering note.

For example, if you configure `agent-onnote-tstatus1` to the value `NotReady, I am busy`, when the Knowledge Worker enters "`I am busy`" in the MOC client, the Genesys agent state is changed to `NotReady`.

Note: Use the variable <x> in the name of this option when creating multiple triggering notes (set X to 1, 2, 3 and so on as you add more notes). If you have only one triggering note, you can leave out this variable.

agent-status-ready

Default Value: `0-4499`

Valid Values: Any valid `MS-PRES` protocol interoperability code (see Table 14 on page 90)

Changes Take Effect: Immediately

Specifies the presence status on the Microsoft OCS-side that will be mapped to the Genesys status of `Ready`.

agent-status-notready

Default Value: `4500-17999`

Valid Values: Any valid `MS-PRES` protocol interoperability code (see Table 14 on page 90)

Changes Take Effect: Immediately

Specifies the presence status on the Microsoft OCS-side that will be mapped to the Genesys status of `NotReady`. You can map several status codes to the `NotReady` state, by adding a comma-separate list of status codes. For example, `6500, 9500, 12500, 15500`.

agent-status-logout

Default Value: `18000-`

Valid Values: Any valid `MS-PRES` protocol interoperability code (see Table 14 on page 90)

Changes Take Effect: Immediately

Specifies the presence statuses on the Microsoft OCS-side that will be mapped to the Genesys status of `Logout`. Typically, you will map several OCS presence statuses to match the Genesys `Logout` status.

contact

Default Value: No default value

Valid Values: Valid SIP URI

Changes Take Effect: Immediately

Specifies the Knowledge Worker name for the UC Connector as configured in Microsoft OCS. This value matches the sign-in name for the Knowledge Worker in Microsoft OCS. For example,

`sip:ocs-ucc@your-ocs-address.com`

enable-push-oncall-status

Default Value: `false`

Valid Values: `true, false`

Changes Take Effect: Immediately

Enables the overall Push On Call Status feature. If enabled, when the Knowledge Worker receives a call from the Genesys environment, the MOC status is set on the OCS side, according to the value of the `oncall-status` option.

oncall-status

Default Value: `9500` (recommended `6500`)
Valid Values: Any valid OCS status. See Table 14 on page 90.
Changes Take Effect: Immediately

Specifies the status to be updated in the Microsoft Office Communicator/Lync client when a Knowledge Worker receives a call from the Genesys environment.

Note: Genesys recommends that you change the value of this option from the default `9500` to the value `6500`. The value `9500` corresponds to the OCS status `DoNotDisturb` on the Microsoft side, in which case interaction sessions cannot be established. Use the value `6500` (Busy) instead.

---

invite-message

Default Value: `"Please use the window on the right to access data about current interactions"`
Valid Values: Any text string
Changes Take Effect: Immediately

Specifies the text that will be presented to the Knowledge Worker in the Extensibility Window when an interaction arrives. This option is used for integrations with Microsoft Lync only. The window will only be enabled if this option is present and has a value configured. For an example of a customized welcome message, see Figure 13 on page 46.

registrar-uri

Default Value: No default value
Valid Values: Valid SIP URI
Changes Take Effect: Immediately

Specifies the URI that UC Connector uses to connect with Microsoft OCS. For example,

`sip:pool01.your-ocs-address.com`

IBM-Sametime Section

This section must be created using the correct syntax for the supported UC platform you are integrating with: `IBM-Sametime`.

server-port

Default Value: `1516`
Valid Values: Any valid port
Changes Take Effect: Immediately

Specifies the port number for the IBM Sametime server.

server-fqdn

Default Value: `<required>`
Valid Values: Any valid FQDN
Changes Take Effect: Immediately

Specifies the fully qualified domain name (FQDN) for the IBM Sametime server host computer.

channel-type

Default Value: `62`
Valid Values: `62`
Changes Take Effect: Immediately

Specifies the IBM Sametime delivery channel. The default setting of `62` specifies a voice channel.

agent-status-ready

Default Value: `32,544`
Valid Values: See Table 15 on page 90
Changes Take Effect: Immediately

Specifies the presence status on the IBM Sametime-side that will be mapped to the Genesys status of `Ready`.

agent-status-notready

Default Value: `8,64,96,128,512,16384`
Valid Values: See Table 15 on page 90
Changes Take Effect: Immediately

Specifies the presence status on the IBM Sametime-side that will be mapped to the Genesys status of `NotReady`. You can map several status codes to the `NotReady` state, by adding a comma-separate list of status codes.

agent-status-logout

Default Value: `-32768,0`
Valid Values: See Table 15 on page 90
Changes Take Effect: Immediately

Specifies the presence statuses on the IBM Sametime-side that will be mapped to the Genesys status of `Logout`. Typically, you will map several Sametime presence statuses to match the Genesys `Logout` status.

enable-push-oncall-status

Default Value: `false`
Valid Values: `true, false`
Changes Take Effect: Immediately

Enables the overall Push On Call Status feature. If enabled, when the Knowledge Worker receives a call from the Genesys environment, the IBM

status is set on the Sametime side, according to the value of the `oncall-status` option.

oncall-status

Default Value: `128 (ST_USER_STATUS_DND)`
Valid Values: Any valid IBM status. See Table 15 on .
Changes Take Effect: Immediately

Specifies the status to be updated in the Sametime Connect client when a Knowledge Worker receives a call from the Genesys environment.

oncall-status-message

Default Value: "`I am on a call`"
Valid Values: `<text>`
Changes Take Effect: Immediately

Specifies the text of the message to be shown in the Sametime Connect client, accompanying the change in status when the Knowledge Worker receives a Genesys call.

call-released-status

Default Value: "`reset`"
Valid Values: Any valid IBM status. See Table 15 on .
Changes Take Effect: Immediately

Specifies the Sametime Connect status to be updated when the Knowledge Worker releases a call from the Genesys environment. The default setting "`reset`" means that the Knowledge Worker will revert to their last status in Sametime Connect before the Genesys call started.

License Section

license-file

Default Value: `<required>`
Valid Values: Any valid port address in the format,
       `<your_license_server_port>@<your_license_server_host>`
       or the full path to the license file
Changes Take Effect: After restart

Specifies the location of the license file.

num-of-seat-licenses

Default Value: `max` (all available licenses)
Valid Values: `0` or string `max`, or any integer from `0-9999`.
Changes Take Effect: After restart

Specifies how many seat licenses UC Connector checks out. UC Connector treats a value of 0 (zero) the same as it treats max—that is, it checks out all available licenses.

The sum of all `num-of-seat-licenses` values for all concurrently deployed UC Connector instances must not exceed the number of seat-related licenses in the corresponding license file. The primary and backup UC Connector share the same licenses, and therefore they need to be counted only once. UC Connector checks out the number of licenses indicated by the value for this option, regardless of the number actually in use.

Log Section

There is one UC Connector-specific option available for the `Log` section. For the common `Log` options, see the *Framework 8.0 Deployment Guide*.

internal

Default Value: `error`
Valid Values: `Error, Warn, Info, Debug`
Changes Take Effect: Immediately

Specifies the log priority for internal UC Connector components.

Switch/DN Level Options

Contact Point DNs

For `Routing Point` DNs used to create contact points that the Knowledge Worker uses to direct call transfers or conferences to the contact center, the options in this section are available. All options must be configured in the `UC-Connector` section of the `Annex` tab.

attribute<n>

Default Value: No default value
Valid Values: `format: <display text>: %s, statistic:<object>, <ObjectType>, <TenantName>, <StatType>, <TimeProfile>, <StatServerName>`
Changes Take Effect: Immediately

Specifies the statistic to be displayed when the Knowledge Worker hovers their cursor over the contact point in their Interaction window. You can name the `<n>` variable in this option in numerical order (1, 2, 3) as you add more statistics, or you can leave it empty.

If this option contains both the `format` and `statistic` fields, UC Connector subscribes to retrieve the specified statistic from Stat Server. All statistic types supported by Stat Server are available.

The valid format for this option value is as follows:

- format—Set this option to the text you want displayed in the Interaction window. Include the `%s,` for the statistic to be displayed.

- statistic—Set this option to the parameters required by Stat Server to properly fetch the relevant data: `Object, ObjectType, TenantName, StatType, TimeProfile, StatServerName`

> **Note:** The parameter `StatServerName` is only required if UC Connector connects to more than one Stat Server.

An example of a valid configured value is as follows:

```
format: Calls waiting: %s, statistic:1234@YourSwitch, RoutePoint,
YourTenantName, CurrNumberWaitingCalls, Default, StatServer1
```

> **Note:** For details about statistic types, see the *Framework 7.6 Stat Server Deployment Guide.*

display-name

Default Value: No default value
Valid Values: Any valid string
Changes Take Effect: After application restart

Specifies the name to be displayed for the configured contact point (typically a `Routing Point` DN or ACD Queue) in the Knowledge Worker's Interaction window.

enabled

Default Value: true
Valid Values: true, false
Changes Take Effect: Immediately

Specifies whether this `Routing Point` DN is enabled for use with UC Connector. Set this option to `true` to enable the contact point.

presence-gateway-mode

Default Value: `false`
Valid Values: `true, false`
Changes Take Effect: Immediately

Specifies whether UC Connector prevents sending invite messages to Lync when there is no web client UI connected for the corresponding user, and a new call or preview call is delivered. If `true,` UC Connector does not send SIP "INVITE" to Lync; otherwise, sending the invite is controlled by the invite-message option.

Knowledge Worker Person Object

For Knowledge Worker `Person` objects, the following options are available.

UC-Connector Section

enabled

Default Value: `true`
Valid Values: `true`, `false`
Changes Take Effect: Immediately

Specifies whether this `Person` object is enabled for use with UC Connector. Set this option to `true` to enable this Knowledge Worker `Person` object.

redirect-setup-enabled

Default Value: `false`
Valid Values: `true`, `false`
Changes Take Effect: Immediately

Enables a user to set and enable a redirect number. This values takes priority over the global redirect-setup-enabled option.

redirect-enabled

Default Value: `false`
Valid Values: `true`, `false`
Changes Take Effect: Immediately

Specifies whether the configured redirect number can be used to accept previews. If true, UC Connector uses the configured redirect number to accept previews.

Note: This option can only be configured for a Person.

redirect-number

Default Value: blank
Valid Values: A string representing a valid phone number
Changes Take Effect: Immediately

The redirect number available to the Knowledge Worker.

Note: This option can only be configured for a Person.

Microsoft-OCS Section

agent-status-ready

Default Value: No default value

Valid Values: Any valid `MS-PRES` protocol interoperability code (see Table 14 on page 90)
Changes Take Effect: Immediately

Specifies the presence status for this Knowledge Worker on the Microsoft OCS-side, which will then be mapped to the Genesys status of `Ready`. Typically, you will map several OCS presence status to match the Genesys `NotReady` status. For example, `3500, 3000-4499`.

agent-status-notready

Default Value: No default value
Valid Values: Any valid `MS-PRES` protocol interoperability code (see Table 14 on page 90)
Changes Take Effect: Immediately

Specifies the presence status for this Knowledge Worker on the Microsoft OCS-side, which will then be mapped to the Genesys status of `NotReady`. Typically, you will map several OCS presence status to match the Genesys `NotReady` status. For example, `6500,9500, 12500, 15500,4500-17999`.

agent-status-logout

Default Value: No default value
Valid Values: Any valid `MS-PRES` protocol interoperability code (see Table 14 on page 90)
Changes Take Effect: Immediately

Specifies the presence statuses for this Knowledge Worker on the Microsoft OCS-side, which will then be mapped to the Genesys status of `Logout`. Typically, you will map several OCS presence statuses to match the Genesys `Logout` status. For example, `18500,0-2999,18000-`.

contact

Default Value: `<required>`
Valid Values: Valid SIP URI
Changes Take Effect: Immediately

Specifies the principal for the Knowledge Worker as configured in Microsoft OCS. This value matches the sign-in name for the Knowledge Worker in Microsoft OCS. For example,

`sip:ucc_user1@your-ocs-address.com`

enable-push-oncall-status

Default Value: `false`
Valid Values: `true, false`
Changes Take Effect: Immediately

Enables the overall Push On Call Status feature. If enabled, when the Knowledge Worker receives a call from the Genesys environment, the MOC status is set on the OCS side, according to the value of the `oncall-status`

oncall-status

Default Value: `9500` (recommended `6500`)
Valid Values: Any valid OCS status. See Table 14 on page 90.
Changes Take Effect: Immediately

Specifies the status to be updated in the Microsoft Office Communicator/Lync client when a Knowledge Worker receives a call from the Genesys environment.

Note: Genesys recommends that you change the value of this option from the default `9500` to the value `6500`. The value `9500` corresponds to the OCS status `DoNotDisturb` on the Microsoft side, in which case interaction sessions cannot be established. Use the value `6500` (Busy) instead.

IBM-Sametime Section

agent-status-ready

Default Value: `32,544`
Valid Values: See Table 15 on page 90
Changes Take Effect: Immediately

Specifies the presence status on the IBM Sametime-side that will be mapped to the Genesys status of `Ready`.

agent-status-notready

Default Value: `8,64,96,128,512,16384`
Valid Values: See Table 15 on page 90
Changes Take Effect: Immediately

Specifies the presence status on the IBM Sametime-side that will be mapped to the Genesys status of `NotReady`. You can map several status codes to the `NotReady` state, by adding a comma-separate list of status codes.

agent-status-logout

Default Value: `-32768,0`
Valid Values: See Table 15 on page 90
Changes Take Effect: Immediately

Specifies the presence statuses on the IBM Sametime-side that will be mapped to the Genesys status of `Logout`. Typically, you will map several Sametime presence statuses to match the Genesys `Logout` status.

contact

Default Value: No default value
Valid Value: `CN=<common_name>/0=<Organization>`
Changes Take Effect: Immediately

Specifies the IBM Sametime user configured as the contact between UC Connector and IBM Sametime. The user name must be entered in the same LDAP format as configured in the IBM Sametime environment, where,

CN=common name

O=organization

For example, if the common name for the IBM Sametime user is `Jack Smith` and the organization is `City Bank`, then the option should be entered in the format:

`CN=Jack smith/O=City Bank`

enable-push-oncall-status

Default Value: `false`
Valid Values: `true, false`
Changes Take Effect: Immediately

Enables the overall Push On Call Status feature. If enabled, when the Knowledge Worker receives a call from the Genesys environment, the IBM status is set on the Sametime side, according to the value of the `oncall-status` option.

oncall-status

Default Value: `128 (ST_USER_STATUS_DND)`
Valid Values: Any valid IBM status. See Table 15 on .
Changes Take Effect: Immediately

Specifies the status to be updated in the Sametime Connect client when a Knowledge Worker receives a call from the Genesys environment.

oncall-status-message

Default Value: "`I am on a call`"
Valid Values: `<text>`
Changes Take Effect: Immediately

Specifies the text of the message to be shown in the Sametime Connect client, accompanying the change in status when the Knowledge Worker receives a Genesys call.

call-released-status

Default Value: "`reset`"
Valid Values: Any valid IBM status. See Table 15 on .
Changes Take Effect: Immediately

Specifies the Sametime Connect status to be updated when the Knowledge Worker releases a call from the Genesys environment. The default setting "`reset`" means that the Knowledge Worker will revert to their last status in Sametime Connect before the Genesys call started.

![Genesys logo]

B Log Events for UCC HA

This appendix describes the log events that are generated by UC Connector in a warm standby high-availability (HA) configuration.

Log events for Warm Standby

```
00-04560
Level: Standard
Text:  Warm Standby (backup) mode activated
Attributes: None
Description: Management Layer command: reports that the backup
server is working in Warm Standby mode.


00-04562
Level: Standard
Text:  Warm Standby (Primary) mode activated
Attributes:  None
Description:  Management Layer command: reports that the primary
server is working in Warm Standby mode.
```

When you configure your UC Connector HA deployment, you will create Alarm Conditions (two on the primary UC Connector and two on the backup UC Connector) for warm standby HA log events. You will configure the Alarm Conditions to trigger Alarm Reaction scripts, which will in turn run application objects for scripts that execute the SIP Server switchover.

Genesys

Appendix

C T-Server Compatibility with UC Connector

This appendix lists T-Server support for the emulated agent functionality in UC Connector.

Table 32: UC Connector Compatibility with T-Server

T-Server	Version	UC-Connect Compatibility
Alcatel A4200	7.6	No
Alcatel A4400/OXE	8.0+	Yes
Aspect ACD	8.0	No
Avaya Communication Manager	8.0+	Yes
Avaya INDeX	8.0+	Yes
Avaya TSAPI	8.0+	Yes
Cisco UCCE	8.0+	Yes
Cisco UCM	8.0+	Yes
Digitro AXS/20	7.5	No
EADS Intercom M6880	8.0	No
Ericsson MD110	8.0+	Yes
Huawei NGN	7.6	No
Mitel SX-2000 / MN-3300	7.2	No

Table 32: UC Connector Compatibility with T-Server (Continued)

T-Server	Version	UC-Connect Compatibility
NEC NEAX/APEX	8.0	No
Nortel CS1000	8.0+	Yes
Nortel CS2000	8.0+	Yes
Rockwell Spectrum	8.0	No
Siemens Hicom 300 / Highpath 4000 CSTA I	7.6+	Yes
Siemens HiPath 4000 CSTA III	8.0+	Yes
Siemens HiPath DX	8.0+	Yes
Tadiran Coral	8.0+	Yes
Genesys SIP Server	8.0+	Yes

![Genesys logo]

D Presence Definition Document Examples

This appendix provides an example of the presence definition document that allows you to customize the states available to Knowledge Workers in the UC Connector web client drop-down menu.

This appendix contains the following sections:

Example of Custom State Definitions Document

The example below shows a possible customization of the `presence.xml` file.

> **Note:** In this example, localization has been inserted into the file to allow for simplicity, but other options are available as long as the file follows the rules of the Presence Definition XML Schema.

```
<?xml version="1.0" encoding="utf-8"?>
<presence xmlns="http://genesyslab.com/schemas/ucc/presence"
   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xsi:schemaLocation="http://genesyslab.com/schemas/ucc/presence presence.xsd">

   <state icon="ready.png" id="ready">
   <display>
      <menuText>Set Ready</menuText>
      <statusText>Ready for next call</statusText>
      </display>
      <genesysAgent ready="true"/>
   </state>

   <state icon="acw.png" id="acw">
```

```
<display>
   <statusText>Working after call</statusText>
   </display>
<display language="fr">
   <statusText>Travail apres appel</statusText>
   </display>
<display language="de">
   <statusText>Arbeiten nach Anruf</statusText>
   </display>
   <genesysAgent ready="false" mode="acw" />
</state>

<state id="away" icon="away.png">
<display>
   <menuText>Set away</menuText>
   <statusText>Away from the desk</statusText>
   </display>
   <genesysAgent ready="false" mode="away" />
</state>

<state id="lunch" icon="away.png">
<display>
   <menuText>Lunch</menuText>
   <statusText>Away for a lunch</statusText>
   </display>
   <genesysAgent ready="false" mode="away">
      <reason ext="true">500</reason>
   </genesysAgent>
</state>

<state id="auto-away" icon="away.png">
<display>
   <statusText>Automatic away</statusText>
   </display>
   <genesysAgent ready="false" mode="away">
      <reason>timeout</reason>
   </genesysAgent>
</state>

<state id="busy-preview" icon="busy.png">
<display>
   <menuText>On a call</menuText>
   <statusText>On a remote call</statusText>
   </display>
   <genesysAgent ready="false" mode="acw">
      <reason>redirect</reason>
   </genesysAgent>
</state>

<state id="reject" icon="busy.png">
<display><statusText>Busy after reject</statusText></display>
```

```
        <genesysAgent ready="false" mode="auxwork">
            <reason>reject</reason>
        </genesysAgent>
    </state>

    <auto>
        <event name="preview-timeout" onevent="auto-away" />
        <event name="preview-reject" onevent="reject" postevent="ready" />
        <event name="preview-redirect" onevent="busy-preview"/>
    </auto>
</presence>
```

Presence Definition XML Schema

```
<?xml version="1.0" encoding="UTF-8"?>
<schema targetNamespace="http://genesyslab.com/schemas/ucc/presence"
        xmlns="http://www.w3.org/2001/XMLSchema"
        xmlns:uccp="http://genesyslab.com/schemas/ucc/presence"
        elementFormDefault="qualified">
  <annotation>
    <documentation>Presence states specification in Smart Link solution
    </documentation>
  </annotation>
  <element name="presence">
    <annotation>
      <documentation>The root element of Smart Link presence containing all
      state definitions
      </documentation>
    </annotation>
    <complexType>
      <sequence>
  <element name="state" type="uccp:StateType" minOccurs="1" maxOccurs="unbounded">
    <annotation>
      <documentation>Definition of a distinct state</documentation>
    </annotation>
  </element>
  <element name="auto" type="uccp:AutoStateEvents" minOccurs="0" maxOccurs="1">
    <annotation>
      <documentation>Automatic state transitions executed by UCC
      </documentation>
    </annotation>
    <key name="event-name">
      <selector xpath="event" />
      <field xpath="@name" />
    </key>
  </element>
  <element name="composite" minOccurs="0" maxOccurs="unbounded">
    <annotation>
      <documentation>Definitions of composite multi-channel states.</documentation>
    </annotation>
```

```
    〈complexType〉
      〈sequence〉
        〈any〉
      〈annotation〉
        〈documentation〉
          The schema for complex combined presence states across multiple channels is to
be defined.
          Not yet implemented.
        〈/documentation〉
      〈/annotation〉
        〈/any〉
      〈/sequence〉
    〈/complexType〉
  〈/element〉
      〈/sequence〉
    〈/complexType〉
    〈key name="state-id"〉
      〈selector xpath="state" /〉
      〈field xpath="@id" /〉
    〈/key〉
    〈keyref name="auto-ref" refer="uccp:state-id"〉
      〈selector xpath="auto/*" /〉
      〈field xpath="@onevent" /〉
    〈/keyref〉
    〈keyref name="auto-ref-post" refer="uccp:state-id"〉
      〈selector xpath="auto/*" /〉
      〈field xpath="@postevent" /〉
    〈/keyref〉
  〈/element〉

  〈complexType name="StateText"〉
    〈annotation〉
      〈documentation〉
      Localizable strings for state-related GUI.
      〈/documentation〉
    〈/annotation〉
    〈all〉
      〈element name="menuText" type="string" minOccurs="0"〉
        〈annotation〉
          〈documentation〉
            The text that is displayed as GUI menu string.
            Absent element means GUI menu is not available (such state can only be
            entered automatically and cannot be requested from the menu)
          〈/documentation〉
        〈/annotation〉
      〈/element〉
      〈element name="statusText" type="string" minOccurs="0"〉
        〈annotation〉
          〈documentation〉Text string displayed as current status indicator
          when in this state.
          Absent element means that the status is not reflected in the status bar.
```

```
          </documentation>
        </annotation>
      </element>
    </all>
    <attribute name="language" type="language" use="optional" />
  </complexType>

  <complexType name="StateType">
    <annotation>
      <documentation>Full definition of a state consists of GUI string in one or more
languages,
      matching Genesys agent states and UC states.
      </documentation>
    </annotation>
    <sequence>
      <element name="display" type="uccp:StateText" minOccurs="1"
maxOccurs="unbounded">
        <annotation>
          <documentation>GUI appearance of the state</documentation>
        </annotation>
      </element>
      <element name="genesysAgent" type="uccp:GenesysAgentState" minOccurs="1"
maxOccurs="unbounded">
        <annotation>
          <documentation>Genesys agent state matching this state definition.
          UCC user will transition to this state if Genesys agent status
          event is received matching this element.
          </documentation>
        </annotation>
      </element>
    </sequence>
    <attribute name="id" type="ID">
      <annotation>
        <documentation>Unique ID of the state. It can be used to reference
        this state from the "auto" section or "partial" section.
        </documentation>
      </annotation>
    </attribute>
    <attribute name="icon" type="anyURI">
      <annotation>
        <documentation>
          Graphic representation of this state in GUI.
          Use HTTP URL or one of the pre-defined icon identifiers in the
            "tag:ucc.genesyslab.com,2013:icons/status/" namespace
        </documentation>
      </annotation>
    </attribute>
  </complexType>

  <complexType name="AutoStateEvents">
    <sequence>
```

```
        <element name="event" type="uccp:AutoState" minOccurs="0" maxOccurs="4" />
    </sequence>
  </complexType>

  <complexType name="AutoState">
    <annotation>
      <documentation>
      State transition after event occurrence.
      </documentation>
    </annotation>
    <attribute name="name" use="required">
      <simpleType>
        <restriction base="token">
          <enumeration value="preview-reject" />
          <enumeration value="preview-timeout" />
          <enumeration value="preview-redirect" />
        </restriction>
      </simpleType>
    </attribute>
    <attribute name="onevent" type="IDREF" use="required">
      <annotation>
        <documentation>The state that is automatically entered after
        the event. It must refer to a defined state id.
        </documentation>
      </annotation>
    </attribute>
    <attribute name="postevent" type="IDREF" use="optional">
      <annotation>
        <documentation>
          The state after the timer. Must refer to one of the defined states by its id.
          If not specified, the state prior to the event will be restored.
          This element is only used if the corresponding duration time is set in CME.
        </documentation>
      </annotation>
    </attribute>
  </complexType>

  <complexType name="GenesysAgentState">
    <annotation>
      <documentation>Definition of Genesys agent state</documentation>
    </annotation>
    <all>
      <element name="reason" minOccurs="0" maxOccurs="1">
        <annotation>
          <documentation>Reason code of the state. Any string.</documentation>
        </annotation>
        <complexType>
          <simpleContent>
            <extension base="string">
              <attribute name="in">
                <annotation>
```

```
                    <documentation>
                        Whether to match or use the reason in AttributeReasons or
AttributeExtensions.
                    </documentation>
                </annotation>
                <simpleType>
            <restriction base="token">
          <enumeration value="reasons"></enumeration>
          <enumeration value="extensions"></enumeration>
              </restriction>
                </simpleType>
              </attribute>
              <attribute name="keyname" type="string" default="ReasonCode">
                <annotation>
                  <documentation>
                    Which data key to use
                  </documentation>
                </annotation>
              </attribute>
            </extension>
          </simpleContent>
        </complexType>
      </element>
    </all>
    <attribute name="ready" type="boolean" use="required">
      <annotation>
        <documentation>
 Whether the definition concerns "ready" or "not ready" agent state.
        </documentation>
      </annotation>
    </attribute>
    <attribute name="mode" type="uccp:GenesysAgentMode" default="none">
      <annotation>
        <documentation>
          Agent mode in the event/request. Not all modes may be supported by T-Server.
        </documentation>
      </annotation>
    </attribute>
  </complexType>

  <simpleType name="GenesysAgentMode">
    <restriction base="token">
      <enumeration value="none" />
      <enumeration value="manual" />
      <enumeration value="acw" />
      <enumeration value="legal" />
      <enumeration value="auxwork" />
      <enumeration value="away" />
      <enumeration value="back" />
    </restriction>
  </simpleType>
```

```
</schema>
```

![Genesys logo]

Related Documentation Resources

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

Management Framework

*   *Framework 8.1 Deployment Guide,* which provides information to configure, install, start, and stop Framework components.
*   *Framework 8.1 Configuration Options Reference Manual*, which provides descriptions of configuration options for Framework components.

Universal Routing

Consult these additional resources as necessary:

*   *Universal Routing 8.1 Reference Manual,* which contains descriptions of all routing strategy objects.
*   *Universal Routing 8.1 Strategy Samples,* which describes the sample strategies supplied with Universal Routing.
*   *Universal Routing 8.1 Business Process User's Guide,* which contains step-by-step instructions for using Interaction Routing Designer to design interaction workflows. It also describes the sample business processes.
*   *Universal Routing 8.1 Interaction Routing Designer Help,* which is a guide to Interaction Routing Designer, including the portion of it that designs interaction workflows and business processes.

T-Server

*   *Framework 8.1 SIP Server Deployment Guide*, which provides information to configure and install SIP Server.
*   *Framework 8.1 T-Server for Alcatel A4400/OXE Deployment Guide.*

- *Framework 8.1 T-Server for Cisco Unified Communications Manager Deployment Guide.*

Microsoft Lync Server

- Microsoft Lync Server 2010 Documentation Help File. This download contains a compiled help file (chm) of all the available Lync Server 2010 IT professional documentation on the Technical Library.

  `http://www.microsoft.com/en-ca/download/details.aspx?id=23888`

Microsoft Office Communications Server 2007 R2

- Microsoft Office Communications Server 2007 R2 Documentation. Latest version is available at:

  `http://www.microsoft.com/en-us/download/details.aspx?id=24402`

- Configuring Custom Tabs in Communicator 2007 R2. Latest version available at:

  `http://technet.microsoft.com/en-us/library/dd425110(office.13).aspx`

IBM Sametime 8.5

- IBM Lotus Sametime 8.5.x Documentation. Latest version is available at:

  `http://www.ibm.com/developerworks/lotus/documentation/sametime/index.html`

Open Standards

- *RFC 3261 SIP: Session Initiation Protocol*
- *RFC 3863 Presence Information Data Format (PIDF)*
- *RFC 3265 Session Initiation Protocol (SIP)-Specific Event Notification*

Genesys

- *Genesys Technical Publications Glossary,* which ships on the Genesys Documentation Library DVD and which provides a comprehensive list of the Genesys and computer-telephony integration (CTI) terminology and acronyms used in this document.

- *Genesys Migration Guide*, which ships on the Genesys Documentation Library DVD, and which provides documented migration strategies for Genesys product releases. Contact Genesys Technical Support for more information.

- Release Notes and Product Advisories for this product, which are available on the Genesys Technical Support website at `http://genesyslab.com/support`.

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

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

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

Genesys product documentation is available on the:

- Genesys Technical Support website at `http://genesyslab.com/support`.
- Genesys Documentation wiki at `http://docs.genesyslab.com/`.
- Genesys Documentation Library DVD and/or the Developer Documentation CD, which you can order by e-mail from Genesys Order Management at `orderman@genesyslab.com`.

Document Conventions

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

Document Version Number

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

80fr_ref_06-2008_v8.0.001.00

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

Screen Captures Used in This Document

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

Type Styles

Table 33 describes and illustrates the type conventions that are used in this document.

Table 33: 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 227).	Please consult the *Genesys Migration Guide* for more information. Do *not* use this value for this option. A *customary and usual* practice is one that is widely accepted and used within a particular industry or profession. The formula, $x + 1 = 7$ where x stands for . . .

Table 33: Type Styles (Continued)

Type Style	Used For	Examples
Monospace font (Looks like `teletype` or `typewriter text`)	All programming identifiers and GUI elements. This convention includes: • The *names* of directories, files, folders, configuration objects, paths, scripts, dialog boxes, options, fields, text and list boxes, operational modes, all buttons (including radio buttons), check boxes, commands, tabs, CTI events, and error messages. • The values of options. • Logical arguments and command syntax. • Code samples. Also used for any text that users must manually enter during a configuration or installation procedure, or on a command line.	Select the `Show variables on screen` check box. In the `Operand` text box, enter your formula. Click `OK` to exit the `Properties` dialog box. T-Server distributes the error messages in `EventError` events. If you select `true` for the `inbound-bsns-calls` option, all established inbound calls on a local agent are considered business calls. Enter `exit` on the command line.
Square brackets ([ ])	A particular parameter or value that is optional within a logical argument, a command, or some programming syntax. That is, the presence of the parameter or value is not required to resolve the argument, command, or block of code. The user decides whether to include this optional information.	`smcp_server -host [/flags]`
Angle brackets (< >)	A placeholder for a value that the user must specify. This might be a DN or a port number specific to your enterprise. **Note:** In some cases, angle brackets are required characters in code syntax (for example, in XML schemas). In these cases, italic text is used for placeholder values.	`smcp_server -host <confighost>`

Document Conventions

Index

Symbols

Index

UC Connector 8.0